



ВІСНИК

**КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Серія: ФІЗИКО-МАТЕМАТИЧНІ НАУКИ

Випуск № 1

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ВІСНИК

**КИЇВСЬКОГО НАЦІОНАЛЬНОГО
УНІВЕРСИТЕТУ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

СЕРІЯ ФІЗИКО-МАТЕМАТИЧНІ НАУКИ

Наукова бібліотека
ім. М. Максимовича
КНУ

ім. ТАРАСА ШЕВЧЕНКА



16586JB

філія (вул. Сеченова)

Ц 40.00

ВИПУСК №1 2011

**Вісник Київського національного університету імені Тараса Шевченка,
выпуск №1, 2011**

Серія фізико-математичні науки

З 1991 року серії вісників Київського університету “Математика і механіка”, “Фізика”, “Моделирование и оптимизация сложных систем” реорганізовано у “Вісник Київського університету. Серія: фізико-математичні науки”. У віснику містяться результати нових досліджень у різних галузях математики, інформатики, механіки, фізики та радіофізики для наукових працівників, викладачів, аспірантів, інженерів і студентів. Друкується за рекомендаціями Вчених Рад фізичного, радіофізичного, механіко-математичного факультетів та факультету кібернетики.

Журнал “Вісник Київського національного університету імені Тараса Шевченка. Серія фізико-математичні науки” включено до переліку фахових видань ВАК України.

Редакційна колегія:

- Анісімов Анатолій Васильович, член-кореспондент НАН України, доктор фізико-математичних наук, професор, **головний редактор**;
- Хусаїнов Денис Ях'євич, доктор фізико-математичних наук, професор, **заступник головного редактора, відповідальний за видання**;
- Акіменко Віталій Володимирович, доктор технічних наук, професор;
- Анісімов Ігор Олексійович, доктор фізико-математичних наук, професор;
- Буй Дмитро Борисович, доктор фізико-математичних наук, старший науковий співробітник;
- Булавін Леонід Анатолійович, академік НАН України, доктор фізико-математичних наук, професор;
- Волошин Олексій Федорович, доктор технічних наук, професор;
- Гаращенко Федір Георгійович, доктор технічних наук, професор;
- Данилов Вадим Васильович, доктор фізико-математичних наук, професор;
- Єжов Станіслав Миколайович, доктор фізико-математичних наук, професор;
- Заславський Володимир Анатолійович, доктор технічних наук, доцент;
- Кириченко Володимир Васильович, доктор фізико-математичних наук, професор;
- Козаченко Юрій Васильович, доктор фізико-математичних наук, професор;
- Кудін Володимир Іванович, доктор технічних наук, старший науковий співробітник;
- Левитський Сергій Михайлович, доктор фізико-математичних наук, професор;
- Макара Володимир Арсенійович, член-кореспондент НАН України, доктор фізико-математичних наук, професор;
- Макарець Микола Володимирович, доктор фізико-математичних наук, професор;
- Мелешко В'ячеслав Володимирович, доктор фізико-математичних наук, професор;
- Перестюк Микола Олексійович, академік НАН України, доктор фізико-математичних наук, професор;
- Погорілий Сергій Дем'янович, доктор технічних наук, професор;
- Скришевський Валерій Антонович, доктор фізико-математичних наук, професор.

Редакційний відділ:

- Анісімова Тетяна Харитонівна, **відповідальний секретар**;
- Безущак Оксана Омелянівна, bezusch@univ.kiev.ua;
- Мороз Костянтин Олександрович, morozko@univ.kiev.ua;
- Родіонова Тетяна Василівна, rodtv@univ.kiev.ua;
- Хмелюк Надія Кузьмівна, khmeluk@univ.kiev.ua;
- Сільвейструк Людмила Миколаївна, **технічний редактор**, slm-klm@ukr.net.

Адреса редакційної колегії:

Факультет кібернетики, Київський національний університет імені Тараса Шевченка,
пр. Глушкова, 4 д, 03680 Тел. (044) 259-01-49

ISBN 978-617-571-036-4
ISSN 1812-5409

ЗМІСТ

АЛГЕБРА, ГЕОМЕТРІЯ ТА ТЕОРІЯ ІМОВІРНОСТЕЙ

Гудзенко С.В. Автоморфізми напівгрупи $FP^+(\Omega_N)$	7
Довгей Ж.І. Вербальні підгрупи групи трикутних автоморфізмів кільця многочленів від двох змінних над полем характеристики	13
Касянюк М.В. Напівланцюгові квазіфробеніусові кільця	18
Леонов Ю.Г. Про 2-породженні нескінченні $\{2, p\}$ -групи	22
Швилов В. Про напівланцюгові справа нетерові зліва напівпервинні кільця	25

ДИФЕРЕНЦІАЛЬНІ РІВНЯННЯ, МАТЕМАТИЧНА ФІЗИКА ТА МЕХАНІКА

Богданов В.Р., Сулим Г.Т. Динамічний розвиток тріщини у компактному зразку за пружно-пластичною моделлю плоского напруженого стану	31
Григоренко О.Я., Єфімова Т.Л., Власова І.В. Власні коливання прямокутних анізотропних пластин змінної товщини	35
Кіпніс О.Л. Згин пружної клиноподібної пластини зосередженою силою	39
Краснопольська Т.С., Печук Є.Д. Побудова еволюційних рівнянь по даним вихідного сигналу	43
Ободан Н.І., Гук Н.А. Обернена задача визначення зовнішніх навантажень при деформації тонкостінних оболонок	47
Таран Є.Ю., Кондрат Р.Я. Вплив мікродинаміки недеформівних ланцюгових макромолекул у розведеному розчині на його макрореологію	51

КОМП'ЮТЕРНІ НАУКИ ТА ІНФОРМАТИКА

(The Proceedings of International Conference "Theoretical and Applied Aspects of Program Systems Development")

Діблік Й., Кухаренко О.В., Хусаїнов Д.Я. Зображення розв'язку першої крайової задачі системи рівнянь із сталим запізнюванням	59
Тарануха В.Ю., Порхун О.В. Автоматичне встановлення авторства текстів з використанням аналізу звукової організації мови	63
Бісікало О.В. Побудова нечітких відношення і простору сенсу образних конструкцій	70
Бондар Є.С., Глибовець М.М., Гороховський С.С. Хмарні обчислення та їх застосування	74
Борисенко О.А., Петров В.В. Матричне біноміальне кодування	82
Буй Д.Б., Глушко І.М. Узагальнена таблична алгебра, узагальнене числення рядків, узагальнене числення на домені та їх еквівалентність	86
Вижол Ю.О., Жорова А.М., Муленко І.О., Хомкін О.Л. Розробка системи аналітичних обчислень на ЕВМ та її застосування в кінетиці плазми	96
Галкін О.В., Верес М. М. Функціональне програмування. Лямбда числення в мовах FP та F#	103
Грунський І.С., Козловський В.А. Дескрипція автоматів їхньою поведінкою	108
Губа А.А. Методи синхронізації агентів при верифікації систем в термінах мультиагентних середовищ	115
Гученко І.В. Аналіз експертної бази на узгодженість методами Data Mining	119
Д'яченко Л.І., Мінов Є.В., Остапов С.Е., Фочук П.М. Програмний комплекс моделювання точкових дефектів у напівпровідникових кристалах	123
Єпіфанов А.С., Автоматна інтерпретація генетичних послідовностей	127
Іванов Є.В. Операційна семантика програм обробки складноіменних даних	133
Ковальов О.М., Савченко О.Я., Козловський В.А., Щербак В.Ф. Обернені системи керування в алгоритмах перетворення інформації	137
Кульчицький Ю.М. Паралельне програмування на основі скелетонів	145
Лавріщева К.М., Слабоспицька О.О., Колесник А.Л., Коваль Г.І. Теоретичні аспекти керування варіабельністю в сімействах програмних систем	151
Лілікович С.О., Логінов А.В. Аналіз систем шаблонізації сучасних CMS	159
Лук'янова О.О., Дереза А.В. Про автоматичну систему аналізу деяких властивостей алгоритмічних схем	163

Маєвський Д.А., Яремчук С.О. Перехідні процеси в лінійних системах та моделювання надійності програмного забезпечення	169
Міроненко Д.С. Вибір варіанту змінно-добового завдання для машинобудівного виробництва з використанням нечіткої моделі прийняття плану	173
Парасюк І.М., Єршов С.В. Мультиагентні системи на основі нечітких моделей: архітектура, координація та трансформації	179
Погорілий С.Д. До методології проектування алгоритмічного забезпечення новітніх паралельних архітектур обчислювальних систем	187
Пряничникова О.О. Алгебра мов, що можуть бути представлені в помічених графах	193
Тараненко К.Г., Коломійченко Т.В., Бочарников В.П. Автоматизація підтримки прийняття рішень у реальному часі в доказовій медицині	199
Твердохлебов В.О. Геометричні моделі автоматних відображень та автоматів	202
Ткачук І.Ю. Елементи формалізації Бі-методу	208
Туркін І.Б., Соколова Є.В., Юр'єва О. І. Афінний інтервальний метод автоматичного контролю достовірності телеметричної інформації	210
Чайка Л.О., Жихаревич В.В., Остапов С.Е. Криптостійкий генератор бінарної ключової послідовності на основі клітинних автоматів	215
Ченцов О.І. Програмна реалізація теорії категорій	220
Чичкарьов Є.А., Кривенко О.В. Розробка сервера обчислень і web-інтерфейсу для віддаленої роботи систем комп'ютерної математики в середовищі пакету Moodle	228
Шкуліпа І.Ю., Погорілий С.Д. Система автоматизованого параметричного проектування алгоритмів	232
Яковлева І. Д., Лісовенко І. Д., Кудринський З. Р. Автоматизована верифікація VHDL-моделей алгоритмічних операційних пристроїв швидкого перетворення Фур'є з фіксованою комою	237

РАДІОФІЗИКА

Богданов Р.В. Комп'ютерне моделювання розпорошувального магнетронного розряду: енергетичний розподіл бомбардуючих іонів	243
Костенко В. І., Нечипорук О.Ю., Сорочак А. М., Чамор Т. Г., Чевнюк Л. В. Експериментальне дослідження високочастотних властивостей магнітного гістерезису одновісних монокристалічних гексаферитів	249
Недибалюк О.А., Ольшевський С.В., Черняк В.Я, Забашта Ю.Ф., Актан О.Ю., Свечнікова О.С., Орловська С.Г., Карімова Ф.Ф., Шкоропато М.С. Стимульоване плазмою спалення парафіну в плазмово-динамічній системі	253
Первак Ю.О., Федоров В.В. Спектральні властивості багат шарових структур з трьома півхвильовими резонаторами	259

СУЧАСНА ФІЗИКА

Альохін О.Д. Структурні параметри надкритичного флюїду	265
Бардік В.Ю., Слинчак О.Л. Особливості розповсюдження звуку у розбавлених водних розчинах спиртів в околі їх особливих точок	269
Войтешенко І.С., Ніколаєнко Т.Ю., Булавін Л.А., Говорун Д.М. Пружні властивості низькомолекулярних сполук, що моделюють цукрово-фосфатний кістяк 2'- дезоксирибopolінуклеотидів	273
Гречко Л.Г., Породько Л.В., Лерман Л. Б., Семчук О. Ю. Просторовий та часовий розподіл температури на поверхні твердого тіла, викликаний одномірною лазер інтерференційною картинкою	277
Копань В.С., Хуторянська Н.В., Копань Т.В. Термоелектричний адсорбційний ефект в нікелевому порошку	281
Королович В.Ф., Пур Хосро П., Мороз К.О., Меленевський Д.О., Держипольський А.Г. Водні системи з вуглецевими нанотрубками: молекулярна структура і токсичність	285

АЛГЕБРА, ГЕОМЕТРІЯ ТА ТЕОРІЯ ІМОВІРНОСТЕЙ

УДК 519.5

С.В. Гудзенко

Автоморфізми напівгрупи $FP^+(\mathfrak{S}_N)$

У роботі розглядається напівгрупа $FP^+(\mathfrak{S}_N)$ — факторступінь симетричної групи зліченного порядку. Основним результатом є теорема про те, що всі автоморфізми $FP^+(\mathfrak{S}_N)$ — внутрішні.

Ключові слова: напівгрупа, факторступінь, автоморфізми.

E-mail: S.Gudzenko@ipnet.ua

Статтю представив д.ф.-м.н., професор В.В.Кириченко

1 Вступ

Поняття факторступеня $FP^+(S)$ вперше було введено в [1]. Факторступінь напівгрупи перетворень (S, M) визначається як факторнапівгрупа

$$P(S) / \sim_M,$$

де $P(S)$ — глобальна наднапівгрупа напівгрупи S , а конгруенція $\sim_M$ визначається наступним чином:

$$\forall A, B \in P(S) \quad A \sim_M B \Leftrightarrow \forall m \in M (A(m) = B(m)).$$

Таким чином, кожен елемент $A \in FP(S)$ повністю визначається множинами $A(1), A(2), A(3), \dots$, що дає нам спосіб задання елементів цієї напівгрупи.

Альтернативним способом задання елементів $FP(S)$ є задання за допомогою представників відповідних класів конгруенції $\sim_M$. Скориставшись тим, що серед конгруентних елементів із $P(S)$, впорядкованих за включенням, є найбільший, виберемо його як канонічного представника даного класу конгруентності $A \in FP(S)$. Завдяки цьому, для елементів факторступеня $FP(S)$ можна використовувати такі поняття, як включення, потужність та інші.

Ще одним способом є задання елемента $A \in FP(S)$ за допомогою орграфа на множині M . Із вершини i у вершину j існує стрілка тоді і тільки тоді, коли $j \in A(i)$.

Нас цікавить випадок, коли як множина перетворень S виступає симетрична група зліченного степеня $\mathfrak{S}_N$. Напівгрупа $FP^+(\mathfrak{S}_N)$ отримується з $FP(\mathfrak{S}_N)$ відкиданням нуля $\{\emptyset\}$, який є

S.V. Gudzenko

Automorphisms of semigroup $FP^+(\mathfrak{S}_N)$

The paper deals with the semigroup $FP^+(\mathfrak{S}_N)$ of a factorpower of the symmetric group of the denumerable order. The main result is the theorem, that all the automorphisms of $FP^+(\mathfrak{S}_N)$ are inner.

Key Words: semigroup, factorpower, automorphisms.

приєднанням:

$$FP^+(\mathfrak{S}_N) = FP(\mathfrak{S}_N) \setminus \{\emptyset\}.$$

Елементи $FP^+(\mathfrak{S}_N)$ називатимемо узагальненими підстановками.

Група $\mathfrak{S}_N$ природно занурюється в $FP^+(\mathfrak{S}_N)$ наступним чином: кожній підстановці $\pi \in \mathfrak{S}_N$ співставимо узагальнену підстановку $A \in FP^+(\mathfrak{S}_N)$ таку, що $A(i) = \{\pi(i)\}$. Крім того, для кожного $n \in \mathbb{N}$ $\mathfrak{S}_n$ природно занурюється в $\mathfrak{S}_N$. Тому вважатимемо, що $\mathfrak{S}_n \subset \mathfrak{S}_N \subset FP^+(\mathfrak{S}_N)$. Для зручності, щоб підкреслити належність елемента $FP^+(\mathfrak{S}_N)$ до $\mathfrak{S}_N$, іноді будемо позначати його маленькою літерою, у той час, як елементи $FP^+(\mathfrak{S}_N)$ зазвичай позначатимемо великими. $\mathfrak{S}_N$ є максимальною підгрупою $FP^+(\mathfrak{S}_N)$, яка складається з усіх оборотних елементів. Також $\mathfrak{S}_N$ можна визначити як піднапівгрупу $FP^+(\mathfrak{S}_N)$, яка складається з елементів потужності 1.

Напівгрупа $FP^+(\mathfrak{S}_N)$ містить нуль — повний орграф на множині N , і одиницю $e \in \mathfrak{S}_N$ — тотожну підстановку.

Основним результатом роботи є теорема 5.1 про те, що всі автоморфізми напівгрупи $FP^+(S(N))$ є внутрішніми.

2 Напівгрупа F

У напівгрупі $FP^+(S(N))$ визначимо наступну піднапівгрупу

$$F = \left\{ A \in FP^+(S(N)) : \inf_{i \in N} |A(i)| > 1 \right\}.$$

Лема 1. Для всіх $A \in F$ та $i \in \mathbb{N}$ виконується

$$A(\mathbb{N} \setminus \{i\}) = \mathbb{N}.$$

Доведення. Розглянемо довільні $A \in F$, $i \in \mathbb{N}$.

З означення F випливає, що $|A(i)| > 1$. Тому існують два такі числа $n_1 \neq n_2$, що $\{n_1, n_2\} \subseteq A$. Отже існують такі $a_1, a_2 \in A$, що $a_1(i) = n_1$, $a_2(i) = n_2$. Оскільки a_1, a_2 — елементи $\mathfrak{S}_{\mathbb{N}}$, то $a_1(\mathbb{N} \setminus \{i\}) = \mathbb{N} \setminus \{n_1\}$, $a_2(\mathbb{N} \setminus \{i\}) = \mathbb{N} \setminus \{n_2\}$. Тому

$$\begin{aligned} A(\mathbb{N} \setminus \{i\}) &\supseteq a_1(\mathbb{N} \setminus \{i\}) \cup a_2(\mathbb{N} \setminus \{i\}) = \\ &(\mathbb{N} \setminus \{n_1\}) \cup (\mathbb{N} \setminus \{n_2\}) = \mathbb{N}, \end{aligned}$$

бо $n_1 \neq n_2$, а отже $A(\mathbb{N} \setminus \{i\}) = \mathbb{N}$, що й вимагалося довести. $\square$

Лема 2. Для будь-яких $A, B \in F$ включення $A \subseteq B$ тоді і тільки тоді, коли для всіх $C, D \in FP^+(\mathfrak{S}_{\mathbb{N}})$ з рівності $CAD = 0$ випливає, що $CBD = 0$.

Доведення. Необхідність. Припустимо, що $A \subseteq B$. Очевидно, це рівносильно тому, що для всіх $i \in \mathbb{N}$ $A(i) \subseteq B(i)$. Розглянемо довільні $C, D \in FP^+(\mathfrak{S}_{\mathbb{N}})$. Тоді для будь-якого $i \in \mathbb{N}$

$$AD(i) = A(D(i)) \subseteq B(D(i)) = BD(i),$$

а тому

$$CAD(i) = C(AD(i)) \subseteq C(BD(i)) = CBD(i).$$

Якщо тепер $CAD = 0$, тобто для всіх натуральних i $CAD(i) = \mathbb{N}$, то і $CBD(i) = 0$, що й означає, що $CBD = 0$.

Достатність. Нехай тепер $A \not\subseteq B$. Тоді існує таке число i , що $A(i) \not\subseteq B(i)$, тобто існує така пара $(i, j) \in \mathbb{N}^2$, що $j \in A(i)$, $i \notin B(i)$. Побудуємо таку пару $C, D \in FP^+(\mathfrak{S}_{\mathbb{N}})$, що $CAD(1) \not\subseteq CBD(1)$.

Визначимо узагальнену перестановку D наступним чином

$$D = \overline{\{d \in \mathfrak{S}_{\mathbb{N}} : d(1) = i\}}.$$

Тоді $D(1) = \{i\}$, $D(k) = \mathbb{N} \setminus \{i\}$ для всіх $k \neq 1$. Тоді $AD(1) = A(i)$, $BD(1) = B(i)$, $AD(k) = A(\mathbb{N} \setminus \{i\}) = \mathbb{N}$ для всіх $k \neq 1$ — з леми 1. З останньої рівності випливає, що для всіх узагальнених підстановок C для всіх $k \neq 1$ виконується $CAD(k) = C(\mathbb{N}) = \mathbb{N}$.

Для множини B можливі два випадки.

$$\bullet B(i) = \mathbb{N} \setminus \{j\}.$$

В цьому випадку візьмемо

$$C = \overline{\{c \in \mathfrak{S}_{\mathbb{N}} : c(j) = 1\}}.$$

Тоді $C(j) = \{1\}$, $C(k) = \mathbb{N} \setminus \{1\}$ для всіх $k \neq j$. Маємо

$$\begin{aligned} CAD(1) &= C(A(i)) = C(j) \cup C(A(i) \setminus \{j\}) = \\ &\{1\} \cup (\mathbb{N} \setminus \{1\}) = \mathbb{N}, \end{aligned}$$

бо $A(i) \setminus \{j\} \neq \emptyset$.

$$CBD(1) = C(B(i)) = C(\mathbb{N} \setminus \{j\}) = \mathbb{N} \setminus \{1\}.$$

Отже, в даному випадку $CAD(1) = \mathbb{N}$, $CBD(1) \neq \mathbb{N}$.

$$\bullet B(i) \neq \mathbb{N} \setminus \{j\}.$$

В цьому випадку існує таке $m \neq j$, що $m \notin B(i)$.

Візьмемо

$$C = \overline{\{c \in \mathfrak{S}_{\mathbb{N}} : (c(j) = 1) \vee (c(m) = 1)\}}.$$

Тоді $C(j) = C(m) = \mathbb{N}$, $C(k) = \mathbb{N} \setminus \{1\}$ для всіх $k \neq j, k \neq m$. Маємо

$$CBD(1) = C(B(i)) = \mathbb{N} \setminus \{1\}.$$

З іншого боку

$$CAD(1) = C(A(i)) = C(j) = \mathbb{N},$$

отже $CAD(1) = \mathbb{N}$.

Отже, і в даному випадку $CAD(1) = \mathbb{N}$, $CBD(1) \neq \mathbb{N}$.

Таким чином в обох випадках існують такі узагальнені підстановки C і D , що $CAD = 0$, $CBD \neq 0$. $\square$

3 Мінімальний ненульовий ідеал $FP^+(\mathfrak{S}_N)$

Лема 3. Мінімальний ненульовий ідеал в $FP^+(\mathfrak{S}_N)$ існує, він єдиний і має вигляд

$$I_{\min} = \{0\} \cup \{A_{ij} | i, j \in N\},$$

де $A_{ij} \in FP^+(\mathfrak{S}_N)$ визначаються наступним чином:

$$A_{ij}(i) = N \setminus \{j\}, A_{ij}(k) = N \text{ для всіх } k \neq i.$$

Доведення. Легко бачити, що $A_{ij} = \{a \in \mathfrak{S}_N : a(i) \neq j\}$, отже, узагальнена підстановка A_{ij} визначена коректно.

Розглянемо тепер орграфи, які відповідають елементам з $I_{\min}$.

Елементу 0 відповідає повний оргграф.

Оргграф, який відповідає узагальненій підстановці A_{ij} отримується з повного вилученням стрілки (i, j) . Отже, для всіх $(i, j) \in N$ включення $A_{ij} \subseteq B$ рівносильне тому, що $B = A_{ij}$ або $B = 0$.

Покажемо, що $I_{\min}$ — ідеал в $FP^+(\mathfrak{S}_N)$. Розглянемо будь-які $A_{ij} \in I_{\min}; B, C \in FP^+(\mathfrak{S}_N)$ і візьмемо деякі $b \in B, c \in C$. Очевидно, що $bA_{ij}c \subseteq BA_{ij}C$. Легко бачити, що $bA_{ij}c = A_{c^{-1}(i)}b(j)$, тому $BA_{ij}C = A_{c^{-1}(i)}b(j)$ або $BA_{ij}C = 0$. Отже $BA_{ij}C \in I_{\min}$, і $I_{\min}$ є ідеалом $FP^+(\mathfrak{S}_N)$.

З іншого боку, для довільних $i, j, k, l \in N$, можна взяти такі $b, c \in \mathfrak{S}_N$ такі, що $c^{-1} = k, b(j) = l$, і отримати $bA_{ij}c = A_{kl}$. Тому $I_{\min}$ — мінімальний ненульовий ідеал.

Доведемо, що $I_{\min}$ — єдиний мінімальний ненульовий ідеал.

Розглянемо будь-яку узагальнену перестановку $A \notin I_{\min}$. Із побудови $I_{\min}$ випливає, що існує така пара $(i, j) \in N$, що $j \notin A(i)$. Візьмемо

$$C = \{c \in \mathfrak{S}_N : c(1) = i\}, D = \{d \in \mathfrak{S}_N : d(j) = 1\}.$$

Тоді $C(1) = \{i\}$, $C(k) = N \setminus \{i\}$ для всіх $k \neq 1$. Відповідно,

$$AC(1) = A(i), AC(k) = \bigcup_{m \neq i} A(m)$$

для всіх $k \neq 1$. Крім того, оскільки $j \notin A(i)$, то існує $l \neq i$ таке, що $j \in A(l)$, а тому $j \in AC(k)$ для всіх $k \neq 1$. Також $D(i) = 1$, $D(k) = N \setminus \{1\}$ для всіх $k \neq 1$, тому

$$DAC(1) = N \setminus \{1\},$$

$$DAC(k) = D(j) \cup D\left(\bigcup_{m \neq i} A(m) \setminus \{j\}\right) = \{1\} \cup (N \setminus \{1\}) = N.$$

Отже, $DAC = A_{11} \in I_{\min} \setminus \{0\}$. Тому, будь-який ненульовий ідеал містить $I_{\min}$. Таким чином, $I_{\min}$ — єдиний мінімальний ідеал в $FP^+(\mathfrak{S}_N)$. $\square$

Лема 4. Для кожного $A \in FP^+(\mathfrak{S}_N)$ наступні умови рівносильні:

- 1) $A \in F$;
- 2) $AI_{\min} = 0$;
- 3) $I_{\min}A = 0$.

Доведення. Покажемо, що з пункту 1) випливають 2) і 3). Візьмемо будь-які $A \in F, B \in I_{\min}$. Оскільки $B \in I_{\min}$, то для будь-якого $k \in N$ буде $B(k) = N$ або $B(k) = N \setminus \{j\}$, де j — деяке число. Тому, за лемою 1, $AB = 0$. Це доводить 2). З іншого боку, з того, що $B \in I_{\min}$, випливає, що для будь-якої не менш ніж двоелементної множини $N \in N$ виконується $B(N) = N$. Тому, за означенням F , $BA = 0$, що доводить 3).

Покажемо тепер, що для будь-якої узагальненої підстановки $A \notin F$ умови 2) і 3) не виконуються. Справді, оскільки $A \notin F$, то існують такі $m, n \in N$, що $A(m) = n$, а тому $A(N \setminus \{m\}) = N \setminus \{n\}$. Тоді, по-перше, $(AA_{1m})(1) = N \setminus \{n\}$, тому, $AA_{1m} \neq 0$. По-друге, $(A_{n1}A)(m) = A_{n1}(n) = N \setminus \{1\}$, тому $A_{n1}A \neq 0$. $\square$

4 Максимальні піднапівгрупи з $FP^+(\mathfrak{S}_N)$ з фіксованим нулем

Для довільного ідемпотента $E \in FP^+(\mathfrak{S}_N)$ позначимо

$$S_{\max}(E) = \{A \in FP^+(\mathfrak{S}_N) : AE = EA = E\}.$$

Легко бачити, що для всіх $A, B \in FP^+(\mathfrak{S}_N)$ виконується рівність $ABE = EAB = E$, а тому $AB \in FP^+(\mathfrak{S}_N)$. Тому $S_{\max}(E)$ є максимальною піднапівгрупою з $FP^+(\mathfrak{S}_N)$ з нулем E , і така піднапівгрупа — єдина.

Для будь-якої пари $(i, j) \in N^2$ позначимо $E_{ij} = \{e, (i, j)\}$. Очевидно, що E_{ij} — ідемпотент.

Лема 5. Якщо ідемпотент $E \in FP^+(\mathfrak{S}_N)$ містить дві різні неодиначні підстановки, то $S_{\max}(E)$ містить не менше 4 елементів.

Доведення. Нехай ідемпотент $E \in FP^+(\mathfrak{S}_N)$ містить дві різні неодиначні підстановки a_1, a_2 .

Розглянемо наступні чотири узагальнені підстановки:

$$B_1 = e, B_2 = \overline{\{e, a_1\}}, B_3 = \overline{\{e, a_2\}}, B_4 = \overline{\{e, a_1, a_2\}}.$$

Не обмежуючи загальності, можна вважати, що жодна з узагальнених підстановок B_2, B_3 не містить іншої. Дійсно, припустимо, наприклад, що $B_2 \subseteq B_3$. Це означає, що для всіх натуральних i $a_1(i) \in \{i, a_2(i)\}$, причому, оскільки $a_1 \neq a_2$, існують такі i , що $a_1(i) = i, a_2(i) \neq i$. Замінімо тоді a_2 на $a'_2 = a_1^{-1}a_2$. Маємо 1) $a'_2 \neq e$, бо $a_1 \neq a_2$; 2) $a'_2 \neq a_1$, бо інакше б виконувалося $a_1^2 = a_2$, і не існувало таких i , що $a_1(i) = i, a_2(i) \neq i$; 3) $a'_2 \in E$, бо для всіх i $a_1(i) \in \{i, a_2(i)\} \subseteq E(i)$. Таким чином a_1, a'_2 — дві різні неодиначні підстановки, що належать до E , і жодна з двох узагальнених підстановок $\overline{\{e, a_1\}}, \overline{\{e, a'_2\}}$ не містить в собі іншу.

Оскільки $\overline{\{e, a_1\}} \not\subseteq \overline{\{e, a'_2\}}$, то існує таке число i , що $1 \neq a_1(i) \neq a_2(i)$. Тоді маємо

$$B_3(i) = \{i, a_2(i)\}, B_4(i) = \{i, a_1(i), a_2(i)\},$$

і $B_3 \neq B_4$. Аналогічно, $B_2 \neq B_4$. Нерівність $B_i \neq B_j$ для будь-якої іншої пари $(i, j), i = \overline{1, 4}, j = \overline{1, 4}, i \neq j$ — очевидна, отже B_1, B_2, B_3, B_4 — всі різні.

Кожну з узагальнених підстановок $B_i, 1 \leq i \leq 4$ можна подати у вигляді $B_i = \overline{\{b_0, b_1, \dots, b_r\}}$ для деякого $r \geq 0$, причому $b_0 \in S_{\max}(E), b_i \in E(0 < i \leq r)$. Тоді для будь-якого числа k

$$b_0 E(k) = E(k), b_i E(k) \in E(k) (0 < i \leq r),$$

тому

$$\overline{\{b_0, b_1, \dots, b_r\}} E = (b_0 \cup b_1 \cup \dots \cup b_r) E =$$

$$b_0 E \cup b_1 E \cup \dots \cup b_r E = E.$$

Аналогічно, $E \overline{\{b_0, b_1, \dots, b_r\}} = E$, тому $\overline{\{b_0, b_1, \dots, b_r\}} \in S_{\max}(E)$. Отже, $\{B_1, B_2, B_3, B_4\} \subseteq S_{\max}(E)$.

Таким чином, $S_{\max}(E)$ містить щонайменше чотири різних елементи: B_1, B_2, B_3, B_4 , що й доводить лему. $\square$

Лема 6. 1) $|S_{\max}(E_{ij})| = 3$ для будь-яких $i, j \in \mathbb{N}$.

2) Якщо ідемпотент $E \in FP^+(\mathfrak{S}_N)$ неодиначний і $E \neq E_{ij}$ для якогось $i, j \in \mathbb{N}$, то $|S_{\max}(E_{ij})| > 3$.

Доведення. 1) Розглянемо довільні $i, j \in \mathbb{N}$ і такий елемент $A \in FP^+(\mathfrak{S}_N)$, що $AE_{ij} = E_{ij}A = E_{ij}$. Оскільки, за означенням, $E_{ij}(k) = k$ для всіх $k \neq i, k \neq j$, то й $A(k) = k$ для всіх таких k . Розглянемо $a \in A$. Очевидно, $a(i) \in \{i, j\}; a(j) \in \{i, j\}; a(k) = k, k \notin \{i, j\}$, а тому можливі лише два варіанти: або $a = e$, або $a = (i, j)$. Отже, $S_{\max} = \{e, (i, j), \{e, (i, j)\}\}$ і $|S_{\max}(E_{ij})| = 3$.

2) Розглянемо будь-який неодиначний ідемпотент $E \in FP^+(\mathfrak{S}_N)$ і деяку неодиначну підстановку $a \in E$.

Припустимо спочатку, що порядок a більший за 2. У цьому випадку a містить цикл довжини більшої за 2, і $a(i) \neq a^2(i)$, якщо i належить до цього циклу. Оскільки $a^2 \in EE = E$, то маємо включення $a^2 \in E$. За лемою 5, $S_{\max}(E)$ містить щонайменше 4 елементи.

Припустимо тепер, що порядок a дорівнює 2. Це значить, що a містить лише цикли довжини не більшої за 2.

Припустимо, що a містить два різних цикли довжини 2: (i, j) та (k, l) . Покладемо $a_1 = (i, j), a_2 = (k, l)$. Маємо $a, E \subseteq eE \cup aE = E, s = 1, 2$. Отже, $a_1, a_2 \in E$ і, як і в попередньому випадку, $S_{\max}(E)$ містить щонайменше чотири елементи.

Також, за лемою 5, якщо E містить дві різні ненульові підстановки, то порядок $S_{\max}(E)$ більший за 3.

Таким чином, в усіх випадках, коли E не має вигляду E_{ij} , напівгрупа $S_{\max}(E)$ містить більше 3 елементів. $\square$

5 Автоморфізми $FP^+(\mathfrak{S}_N)$

В [2] показано, що всі автоморфізми напівгрупи $FP^+(\mathfrak{S}_n)$ є внутрішніми.

Подальші міркування спираються на деякі факти, що стосуються введеного в [3] фінітарного факторстепеня $FP_{fin}^+(\mathfrak{S}_{fin}(\mathbb{N}))$. Вони визначаються як

$$FP_{fin}^+(\mathfrak{S}_{fin}(\mathbb{N})) = FP_{fin}(\mathfrak{S}_{fin}(\mathbb{N})) \setminus \{\emptyset\},$$

де

$$\mathfrak{S}_{fin}(\mathbb{N}) = \{\pi \in \mathfrak{S}_N : |\{i : \pi(i) \neq i\}| < \infty\},$$

$FP_{fin}(S) = \{A \in FP(S) : \text{існує така скінченна підмножина } \{\sigma_1, \dots, \sigma_k\} \subset S, \text{ що } A = \overline{\{\sigma_1, \dots, \sigma_k\}}\}$

У [4] показано, що всі автоморфізми $FP_{fin}^+(\mathfrak{S}_{fin}(N))$ індукуються підстановками з $\mathfrak{S}_N$.

Лема 7. Нехай $\varphi \in \text{Aut}(FP^+(S(N)))$. Тоді:

- 1) $\varphi(I_{min}) = I_{min}$
- 2) $\varphi(FP_{fin}^+(\mathfrak{S}_{fin}(N))) = FP_{fin}^+(\mathfrak{S}_{fin}(N))$;
- 3) $\varphi(F) = F$.

Доведення. Оскільки I_{min} — мінімальний нетривіальний ідеал, то і його образ повинен бути мінімальним нетривіальним ідеалом, а оскільки він — єдиний, то $\varphi(I_{min}) = I_{min}$. Пункт 1) доведено.

Нехай $A \in FP_{fin}^+(\mathfrak{S}_{fin}(N))$. За означенням $FP_{fin}^+(\mathfrak{S}_{fin}(N))$, множина $M = \{i \in N : A(i) \neq \{i\}\}$ — скінченна. Позначимо

$$E = \overline{\{a \in \mathfrak{S}_N : a(M) = M, a(k) = k \ \forall k \notin M\}}.$$

Тоді $E(k) = M$, якщо $k \in M$ і $E(k) = \{k\}$, якщо $k \notin M$. Очевидно, E — ідемпотент і $AE = EA = E$. Тому $A \in S_{max}(E)$. Очевидно також,

$$E = \prod_{(i,j) \in M^2} E_{ij},$$

і цей добуток — скінченний.

Нехай тепер $A \notin FP_{fin}^+(\mathfrak{S}_{fin}(N))$. Тоді M — нескінченна. Розглянемо будь-який ідемпотент E , який можна представити у вигляді $E = \prod_{(i,j) \in M^2} E_{ij}$, де $N \subseteq N^2$ — скінченна множина. З скінченності N випливає, що і $\{i : E(i) \neq \{i\}\}$ — також скінченна множина. Очевидно, $e \in E$. Тому $\{i : AE(i) \neq \{i\}\}$ — нескінченна множина, бо вона містить в собі M . Тому $AE \neq E$, а отже $A \notin S_{max}(E)$.

Таким чином, $A \notin FP_{fin}^+(\mathfrak{S}_{fin}(N))$ тоді і тільки тоді, коли існує ідемпотент E , який розкладається в скінченний добуток ідемпотентів вигляду E_{ij} , $(i, j) \in N^2$ і $A \in S_{max}(E)$.

Ідемпотент вигляду E_{ij} , $(i, j) \in N^2$ при будь-якому автоморфізмі $\varphi \in \text{Aut}(FP^+(S(N)))$, очевидно, переходить у ідемпотент. При цьому порядки відповідних напівгруп $S_{max}(E_{ij})$ і $S_{max}(\varphi(E_{ij}))$ співпадають. Тому, за лемою 6, ідемпотент вигляду

E_{ij} , $(i, j) \in N^2$ переходить в ідемпотент такого ж вигляду. Тому властивість ідемпотентів розкладатися в скінченний добуток ідемпотентів вигляду E_{ij} , $(i, j) \in N^2$ також зберігається при будь-якому автоморфізмі. Отже, будь-який автоморфізм $FP^+(S(N))$ переводить $FP_{fin}^+(\mathfrak{S}_{fin}(N))$ в себе. Доведено пункт 2).

За лемою 4, $A \in F$ тоді і тільки тоді, коли $AI_{min} = 0$. Подіємо φ на обидві частини останньої рівності: $\varphi(AI_{min}) = 0$. Оскільки I_{min} — єдиний мінімальний ідеал, то $\varphi(I_{min}) = I_{min}$, тому $\varphi(A)I_{min} = 0$. За лемою 4, остання рівність означає, що $\varphi(A) \in F$, що й доводить пункт 3). $\square$

Теорема 5.1. Всі автоморфізми $FP^+(\mathfrak{S}_N)$ — внутрішні.

Доведення. Нехай $\varphi \in \text{Aut}(FP^+(\mathfrak{S}(N)))$. За лемою 7,

$$\varphi(FP_{fin}^+(\mathfrak{S}_{fin}(N))) = FP_{fin}^+(\mathfrak{S}_{fin}(N)).$$

Тому можна розглянути $\varphi|_{FP_{fin}^+(\mathfrak{S}_{fin}(N))}$, і це буде автоморфізм напівгрупи $FP_{fin}^+(\mathfrak{S}_{fin}(N))$.

В [4] доведено, що всі автоморфізми $FP_{fin}^+(\mathfrak{S}_{fin}(N))$ визначаються елементами $\mathfrak{S}_N$, тобто існує таке $\tau \in \mathfrak{S}_N$, що для всіх $A \in FP_{fin}^+(\mathfrak{S}_{fin}(N))$ $\varphi(A) = \tau^{-1}A\tau$. Визначимо φ_τ наступним чином: $\forall A \in FP^+(\mathfrak{S}_N)$

$$\varphi_\tau(A) = \tau^{-1}A\tau.$$

φ_τ є автоморфізмом $FP^+(S_N)$. За побудовою, φ_τ і φ співпадають на $FP_{fin}^+(\mathfrak{S}_{fin}(N))$.

Співставимо кожній узагальненій підстановці $A \in I_{min}$ наступну пару множин:

$$L(A) = \{B \in FP_{fin}^+(\mathfrak{S}_{fin}(N)) : BA = 0\},$$

$$R(A) = \{B \in FP_{fin}^+(\mathfrak{S}_{fin}(N)) : AB = 0\}.$$

Розглянемо деякі $A_1, A_2 \in I_{min}$, $A_1 \neq A_2$. Покажемо, що $(L(A_1), R(A_1)) \neq (L(A_2), R(A_2))$.

Припустимо, що одна з цих двох узагальнених підстановок, наприклад A_1 , дорівнює 0. Тоді, очевидно, $\mathfrak{S}_{fin}(N) \subseteq L(A_1)$. Оскільки $A_1 \neq A_2$, то $A_2 \neq 0$. Тому існують такі числа i, j , що $j \notin A_2(i)$, а тому $(i, j) \notin A_2$. Оскільки $(i, j) \in \mathfrak{S}_{fin}(N)$, то $\mathfrak{S}_{fin}(N) \not\subseteq L(A_1)$. Тому $L(A_1) \neq L(A_2)$, зокрема $(L(A_1), R(A_1)) \neq (L(A_2), R(A_2))$.

Припустимо тепер, що A_1, A_2 — ненульові. Тоді $A_1 = A_{i_1 j_1}, A_2 = A_{i_2 j_2}$ для деяких $i_1, i_2, j_1, j_2 \in \mathbb{N}$, і $i_1 \neq i_2$ або $j_1 \neq j_2$.

Якщо $i_1 \neq i_2$, то візьмемо

$$B = \overline{\{e, (i_1, m)\}}$$

для деякого $m \neq i_1, m \neq i_2$. Маємо

$$B(i_1) = \{i_1, m\}, B(i_2) = \{i_2\}.$$

Тому $A_1 B = 0, A_2 B \neq 0$, отже $B \in R(A_1), B \notin R(A_2)$.

Якщо ж $i_1 = i_2$, то $j_1 \neq j_2$. Візьмемо

$$B = \overline{\{e, (j_1, m)\}}$$

для деякого $m \neq j_1, m \neq j_2$. Маємо

$$B(\mathbb{N} \setminus \{j_1\}) = \mathbb{N}, B(\mathbb{N} \setminus \{j_2\}) = \mathbb{N} \setminus \{j_2\}.$$

Тому $BA_1 = 0, BA_2 \neq 0$, отже $B \in L(A_1), B \notin L(A_2)$.

Отже, і в цьому випадку $(L(A_1), R(A_1)) \neq (L(A_2), R(A_2))$.

Оскільки для всіх $A \in I_{\min} L(A), R(A) \in FP_{fin}^+(\mathfrak{S}_{fin}(\mathbb{N}))$ то

$$\varphi(L(A)) = \varphi_\tau(L(A)),$$

$$\varphi(R(A)) = \varphi_\tau(R(A)).$$

Оскільки $\varphi, \varphi_\tau \in \text{Aut}(FP^+(S_{\mathbb{N}}))$, то з означення $L(A), R(A)$ випливає, що

$$\varphi(L(A)) = L(\varphi(A)), \varphi(R(A)) = R(\varphi(A)),$$

$$\varphi_\tau(L(A)) = L(\varphi_\tau(A)), \varphi_\tau(R(A)) = R(\varphi_\tau(A)).$$

Оскільки узагальнена підстановка $A \in I_{\min}$ однозначно визначається парою множин $L(A), R(A)$, то

$$\varphi(A) = \varphi_\tau(A).$$

Співставимо кожній узагальненій підстановці $A \in F$ множину $N(A) = \{B \in I_{\min} : A \subseteq B\}$. Розглянемо деякі $A_1, A_2 \in F, A_1 \neq A_2$. З останньої нерівності випливає, що існує така пара $(i, j) \in \mathbb{N}^2$, що, наприклад, $j \in A_1(i), j \notin A_2(i)$. Очевидно, що $A_1 \notin A_{ij}, A_2 \in A_{ij}$, отже $A_{ij} \notin N(A_1), A_{ij} \in N(A_2)$, і $N(A_1) \neq N(A_2)$. Як і в попередньому випадку, можливість однозначно визначити $A \in F$ через множину $N(A)$ дає нам рівність

$$\varphi(A) = \varphi_\tau(A).$$

Розглянемо тепер загальний випадок $A \in FP^+(S_{\mathbb{N}})$. Кожній узагальненій підстановці A співставимо наступні множини: $K_A = \{i \in \mathbb{N} : |B(i)| = 1\}, M_A = \{i \in \mathbb{N} : |B(i)| > 1\} =$

$\mathbb{N} \setminus K_B$. Виберемо будь-яку підстановку $a \in A$ і позначимо $B = a^{-1}A$. Очевидно, $e \in K_B$, тому $K_B = \{i \in \mathbb{N} : B(i) = \{1\}\}$. Очевидно також, що $B(K_B) = K_B$. Крім того, для кожного $b \in B$ $b(K_B) = K_B$, а тому $b(M_B) = M_B$, і $B(M_B) = M_B$.

Для всіх $i, j \in \mathbb{N}$

$$BA_{ij} \neq 0 \Leftrightarrow i \in K_B.$$

Отже, множина K_B однозначно визначається набором $\{C \in I_{\min} : BC \neq 0\} \subseteq I_{\min}$, а тому, $K_{\varphi(B)} = K_{\varphi_\tau(B)}$. Аналогічно, $M_{\varphi(B)} = M_{\varphi_\tau(B)}$. Отже, автоморфізми φ і φ_τ можна розкласти в пряму суму ізоморфізмів

$$\varphi_\tau = \varphi_\tau|_{FP^+(S_{K_B})} \oplus \varphi_\tau|_{FP^+(S_{M_B})},$$

$$\varphi = \varphi|_{FP^+(S_{K_B})} \oplus \varphi|_{FP^+(S_{M_B})}.$$

При цьому B_{K_B} — одиниця, а тому і його образ повинен бути одиничним елементом як під дією автоморфізма φ , так і під дією автоморфізма φ_τ . Отже, образи $B|_{K_B}$ під дією автоморфізмів $\varphi_\tau|_{FP^+(S_{K_B})}$ і $\varphi|_{FP^+(S_{K_B})}$ збігаються.

З іншого боку, $FP^+(M_B) \simeq FP^+(S_{\mathbb{N}})$, якщо M_B — нескінченна, і $FP^+(M_B) \simeq FP^+(S_n)$ для деякого $n \in \mathbb{N} \cup \emptyset$, якщо M_B — скінченна. Крім того, $\forall i \in m_B |B|_{M_B}(i)| > 1$. Тому образи $B|_{M_B}$ під дією автоморфізмів $\varphi_\tau|_{FP^+(S_{M_B})}$ і $\varphi|_{FP^+(S_{M_B})}$ збігаються.

Отже,

$$\varphi(B) = \varphi_\tau(B),$$

а тому й

$$\varphi(A) = \varphi_\tau(A),$$

що й треба було довести. $\square$

Список використаних джерел

1. Ганюшкин А.Г., Мазорчук В.С. Факторстепени и индуцированные действия полугрупп преобразований // В кн.: Третья международная конференция по алгебре. Сб. тезисов. Красноярск, — 1993. — с. 83 - 84.
2. V. Mazorchuk. All Automorphisms of $FP^+(S_n)$ are Inner. // Semigroup Forum, — 2000. — Vol. 60. — P. 486 - 490.
3. Гудзенко С. В. Відношення Гріна на факторстепенях симетричної групи // Вісник Київського ун-ту. Серія: фіз.-мат. науки. — 2004. — 2. — с. 24 - 44.
4. Гудзенко С. В. Автоморфізми фінітарного факторстепеня нескінченної симетричної групи // УМЖ, подано до друку.

Надійшла до редколегії 28.04.2010

УДК 512.54

Ж. І. Довгей, аспірант

**Вербальні підгрупи групи трикутних
автоморфізмів кільця многочленів
від двох змінних над полем
характеристики 0**

*Доведено, що кожна вербальна підгрупа
групи автоморфізмів кільця многочленів від
двох змінних над алгебраїчно замкненим по-
лем характеристики нуль збігається з де-
яким членом ряду комутантів цієї групи.*

Ключові слова: вербальна підгрупа, група
трикутних автоморфізмів кільця многочле-
нів.

E-mail: janascusureas@mail.ru

Статтю представив доктор фіз.-мат. наук, проф. Кириченко В.В.

1 Вступ

Групи автоморфізмів кілець многочленів від скінченного числа змінних над полями складають важливий клас груп, який знаходить застосування в алгебраїчній геометрії, теорії динамічних систем, теорії асоціативних алгебр, гіперболічній геометрії і інших розділах математики (див. [1] - [5]). Випадок двох змінних відрізняється від загального тим, що ця група може бути в явний спосіб сконструйована з двох своїх підгруп: підгрупи трикутних автоморфізмів, яку ще називають групою Жонк'єра та підгрупи афінних автоморфізмів. Це дає змогу більш докладно вивчати теоретико-групові властивості групи автоморфізмів кільця многочленів $K[x, y]$, K - деяке поле, і останнім часом цілий ряд робіт різних авторів присвячено її вивченню. Серед них зазначимо роботи [6] - [8], в яких вивчаються класи спряженості елементів і скінченних абелевих підгруп цієї групи та праці [9] - [11], в яких розглядаються геометричні аспекти її дослідження.

Метою даної публікації є дослідження вербальної будови групи Жонк'єра над полем характеристики нуль. Виявляється, що для алгебраїчно замкненого поля характеристики нуль, вона досить бідна: кожна вербальна підгрупа цієї групи збігається з деяким членом її ряду

Zh.I. Dovghey, graduate student

**Verbal subgroup of the unitriangle
automorphism group of the polynomial
ring from two variables over a fields of
zero characteristic**

*It is shown that every verbal subgroup of the
automorphism group of polynomial ring in two
variables over an algebraically closed field of zero
characteristic coincide with some term of derived
series of this group.*

Key Words: the verbal subgroup, the triangle
automorphism group, the polynomial ring.

комутантів. На випадок полів характеристики p це твердження безпосередньо не переноситься.

Всі позначення в роботі загальноприйняті; для визначення понять, пов'язаних з вербальними підгрупами див. [12].

2 Автоморфізми Жонк'єра

Нехай K - фіксоване поле, $K[x, y]$ - алгебра многочленів від двох змінних x, y над полем K , $G = \text{Aut } K[x, y]$ - група автоморфізмів цієї алгебри. Кожен автоморфізм групи G однозначно визначається образами твірних x, y :

$$x \mapsto a(x, y), y \mapsto b(x, y), \quad (1)$$

які повинні бути такими многочленами з $K[x, y]$, щоб для відображення $K[x, y]$ в себе, визначеного відповідністю (1), існувало обернене такого ж вигляду. Далі позначатимемо автоморфізми, які задаються відповідностями (1), як пари многочленів $(a(x, y), b(x, y))$.

Виділяються два природні класи автоморфізмів в групі G .

1) Афінні автоморфізми.

Це автоморфізми, які задаються парами многочленів вигляду

$$(\alpha_1 x + \beta_1 y + \gamma_1, \alpha_2 x + \beta_2 y + \gamma_2),$$

де $\alpha_1, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2$ – довільні елементи поля K такі, що $\begin{vmatrix} \alpha_1 & \beta_1 \\ \alpha_2 & \beta_2 \end{vmatrix} \neq 0$.

2) **Трикутні автоморфізми або автоморфізми Жонк'єра.**

Автоморфізм $\langle a(x, y), b(x, y) \rangle$ називається трикутним, якщо його перша компонента не залежить від змінної y . З умови оборотності випливає, що в такому разі многочлен a повинен бути лінійним, тобто мати вигляд $\alpha x + \beta$. Враховуючи це, легко дістаємо, що многочлен $b(x, y)$ також повинен мати "трикутний" вигляд $\alpha y + f(x)$, $f(x) \in K[x]$. А тому, автоморфізмами Жонк'єра називають автоморфізми, що задаються парами вигляду

$$\langle \alpha_1 x + \beta_1, \alpha_2 y + f(x) \rangle,$$

де $\alpha_1, \beta_1, \alpha_2 \in K, \alpha_1 \cdot \alpha_2 \neq 0, f(x) \in K[x]$.

Автоморфізм із групи G називається ручним, якщо він є добутком афінних автоморфізмів і автоморфізмів Жонк'єра. Відома теорема (див. напр. [2], [11]) стверджує, що кожен автоморфізм із групи G є ручним, тобто G породжується своїми підгрупами афінних автоморфізмів і автоморфізмів Жонк'єра. А саме, нехай A – підгрупа афінних автоморфізмів із групи G , а J – підгрупа автоморфізмів Жонк'єра. Тоді перетин $C = A \cap J$ складається з автоморфізмів вигляду

$$\langle \alpha_1 x + \gamma_1, \alpha_2 x + \beta_2 y + \gamma_2 \rangle,$$

де $\alpha_1, \gamma_1, \alpha_2, \beta_2, \gamma_2 \in K, \alpha_1 \cdot \beta_2 \neq 0$.

Теорема 2.1. Для довільного поля K група $\text{Aut} K[x, y]$ розкладається на амальгамований вільний добуток своїх підгруп A, J з об'єднаною підгрупою C :

$$G = A *_C J. \quad (2)$$

Доведення див. [11].

Група $\text{Aut} K[x, y]$ ізоморфна групі автоморфізмів вільної асоціативної алгебри $A_K \langle x, y \rangle$ з двома твірними x, y над K (див. [4], [5]), а тому ці групи можна вивчати одночасно. Оскільки $\text{Aut} K[x, y]$ є амальгамованим вільним добутком, то з нею можна стандартним чином асоціювати дерево Басса – Серра [18]. За допомогою техніки, пов'язаної з дією групи на дереві, було встановлено, що група $\text{Aut} K[x, y]$ містить різноманітні "цікаві" підгрупи: вільні підгрупи, підгрупи, що ізоморфні фундаментальним групам поверхонь, HNN – розширення та ін.

3 Елементарні властивості групи Жонк'єра

Кожен автоморфізм Жонк'єра алгебри $K[x, y]$ задається парою многочленів вигляду

$$\langle \alpha_1 x + \beta_1, \alpha_2 y + b(x) \rangle,$$

де $\alpha_1, \alpha_2 \in K \setminus \{0\}, \beta_1 \in K, b(x) \in K[x]$.

Множення пар такого вигляду здійснюється за правилом

$$\begin{aligned} & \langle \alpha_1 x + \beta_1, \alpha_2 y + b(x) \rangle \cdot \langle \gamma_1 x + \delta_1, \gamma_2 y + d(x) \rangle = \\ & = \langle (\alpha_1 \gamma_1)x + (\beta_1 \gamma_1 + \delta_1), \alpha_2 \gamma_2 y + \gamma_2 b(x) + d(\alpha_1 x + \beta_1) \rangle, \end{aligned} \quad (3)$$

а оберненим до автоморфізму вказаного вигляду є

$$\langle \alpha_1^{-1}x - \frac{\beta_1}{\alpha_1}, \alpha_2^{-1}y \cdot \alpha_2 b(\alpha_1^{-1}x - \frac{\beta_1}{\alpha_1}) \rangle. \quad (4)$$

Автоморфізми Жонк'єра вигляду $\langle x + \beta, y + b(x) \rangle$ називатимемо унітрикутними.

Лема 1. Множина всіх унітрикутних автоморфізмів Жонк'єра є нормальним дільником групи $J_2(K)$.

Доведення. З рівностей (3), (4) випливає, що множина всіх унітрикутних автоморфізмів є підгрупою в $J_2(K)$. Трансформа $u^{-1}vu$ автоморфізму $v = \langle x + \alpha, y + b(x) \rangle$ за допомогою автоморфізму $u = \langle \beta x + \gamma, \delta y + c(x) \rangle$ є автоморфізмом

$$\begin{aligned} & \langle x + \alpha\beta, y - \delta^2(c(\beta^{-1}x - \frac{\gamma}{\beta}) + b(\beta^{-1}x - \frac{\gamma}{\beta})) + \\ & + c(\beta^{-1}x - \frac{\gamma}{\beta} + \alpha) \rangle, \end{aligned}$$

який міститься в підгрупі унітрикутних автоморфізмів. Отже, підгрупа всіх унітрикутних автоморфізмів є нормальним дільником в групі $J_2(K)$. $\square$

Нормальний дільник всіх унітрикутних автоморфізмів позначатимемо $UJ_2(K)$.

Зрозуміло, що має місце ізоморфізм

$$J_2(K)/UJ_2(K) \simeq K^* \times K^*,$$

де K^* – мультиплікативна група поля K . Правила множення і обертання (3), (4) для унітрикутних автоморфізмів спрощуються. А саме, добуток унітрикутних автоморфізмів $\langle x + \alpha, y + a(x) \rangle, \langle x + \beta, y + b(x) \rangle$ знаходиться з рівності

$$\langle x + \alpha, y + a(x) \rangle \cdot \langle x + \beta, y + b(x) \rangle = \langle x + \alpha + \beta, y + a(x) + b(x + \alpha) \rangle, \quad (5)$$

а оберненим до $\langle x + \alpha, y + a(x) \rangle$ буде автоморфізм

$$\langle x - \alpha, y - a(x - \alpha) \rangle. \quad (6)$$

Символом $H_2(K)$ позначатимемо підгрупу групи $UJ_2(K)$, яка складається з автоморфізмів вигляду $\langle x, y + a(x) \rangle$. З (5), (6) випливає, що $H_2(K)$ є нормальним дільником $UJ_2(K)$, причому $H_2(K)$ ізоморфна адитивній групі кільця $K[x]$. Фактор-група $J_2(K)/H_2(K)$ очевидним чином ізоморфна адитивній групі кільця K .

Теорема 3.1. Нормальний ряд

$$J_2(K) \supset UJ_2(K) \supset H_2(K) \supset E,$$

де E – тривіальна підгрупа в $J_2(K)$, збігається з рядом комутантів групи $J_2(K)$.

Доведення. Згідно з (5) фактор-група $J_2(K)/H_2(K)$ є абелевою, тобто $UJ_2(K)$ містить комутант групи $J_2(K)$. З іншого боку для довільного власного нормального дільника підгрупи $UJ_2(K)$ відповідна фактор-група абелевою не буде. А тому $UJ_2(K)$ – найбільша за включенням підгрупа групи $J_2(K)$, фактор-група за якою є абелевою. Отже, це комутант групи $J_2(K)$. Цілком аналогічно перевіряється, що $H_2(K)$ – максимальна підгрупа групи $UJ_2(K)$, фактор-група за якою абелева. А тому $H_2(K)$ є комутантом $UJ_2(K)$. Оскільки $H_2(K)$ – абелева, то все доведено. $\square$

Наслідок 1. Група $J_2(K)$ є розв'язною довжини 3, а група $UJ_2(K)$ – метабелевою.

Група Жонк'єра $J_2(K)$ містить підгрупу $T_2(K)$ автоморфізмів вигляду $\langle \alpha x, \beta y \rangle, \alpha\beta \neq 0$, яка ізоморфна прямому добутку $K^* \times K^*$.

Лема 2. Група $J_2(K)$ розкладається в напівпрямий добуток

$$J_2(K) = T_2(K) \ltimes UJ_2(K) \quad (7)$$

своїх підгруп $T_2(K)$ та $UJ_2(K)$.

Доведення. Очевидно маємо $T_2(K) \cap UJ_2(K) = E$. Оскільки $UJ_2(K) \triangleleft J_2(K)$ і $\langle UJ_2(K), T_2(K) \rangle = J_2(K)$, то має місце рівність (7). $\square$

4 Вербальні підгрупи групи Жонк'єра $J_2(K)$

Нагадаємо, що вербальною підгрупою групи G , яка відповідає деякій множині групових слів V (підмножині елементів вільної групи $F[x_1, x_2, \dots]$ нескінченного рангу) називається (див. [12]) підгрупа, породжена всіма значеннями слів із V в групі G . Кожна вербальна підгрупа групи G є цілком інваріантною в ній, але не навпаки – група G може містити цілком інваріантні не вербальні підгрупи.

Всі вербальні підгрупи групи G утворюють ґратку, проте вона не буде підґраткою в ґратці всіх підгруп групи G , оскільки перетин вербальних підгруп може і не бути вербальною підгрупою в даній групі.

Якщо $\varphi : G \rightarrow H$ – епіморфізм, то образом вербальної підгрупи групи G буде вербальна підгрупа групи H , яка визначається тим же набором слів, що й перша підгрупа. А тому гомоморфізм φ індукує гомоморфізм $\bar{\varphi}$ ґратки всіх вербальних підгруп групи G на ґратку всіх вербальних підгруп групи H .

Лема 3. Кожне групове слово $v \in F[x_1, x_2, \dots]$ рівносильне парі слів v_1, v_2 , де v_1 має вигляд x^m , а слово v_2 є комутаторним.

Доведення див. [12].

Лема 4. Для довільної групи G при будь-якому натуральному k ґратки вербальних підгруп груп G і $G^k = G \times \dots \times G$ (k разів) ізоморфні.

Доведення. Для довільного набору слів V вербальна підгрупа групи G^k , яка задається набором слів V визначається рівністю

$$V(G^k) = V(G)^k.$$

А тому відображення

$$V(G) \rightarrow V(G^k), V \in F[x_1, x_2, \dots],$$

є ізоморфним відображенням ґратки вербальних підгруп групи G на ґратку вербальних підгруп групи G . $\square$

Зауваження 1. Лема 4 залишається правильною, якщо k -тий степінь групи G замінити на її злічений обмежений прямий степінь. Але вона стає неправильною для (необмеженого) декартового степеня групи G .

Лема 5. Нехай K – поле характеристики нуль. Тоді кожна вербальна підгрупа адитивної групи поля K є або тривіальною або збігається з групою K .

Доведення. Оскільки кожне групове слово еквівалентне парі слів, одне з яких має вигляд x^m , а друге – комутаторним, то досить обмежитися розглядом слів такого вигляду. Комутаторне слово породжує в адитивній групі поля K тривіальну підгрупу, адже вона абелева, а слово вигляду x^m , m – довільне натуральне число, породжує всю цю групу, оскільки вона є подільною. $\square$

Лема 6. Нехай K є алгебраїчно замкненим полем характеристики нуль. Кожна вербальна підгрупа групи K^* є або одиничною або збігається з усією групою K^* .

Доведення. Як і в доведенні попередньої леми досить розглянути 2 випадки. Комутаторні слова тут також породжують тривіальну підгрупу $\{1\}$, а слова вигляду x^m при довільному натуральному m породжують всю групу K^* , адже в ній добуваються корені будь-якого степеня з довільного елемента. $\square$

Зауваження 2. Якщо поле K не є алгебраїчно замкненим, то твердження леми 6 для нього немає місця.

Приклад 1. Мультиплікативна група поля дійсних чисел R містить власну вербальну підгрупу, яка породжується словом x^m , де m – парне число. Ця підгрупа збігається з підгрупою R^+ . Таким чином, ґратка вербальних підгруп групи R має вигляд (рис.1)



Рис. 1. Ґратка вербальних підгруп групи R .

Використаємо ці властивості для доведення такої теореми.

Теорема 4.1. Нехай K є алгебраїчно замкненим полем характеристики нуль. Тоді кожна вербальна підгрупа групи $J_2(K)$ збігається з деяким членом ряду комутантів цієї групи.

Доведення. Нехай $H = v(J_2(K))$ – деяка вербальна підгрупа групи $J_2(K)$. Розглянемо її образ при дії природного гоморфізму $\varphi : J_2(K) \rightarrow J_2(K)/UJ_2(K)$. Оскільки має місце ізоморфізм $J_2(K)/UJ_2(K) \simeq K^* \times K^*$, то $\varphi(V(J_2(K))) = V(\varphi(J_2(K))) = V(K^* \times K^*)$.

Враховуючи леми 4 і 6 звідси дістаємо, що $\varphi(H)$ або тривіальне, або збігається з усією групою $J_2(K)$. Тривіальність $\varphi(H)$ означає, що $H \subseteq UJ_2(K)$. Припустимо, що H є власною підгрупою в $UJ_2(K)$. Знову розглядаючи природний гомоморфізм $\psi : UJ_2(K) \rightarrow UJ_2(K)/H_2(K)$ дістанемо, що $\psi(H)$ ізоморфна деякій підгрупі адитивної групи поля K . Оскільки ця підгрупа є вербальною, то за лемою 5 або збігається з $UJ_2(K)$ або міститься в $H_2(K)$. Залишилося розглянути останню з можливостей. Якщо H міститься в $H_2(K)$, то або $H = H_2(K)$, або згідно з зауваженням 1 і лемою 5 підгрупа H_2 – тривіальна. Таким чином $H = J_2(K)$ або $H = UJ_2(K)$, або $H = H_2(K)$ або ж H – тривіальна підгрупа. Теорему доведено. $\square$

У випадку не алгебраїчно замкненого поля характеристизація ґратки вербальних підгруп може бути дана в термінах приєднання ґраток.

Означення 4.1. Приєднанням ґратки Γ_1 до ґратки Γ_2 називається ґратка Γ утворена отождивленням найменшого елемента Γ_1 з найбільшим елементом Γ_2 (якщо такий існує), причому всі елементи Γ_2 вважаються меншими від кожного з елементів Γ_1 , а впорядкованість елементів Γ_1 і елементів Γ_2 зберігається.

Позначатимемо цю ґратку символом $\Gamma_1 \circ \Gamma_2$.

Теорема 4.2. Нехай K – поле характеристики 0. Ґратка вербальних підгруп групи Жонк'єра $J_2(K)$ має вигляд $\Gamma_1 \circ \Gamma_2$, де Γ_1 – ґратка вербальних підгруп мультиплікативної групи поля K , а Γ_2 – ланцюг довжини 3.

Доведення. В доведенні теореми 4.1 алгебраїчна замкненість поля K використовувалася лише на кроці, коли було встановлено, що вербальна підгрупа $J_2(K)$ або збігається з нею,

або міститься в $UJ_2(K)$. В подальших міркуваннях алгебраїчна замкненість K не використовувалась. А тому на першому кроці міркувань вербальна підгрупа $H = v(J_2(K))$ може в даному випадку бути власною підгрупою $J_2(K)$, але такою, що містить $UJ_2(K)$. А її гомоморфний образ при природному гомоморфізмі $\varphi : J_2(K) \rightarrow J_2(K)/UJ_2(K)$ буде певною вербальною підгрупою цієї фактор-групи, тобто групи $K^* \times K^*$. Оскільки ця відповідність однозначна, то ґратка тих вербальних підгруп групи $J_2(K)$, які містять $UJ_2(K)$ є ізоморфною ґратці вербальних підгруп групи $K^* \times K^*$, яка, в свою чергу, за лемою 4 ізоморфна ґратці вербальних підгруп групи K^* . Це й означає, що ґратка вербальних підгруп групи $J_2(K)$ утворюється в результаті приєднання до ґратки Γ_1 ґратки всіх вербальних підгруп групи $UJ_2(K)$, яка є ланцюгом довжини 3. Теорему доведено. $\square$

Приклад 2. Ґратка вербальних підгруп групи $J_2(R)$ має вигляд (рис.2), де $M(R)$ – підгрупа перетворень вигляду

$$\langle \alpha_1 x + \beta_1, \alpha_2 y + \beta_2(x) \rangle,$$

причому $\alpha_1, \alpha_2 \in R^*, \beta_1 \in R, \beta_2 \in R[x]$.

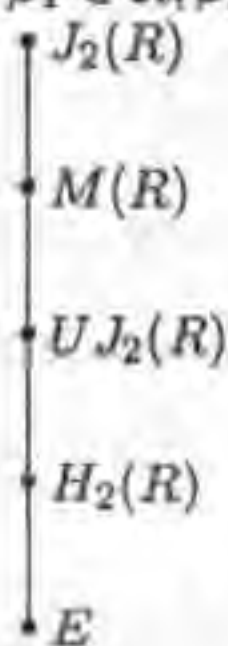


Рис. 2. Ґратка вербальних підгруп групи $J_2(R)$

Список використаних джерел

1. Arno van der Essen. Polynomial Automorphisms and the Jacobian Conjecture. – Basel: Birkhauser Verlag, 2000. – 329 p.

2. Alexander A.Mikhalev, Vladimir Shpilrain, Jie-Tai Yu. Combinatorial Methods. Free groups, Polynomials and Free Algebras. – New-York etc: Springer, 2004. – 314 p.
3. Shmuel Friedland, John Milnor. Dynamical properties of plane polynomial automorphisms. – Ergod. Th. & Dynam. Syst., 9(1989). – P. 67-99.
4. A.G.Czerniakiewicz. Automorphisms of free Associative Algebra of rang 2. I,II. – Trans. Amer. Math.Soc., 160 (1971), P. 393-401, 171 (1972). – P. 309-315
5. Л. Макаp-Лиманов. Автоморфизмы свободной ассоциативной алгебры с двумя образующими, Функциональный анализ и приложения. – 4(1970), 3. – С. 107-108.
6. J.Blanc. Finite abelian subgroups of the Cremona group of the plane. C.R.Acad.Sci.Paris,ser.I,344(2007), 21-26
7. J.Blanc, Lianerization of finite abelian subgroups of Cremona group of the plane. Groups, Geom.,dyn. 3(2009)N2, 215-266
8. T.de fernex, On planar Cremona maps of prime order. Nagoya Math. J.,174(2004)324-336
9. M.alberich-Carramanana, Geometry of the plane Cremona maps. Lect. Notes in Math. 1769, Berlin etc: Springer,2002
10. S.Lami, L'alternative de Tits pour $AutC^2$. J.Algebra, 239(2001) 413-437
11. S.Lami, Une preuve geometrique du theoreme de Jung. Enseign. Math, 48 (2002), 291-325
12. X. Нейман. Многообразия груп. – Москва : Мир, 1969. – 264 с.

Надійшла до редколегії 5.02.2010

УДК 512.552

М. В. Касянюк, аспірант

Напівланцюгові квазіфробеніусові кілець

У даній роботі представлені напівланцюгові квазіфробеніусові кілець як приклад квазіфробеніусових кілець. Напівланцюгові квазіфробеніусові кілець охарактеризовані за довжиною проєктивних нерозкладних модулів.

Ключові слова: напівланцюгове квазіфробеніусове кільце, перестановка Накаяма, моноідеальний ідеал, двосторонній розклад Пірса.

E-mail: marinasayenko@gmail.com

Статтю представив доктор фізико-математичних наук Кириченко В. В.

1 Вступ

Припустимо, всі кілець мають одиничний елемент. Для нерозкладного кілець це означає, що 0 і 1 є лише центральними ідемпотентними елементами кілець. Для правого модуля M над артиновим кільцем R з базисом основних ідемпотентів $e_1, e_2, \dots, e_n$ вектор $[d_1(M), d_2(M), \dots, d_n(M)]$ називається розмірністю вектора M , де $d_i(M)$ – довжина $e_i R e_i$ -модуля $M e_i$. Якщо існує проєктивне покриття модуля M , то проєктивне покриття запишемо $P(M)$. Кільце R називається строго квазіфробеніусовим, якщо $e R e$ квазіфробеніусовий для кожного ідемпотентного елемента $e \in R$. Аналогічно, кожне квазіфробеніусове кільце – квазіфробеніусове. Кільце R називається ланцюговим, якщо воно артинове і для кожного основного ідемпотентного елемента $e \in R$ структура підмодуля $e R$ і структура підмодуля $R e$ повністю впорядковані.

2 Характеристика напівланцюгових квазіфробеніусових кілець

Наступна лема відіграє центральну роль в характеристиці напівланцюгових квазіфробеніусових кілець.

Лема 1. Нерозкладне напівланцюгове кільце квазіфробеніусове тоді і тільки тоді, якщо всі

M. V. Kasyanuk, PhD student

On serial QF rings

A serial QF ring as an example of QF ring is considered. We characterize the serial QF rings by the composition length of the projective indecomposable modules.

Key Words: a serial QF ring, Nakayama permutation, a monomial ideal, a two-sided Peirce decomposition.

проєктивні нерозкладні праві R -модулі мають однакову довжину.

Доведення. Нехай $e_1, e_2, \dots, e_n$ – базис нерозкладних ідемпотентних елементів для R . Тоді індекси можна впорядкувати таким чином, що $e_{i+1} R \cong P(\text{rad}(e_i R))(1 \leq i \leq n-1)$, якщо R – нерозкладне. Отже,

$$|e_{i+1} R| \geq |e_i R| - 1 \quad (1 \leq i \leq n-1)$$

$$|e_1 R| \geq |e_n R| - 1.$$

Ці нерівності строгі, якщо R – квазіфробеніусове. Таким чином,

$$|e_1 R| \leq |e_2 R| \leq \dots \leq |e_n R| \leq |e_1 R|,$$

і всі проєктивні нерозкладні праві R -модулі мають однакову довжину. Обернене твердження береться з того факту, що кожен скінченновимірний правий модуль над напівланцюговим кільцем – це пряма сума однорядних модулів. Ми охарактеризували напівланцюгове квазіфробеніусове кільце R за довжиною проєктивних нерозкладних правих R -модулів.

Теорема 2.1. Нехай A – нерозкладне напівланцюгове артинове кільце. Кільце A – квазіфробеніусове тоді і тільки тоді, якщо сагайдак $Q(A)$ кілець A – цикл, а довжини всіх головних правих і лівих модулів рівні.

Доведення. Припустимо, A – квазіфробеніусове. Оскільки A – нерозкладне, то $Q(A)$ не має

кінцевих точок. Отже, $Q(A)$ -цикл. Крім того, якщо $\varphi: P_j \rightarrow R_i$ -епіморфізм головного A -модуля P_j на радикал головного A -модуля P_i , то φ -не мономорфізм (інакше P_j як ін'єктивний A -модуль відрізнявся б з P_i прямим доданком, що неможливо). Тому

$$l_i = l(P_j) > l(R_i) = l(P_i) - 1, \text{ або } l_j \geq l_i.$$

Враховуючи, що $Q(A)$ -цикл, отримаємо, що $l_1 \leq l_2 \leq \dots \leq l_s \leq l_1$, тобто всі l_i -однакові.

Навпаки, нехай $Q(A)$ -цикл і $l_1 = \dots = l_s = l$. Нехай $P = P_i$ -головний правий A -модуль, M_k -його єдиний підмодуль такий, що $l(P/M_k) = k$ (очевидно, $M_k = PR^k$, де $R = \text{rad } A$). Для зручності позначимо $P_{s+1} = P_1, P_{s+2} = P_2$ і т.д. Тоді $P(M_1) \simeq P_{i+1}$, значить, M_2 -епіморфний образ R_{i+1} , звідки $P(M_2) \simeq P_{i+2}$ і т.д., взагалі $P(M_k) \simeq P_{i+k}$ (якщо тільки $M_k \neq 0$). Зокрема, $\text{soc } P = M_{l-1}$, звідки $P(\text{soc } P_i) = P_{i+l-1}$. Очевидно, при різних $i = 1, \dots, s$ модулі P_{i+l-1} неізоморфні. Отже, для будь-якого головного правого A -модуля P_i його цюколь-простий модуль і при $P_i \neq P_j$ $\text{soc } P_i \neq \text{soc } P_j$. Отже, A -квазіфробеніусове кільце, що і треба було довести.

3 Мономіальні ідеали і перестановки Накаяма напівдосконалих кілець

Нехай $1 = f_1 + \dots + f_s$ -канонічний розклад рівності напівдосконалого кільця A радикалу Джекобсона R . Тоді $A_A = fA \oplus \dots \oplus f_s A$ (${}_A A = Af_1 \oplus \dots \oplus Af_s$), де $f_i A = P_i^{n_i}$ ($Af_i = Q_i^{n_i}$) для $i = 1, \dots, s$ називається канонічним розкладом кільця в пряму суму його головних правих(лівих) A -модулів. Нехай $A_{ij} = f_i Af_j$ ($i, j = 1, \dots, s$). Двосторонній розклад Пірса кільця A

$$A = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1s} \\ A_{21} & A_{22} & \dots & A_{2s} \\ \vdots & \vdots & \ddots & \vdots \\ A_{s1} & A_{s2} & \dots & A_{ss} \end{pmatrix} \quad (1)$$

називається канонічним двостороннім розкладом Пірса. Будь-який канонічний розклад можна отримати з (1) простою перестановкою рядків і стовпчиків і заміною всіх елементів Пірса A_{ij} на $aA_{ij}a^{-1}$ з деяким оберненим елементом $a \in A$.

Нехай $1 = e_1 + \dots + e_n$ -розклад 1 в суму попарно ортогональних ідемпотентів. Ідеа-

лом в цьому випадку позначимо двосторонній ідеал. Для деякого ідеалу I з A абелева група $e_i I e_j$ ($i, j = 1, \dots, n$) очевидно лежить в I і

$$I = \bigoplus_{i,j=1}^n I_{ij} -$$

це розклад I в пряму суму абелевих підгруп. Такий розклад називається двостороннім розкладом Пірса I , відповідно $1 = e_1 + \dots + e_n$. Він має наступну матричну форму:

$$I = \begin{pmatrix} I_{11} & I_{12} & \dots & I_{1n} \\ I_{21} & I_{22} & \dots & I_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ I_{n1} & I_{n2} & \dots & I_{nn} \end{pmatrix}.$$

Якщо

$$J = \bigoplus_{i,j=1}^n J_{ij} -$$

також ідеал, тоді

$$I+J = \begin{pmatrix} I_{11}+J_{11} & I_{12}+J_{12} & \dots & I_{1n}+J_{1n} \\ I_{21}+J_{21} & I_{22}+J_{22} & \dots & I_{2n}+J_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ I_{n1}+J_{n1} & I_{n2}+J_{n2} & \dots & I_{nn}+J_{nn} \end{pmatrix},$$

а кожен елемент Пірса $(IJ)_{ij}$ добутку IJ береться з

$$(IJ)_{ij} = \sum_{k=1}^n I_{ik} J_{kj}$$

($i, j = 1, \dots, n$) так, що додавання і множення елементів I і J можна отримати додаванням і множенням відповідних матриць.

Нехай A -напівдосконале кільце і нехай $1 = f_1 + \dots + f_s$ -канонічний розклад $1 \in A$ в суму попарно ортогональних ідемпотентів. Тоді

$$I = \bigoplus_{i,j=1}^s I_{ij},$$

де $I_{ij} = f_i I f_j$ ($i, j = 1, \dots, s$) називається канонічним двостороннім розкладом Пірса I . Один канонічний розклад Пірса I можна отримати з іншого простою перестановкою рядків і стовпчиків і заміною кожного елемента Пірса I_{ij} на $aI_{ij}a^{-1}$, де $a \in A$.

Означення 1.3. Ідеал I напівдосконалого кільця A будемо називати мономіальним, якщо

кожен рядок і кожен стовпчик канонічного двостороннього розкладу Пірса I містить тільки один ненульовий елемент Пірса.

Якщо I -мономіальний ідеал, то існує перестановка ν з $\{1, \dots, s\}$ така, що $I_{i\nu(i)} \neq 0$. Точніше, ν -єдиновизначений спряжений елемент симетричної групи, що містить s літер. Позначимо таку перестановку через $\nu(I)$.

Лема 2. Нехай A -напівдосконале кільце. Якщо I -мономіальний ідеал A , тоді кожен канонічний двосторонній елемент Пірса I -це ідеал A .

Доведення. Нехай $1 = f_1 + \dots + f_s$ -канонічний розклад $1 \in A$ в суму попарно ортогональних ідемпотентів. Запишемо $\nu = \nu(I)$, тоді

$$I = \bigoplus_{i=1}^s f_i I f_{\nu(i)}.$$

Очевидно, $f_i I f_{\nu(i)} f_k A f_l = 0$, якщо $k \neq \nu(i)$. Більш того, $f_i I f_{\nu(i)} f_{\nu(i)} A f_l \subseteq f_i I f_l$, який ненульовий тоді і тільки тоді, якщо $l = \nu(i)$, поки I -мономіальний. Аналогічно, $f_k A f_l f_i I f_{\nu(i)} \neq 0$ тоді і тільки тоді, якщо $k = l = i$. З цього випливає, що $f_i I f_{\nu(i)}$ -це ідеал A для кожного $i = 1, \dots, s$.

Нехай $P_1, \dots, P_s$ -неізоморфні головні праві A -модулі і нехай $Q_1, \dots, Q_s$ -неізоморфні головні ліві A -модулі двостороннього артинового кільця A . І нехай $A_A = P_1^{n_1} \oplus \dots \oplus P_s^{n_s}$ (${}_A A = Q_1^{n_1} \oplus \dots \oplus Q_s^{n_s}$)-розклад правого(лівого) регулярного A -модуля в пряму суму головних правих(лівих) A -модулів.

Означення 1.4. Двостороннє артинове кільце A називається квазіфробеніусовим, якщо A допускає перестановку Накаяма $\nu(A)$ з $\{1, 2, \dots, s\}$. Квазіфробеніусове кільце A називається фробеніусовим, якщо $n_{\nu(i)} = n_i$ для всіх $i = 1, \dots, s$.

Точніше, ν визначає зв'язок симетричної групи з s літер, а зв'язки відповідають пронумерованим головним ідеалам $P_1, \dots, P_s$.

Побудуємо приклад квазіфробеніусового кільця. Локальне кільце O з єдиним ненульовим максимальним правим ідеалом M називається дискретним кільцем оцінки, якщо кільце не має нульових дільників, праві ідеали O формують єдиний ланцюг:

$$O \supset M \supset M^2 \supset \dots \supset M^n \supset \dots$$

і, більш того, цей ланцюг є також єдиним ланцюгом лівих ідеалів A . Тоді, очевидно, O -нетерове, але не артинове, всі степені M різні і $\bigcap_{k=1}^{\infty} M^k = 0$. Більш того, M -головний, як правий (лівий) ідеал.

Приклад. Запишемо кільце $H_s(O)$ всіх матриць $s \times s$ наступним чином:

$$H = H_s(O) = \begin{pmatrix} O & O & \dots & O & O \\ M & O & & \vdots & \vdots \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & \ddots & O & O \\ M & M & \dots & M & O \end{pmatrix}$$

Очевидно, що радикал R кільця $H_s(O)$ є

$$R = \begin{pmatrix} M & O & \dots & \dots & O \\ M & M & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & O \\ M & M & \dots & M & M \end{pmatrix}$$

і

$$R^2 = \begin{pmatrix} M & M & O & \dots & \dots & O \\ M & M & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & O \\ M & M & \dots & & M & M \\ M^2 & M & \ddots & \dots & M & M \end{pmatrix}.$$

Головні праві модулі H -це ідеали рядків H і підмодулі кожного з них формують ланцюг. Зокрема, підмодулі ідеалів першого рядка формують наступний ланцюг:

$$\begin{pmatrix} O & O & \dots & O \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix} \supset \begin{pmatrix} M & O & \dots & O \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix} \supset \dots$$

$$\supset \begin{pmatrix} M & \dots & M & O \\ 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & 0 \end{pmatrix} \supset \begin{pmatrix} M & M & \dots & M \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

Легко побачити, що кожен наступний рядок ідеалу H ізоморфний підмодулю його модуля. Аналогічно, головні ліві H -модулі є ідеалами стовпчиків, підмодулі яких формують відповідні ланцюги. Тому H -напівланцюгове кільце. Нехай $P_1, \dots, P_s$ головні праві модулі кільця відношення $A = H_s(O)/R^2$ і нехай $Q_1, \dots, Q_s$ головні ліві A -модулі, пронумеровані таким чином, що $P_i = e_{ii}A$, $Q_i = Ae_{ii}$ ($i = 1, \dots, s$), де e_{ij} позначає елементарну матрицю $s \times s$, де (i, j) -та позиція дорівнює 1, поки всі інші позиції дорівнюють 0. Тоді кожен підмодуль P_i і Q_i утворюють скінченні ланцюги і звичайна перевірка показує, що

$$\text{soc } P_1 \cong \text{top } P_2, \text{soc } P_2 \cong \text{top } P_3, \dots,$$

$$\text{soc } P_s \cong \text{top } P_1$$

$$\text{top } Q_1 \cong \text{soc } Q_2, \text{top } Q_2 \cong \text{soc } Q_3, \dots,$$

$$\text{top } Q_s \cong \text{soc } Q_1.$$

Більш того, кожен з цих модулів-одно-мірний векторний простір над O/M . Отже, A -квазіфробеніусове кільце з перестановкою Накаяма $(1, 2, \dots, s)$.

В загальному, кільце відношень $A = H_s(O)/R^m$ ($m \geq 2$) квазіфробеніусове з перестановкою Накаяма $(1, 2, \dots, s)^{m-1}$. Зокрема, з цього слідує, що A має таку ж перестановку, як і перестановка Накаяма тоді і тільки тоді, якщо $m \equiv 1 \pmod{s}$.

Теорема 3.1. Нехай A -квазіфробеніусове кільце. Тоді $\text{soc } ({}_A A) = \text{soc } (A_A)$. Більш того, $Z = \text{soc } ({}_A A)$ -мономіальний ідеал і $\nu(Z)$ співпадає з перестановкою Накаяма $\nu(A)$ з A .

Теорема 3.2. Нехай A -двостороннє артинове кільце. Тоді A -квазіфробеніусове кільце тоді і тільки тоді, якщо виконуються наступні умови:

- 1) цю роль будь-якого головного A -модуля простий;
- 2) якщо P_1 і P_2 -неізоморфні головні A -модулі, тоді $\text{soc } P_1 \not\cong \text{soc } P_2$.

Нехай A -квазіфробеніусове напівланцюгове кільце, $A_A = P_1^{n_1} \oplus \dots \oplus P_s^{n_s}$ -розклад регулярного A -модуля в пряму суму попарно неізоморфних нерозкладних проєктивних A -модулів. За визначенням напівланцюгового кільця кожен A -модуль P_i ($i = 1, \dots, s$) є ланцюговим. За визначенням квазіфробеніусового кільця кільце A є артиновим. Кільце A еквівалентне в сенсі Моріти кільцю $B = \text{End}_A(P_1 \oplus \dots \oplus P_s)$. Кільце B -квазіфробеніусове напівланцюгове зведене кільце, тобто $B/\text{rad } B \cong D_1 \times \dots \times D_s$, де $D_1, \dots, D_s$ -тіла. Тому припустимо, що $A_A = P_1 \oplus \dots \oplus P_s$, тобто кратності всіх модулів $P_1, \dots, P_s$ дорівнюють одиниці.

Приклад. Нехай $\mathcal{O}$ -дискретно нормоване кільце з єдиним максимальним ідеалом. $\mathfrak{M} = \pi\mathcal{O} = \mathcal{O}\pi$,

$$A = H_2(\mathcal{O}) = \begin{pmatrix} \mathcal{O} & \mathcal{O} \\ \pi\mathcal{O} & \mathcal{O} \end{pmatrix} = P_1 \oplus P_2,$$

де $P_1 = (\mathcal{O}, \mathcal{O})$, $P_2 = (\pi\mathcal{O}, \mathcal{O})$. Нехай R -радикал Джекобсона кільця $H_2(\mathcal{O})$. Тоді

$$R = \begin{pmatrix} \pi\mathcal{O} & \mathcal{O} \\ \pi\mathcal{O} & \pi\mathcal{O} \end{pmatrix} \text{ і } R^2 = \begin{pmatrix} \pi\mathcal{O} & \mathcal{O} \\ \pi^2\mathcal{O} & \pi\mathcal{O} \end{pmatrix}.$$

З визначення сагайдака напівдосконалого кільця випливає, що сагайдак $Q(A)$ кільця $A = H_2(\mathcal{O})$ є таким циклом:

$$Q(A) = \{ \bullet \rightleftharpoons \bullet \}. \text{ Розглянемо двосторонній ідеал } I \subset R^2, \text{ де } I = \begin{pmatrix} \pi^2\mathcal{O} & \pi\mathcal{O} \\ \pi^2\mathcal{O} & \pi\mathcal{O} \end{pmatrix}.$$

Нехай $\bar{A} = A/I$. Очевидно, що $\text{soc } \bar{A} = U_1 \oplus U_1$, крім того, $l(\bar{P}_1) = 3$ і $l(\bar{P}_2) = 2$. Таким чином, існує напівланцюгове артинове кільце $\bar{A}$ з сагайдаком, циклом і різними довжинами нерозкладних проєктивних модулів.

Список використаних джерел

1. Michiel Yazewinkel, Nadya Gubareni, V. V. Kirichenko *Algebras, Rings and Modules, volume 2*, Netherlands: Springer.-2007.- p. 166-178
2. Ю.А.Дрозд, В.В.Кириченко, Конечномерные алгебры, Киев "Вища школа". - 1980.- с. 165

Надійшла до редколегії 29.11.2010

УДК 512.54

Леонов Ю.Г., к.ф.-м.н., доц.

Про 2-породженні нескінченні $\{2, p\}$ -групи

У роботі вперше будуються нескінченні скінченно-породжені $\{2, p\}$ -групи.

Ключові слова: періодична група, нескінченна група, автоморфізм дерева.

leonov_yu@yahoo.com

Статтю представив д.ф.-м.н., професор В.В. Кириченко

Вступ

Клас періодичних скінченно-породжених нескінченних груп, або груп типу Бернсайда, складається з цікавих та досить важливих прикладів груп. Ці групи часто відповідають на важливі питання в теорії груп (див., наприклад, [1], [2]), та мають деякі екстремальні властивості. Особливо цікавими та вже добре вивченими є скінченно-породжені нескінченні p -групи автоморфізмів регулярних p -дерев. В цій роботі ми конструємо нескінченні 2-породжені періодичні групи, які є $\{2, p\}$ -групами, та діють на кореневих деревах, які не є p -деревами.

Група $G_{p,2}$

Нехай T – довільне кореневе дерево. Будемо казати, що вершина дерева розтошовується на рівні n , якщо довжина цієї вершини до корня дерева дорівнює n . Піддерево дерева T , яке має своїм коренем вершину рівня n , назовемо піддеревом рівня n . Скажемо, що валентність вершини рівня n дорівнює k , якщо від цієї вершини ісходить рівно k ребер, які з'єднують цю вершину з вершинами дерева рівня $n + 1$. Кожній вершині дерева T припишемо деяку послідовність – код v , якій складається наступним чином. Код корня дерева дорівнює $\emptyset$ (пуста послідовність). Якщо вершина валентності k має код v довжини n , то k "нащадків" цієї вершини рівня $n + 1$ мають відповідно (у певному порядку) коди (послідовності) $v0, v1, \dots, v(k-1)$.

Нехай $T_{p,2}$ – кореневе (нескінченне) дерево, у якому кожна вершина парного рівня має валентність p , а кожна вершина непарного рівня – валентність 2. Будемо розглядати також дерево

Yu. Leonov, assoc. prof.

On infinite 2-generated $\{2, p\}$ -groups

Infinite finite-generated $\{2, p\}$ -groups is constructed at first time.

Key Words: torsion group, infinite group, tree automorphism.

$T_{2,p}$, у якому, навпаки, кожна вершина парного рівня має валентність 2, а кожна вершина непарного рівня – валентність p . Означимо піддерево дерева $T_{p,2}$ з кодом кореня v , як $T_{p,2}(v)$. Зрозуміло, що якщо довжина послідовності v парна, то дерево $T_{p,2}(v)$ можна ототожнити з деревом $T_{p,2}$, а якщо непарна, то з деревом $T_{2,p}$.

Розглянемо групи автоморфізмів $\text{Aut } T_{p,2}$, та $\text{Aut } T_{2,p}$. Нехай $\text{St}(\text{Aut } T_{p,2})$ – підгрупа групи $\text{Aut } T_{p,2}$, яка складається з автоморфізмів дерева, які залишають піддерева першого рівня на місці. Така підгрупа називається стабілізатором групи $\text{Aut } T_{p,2}$. Для будь-якого елемента стабілізатора g можна розглядати звуження його дії на піддерева $T_{p,2}(i)$, $i = 0, \dots, p-1$. Позначимо таке звуження як $g|_i$. Ясно, що $g|_i \in \text{Aut } T_{2,p}$.

Нехай $b \in \text{Aut } T_{p,2}$ елемент групи, який діє на дерево перестановками: піддерева $T_{p,2}(i)$ ставить на місце піддерева $T_{p,2}(i+1)$, для $i = 0, \dots, p-2$, а піддерева $T_{p,2}(p-1)$ – на місце $T_{p,2}(0)$. Таким чином, елемент b є циклом довжини p . За аналогією, будемо розглядати елемент $a \in \text{Aut } T_{2,p}$, який діє на дерево, обмінюючи між собою місцями піддерева $T_{2,p}(0)$ та $T_{2,p}(1)$. Розглянемо елемент $\beta \in \text{St}(\text{Aut } T_{p,2})$, який означимо рекурентним чином за допомогою його звужень на піддерева. А саме, покладемо $\beta|_0 = \alpha \in \text{Aut } T_{2,p}$, $\beta|_i = a$, $1 \leq i \leq p-1$. Елемент α у свою чергу означимо за допомогою звужень таким чином: $\alpha|_0 = \beta \in \text{Aut } T_{p,2}$, $\alpha|_1 = b$. Розглянемо групу $G_{p,2} = \langle \beta, b \rangle$.

Теорема 1. Група $G_{p,2}$ є нескінченною періодичною групою.

Доведення. Для доведення нашого твердження, розглянемо допоміжну групу $G_{2,p} = \langle \alpha, a \rangle$, яка також буде нескінченною та періодичною.

Зауважимо, що порядок $|b|$ елементу b дорівнює p , а $|a| = 2$. Легко перевіряється також, що $|\alpha| = |\beta| = 2p$. Позначимо, $\beta_i = b^{-i}\beta b^i$, $0 \leq i \leq p-1$. Легко бачити, що $\beta_i|_i = \alpha$, $\beta_i|_j = a$, для $j \neq i$. Покладемо також $\alpha_i = a^{-i}\alpha a^i$, $i = 0, 1$. Зрозуміло, $\alpha_i|_i = \beta$, $\alpha_i|_j = b$, для $j \neq i$. Нехай $g \in G_{p,2}$. Тоді, g можна розглянути у вигляді слова виду $g = \beta_{i_1}^{j_1} \cdots \beta_{i_k}^{j_k} b^s$. Найменш можливе k назовемо (слово) довжиною елементу g , та позначимо $l_{G_{p,2}}(g)$. Якщо $s \neq 0$, то $g \notin St(G_{p,2})$. Однак, і в цьому випадку ми зможемо розглянути звуження дії цього елементу на піддерева, розглядаючи звуження для елементу gb^{-s} . Аналогічно, якщо $h = \alpha_{i_1}^{j_1} \cdots \alpha_{i_k}^{j_k} a \in G_{2,p}$, то ми розглядаємо звуження дії елементу h , як для елементу ha .

Таким чином, для кожного $g \in G_{p,2}$, ми можемо розглядати звуження від його звужень. Нехай v – код деякої довільної вершини дерева $T_{p,2}$. Звуження дії елемента g на піддерево з корнем з кодом v позначимо як $g|_v$.

Назвемо глибиною $f(g)$ елемента $g \in G_{p,2}$ таке найменше $m \geq 0$, що

$$g|_v \in \begin{cases} \{\beta, b, e_{G_{p,2}}\}, & m - \text{парне}, \\ \{\alpha, a, e_{G_{2,p}}\}, & m - \text{непарне}, \end{cases}$$

для усіх вершин v рівня m (e_G – нейтральний елемент групи G). Зрозуміло, що $f(\beta) = 0$. Цікаво відмітити, що для довільних $g_1, g_2 \in G_{p,2}$ виконується нерівність $f(g_1 g_2) \leq \max\{f(g_1), f(g_2)\} + 1$. Таким чином, глибина кожного елементу групи $G_{p,2}$ скінченна. За аналогією, будемо розглядати глибину елементів групи $G_{2,p}$, позначаючи функцію глибини також f . Відмітимо, що в групі автоморфізмів дерева виконується рівність $(g_1 g_2)|_v = g_1|_{g_2 \cdot v} g_2|_v$, де $g_2 \cdot v$ – вершина дерева, в яку переходить вершина v під дією автоморфізму g_2 .

1. Нескінченність. Побудуємо нескінченну послідовність елементів групи $G_{p,2}$ з зростаючою глибиною. Таким чином, ми будемо мати попарно різні елементи в цій послідовності, а значить і в нашій групі. Покладемо, $B_0 = \beta_0$, $B_1 = \beta_1$, та $B_k = \beta_0^{B_{k-1}}$, для натуральних $k \geq 2$. Нехай 0_n – послідовність нулів довжини n . Зрозуміло, що $\beta_0|_{0_n}$ дорівнює β , якщо n – парне, та дорівнює α , якщо n – непарне. Крім того, елемент β діє на вершину з кодом 0_n тривіально для будь якого натурального n .

Покажемо по індукції, що $B_n|_{0_n} = b$ і $B_n|_{0_{n-1}} = \beta$ якщо n – парне, та $B_n|_{0_n} = a$,

$B_n|_{0_{n-1}} = \alpha$ якщо n – непарне. Дійсно, $B_1|_0 = \beta_1|_0 = a$, $B_1|_1 = \beta_1|_1 = \alpha$. По індукції, припустимо вірність твердження для усіх $k \leq n$, та доведемо для $k = n$. Нехай n – непарне. Тоді, маємо рівність

$$B_n|_{0_{n-1}} = B_{n-1}^{-1}|_{0_{n-1}} \cdot \beta \cdot B_{n-1}|_{0_{n-1}}.$$

За індукцією, $B_n|_{0_{n-1}} = b^{-1} \cdot \beta \cdot b = \beta_1$. Отже, $B_n|_{0_n} = \beta_1|_0 = a$, $B_n|_{0_{n-1}} = \beta_1|_1 = \alpha$. Якщо n – парне, то $B_n|_{0_{n-1}} = a^{-1} \cdot \alpha \cdot a = \alpha_1$ і $B_n|_{0_n} = \alpha_1|_0 = b$, $B_n|_{0_{n-1}} = \alpha_1|_1 = \beta$. Таким чином, індукція виконується. Із доведення видно, що глибина $f(B_n) \geq n$. Отже, множина $G_{p,2}$ нескінченна.

2. Періодичність. Доведення періодичності здійснюємо за індукцією. База індукції слідує з скінченності порядків елементів β, b, α, a . Нехай, $g = \beta_{i_1}^{j_1} \cdots \beta_{i_n}^{j_n} b^s$, $l_{G_{p,2}}(g) = n$, та вже доведено, що усі елементи довжини $< n$ мають скінченний порядок. Доведемо скінченність порядку елемента g . Якщо $s = 0$, то можна розглянути проекції $g|_x = \delta(x, i_1)^{j_1} \cdots \delta(x, i_n)^{j_n}$, де $\delta(x, i) = \begin{cases} \alpha, & x = i, \\ a, & x \neq i. \end{cases}$ В силу $i_y \neq i_{y+1}$, $y = 1, 2, \dots$, та рівності $|g| = \max_x |g|_x$, маємо $|g| < \infty$, оскільки $l_{G_{p,2}}(g|_x) < n$ для всіх x , $0 \leq x \leq p-1$. За аналогічними міркуваннями, індукцією можна скористуватися і для елемента $h \in St(G_{2,p})$.

Нехай тепер $s \neq 0$. Тоді, розглянемо елемент

$$g^p = \beta_{i_1}^{j_1} \cdots \beta_{i_n}^{j_n} \beta_{i_1+s}^{j_1} \cdots \beta_{i_n+s}^{j_n} \cdots \\ \beta_{i_1+(p-1)s}^{j_1} \cdots \beta_{i_n+(p-1)s}^{j_n},$$

де сума індексів виконується за модулем p . Зазначимо, що для кожного i , множина індексів $\{i, i+s, \dots, i+(p-1)s\}$ співпадає з множиною $\{0, 1, \dots, p-1\}$. Тому, слово $g^p|_x$, $0 \leq x \leq p-1$ до свого можливого скорочення складається із n елементів виду α_i^j і $k \cdot (p-1)$ елементів a . Тобто, число елементів a – парне, і в силу $|a| = 2$, ми маємо $g^p|_x \in St(G_{2,p})$ для усіх x . Зауважимо, що $|g| = p \cdot \max_x |g^p|_x$, та в силу входження $g^p|_x \in St(G_{2,p})$, ми в змозі скористуватися скінченністю порядку елемента $h = g^p|_x$. Як наслідок, $|g| < \infty$.

Розглянемо останній випадок, коли елемент групи $G_{2,p}$ має вигляд $h = \alpha_{i_1}^{j_1} \cdots \alpha_{i_n}^{j_n} a$, $l_{G_{2,p}}(h) = n$. В цьому випадку, маємо

$$h^2 = \alpha_{i_1}^{j_1} \cdots \alpha_{i_n}^{j_n} \alpha_{i_1+1}^{j_1} \cdots \alpha_{i_n+1}^{j_n},$$

де сума в індексах здійснюється по модулю 2. За аналогією випадку в групі $G_{p,2}$, в елемент $h^2|_x$, $0 \leq x \leq 1$, входить (до його можливого скорочення) n елементів виду β_r^i та n елементів b . Тому маємо

$$|h| = 2 \cdot \max_x \{|h^2|_x|\}, \quad l_{G_{p,2}}(h^2|_x) \leq n$$

і для елемента $g_x = h^2|_x \in G_{p,2}$ можна скористатися вище описаними зауваженнями. По індукції, $|g_x| < \infty$, та $|h| < \infty$. Теорему доведено. $\square$

Цікаво відмітити, що 2-породжена група G_{p_1,p_2} , з системою твірних $\langle \alpha, \beta \rangle$, де α і β будуються за аналогією з групами $G_{p,2}$, але для дерев виду T_{p_1,p_2} для непарних простих p_1, p_2 , вже не є періодичною.

Побудовані в роботі групи мають просте означення, але досить нетривіальну природу. Наприклад, постає питання: Чи існує в групі $G_{p,2}$ скінченно-породжені нескінченні 2- або p -підгрупи? Ми вважаємо, що таких груп немає. Зокрема, силівські 2- та p -підгрупи є нескінченно-породженими. Але доведення цих питань є цікавою проблемою.

Наступним питанням є дослідження функції зростання періодів $\pi_{G_{p,2}}$ групи $G_{p,2}$, яка задається як максимум періодів серед елементів обмеженої довжини:

$$\pi_{G_{p,2}}(n) = \max\{|g|, l_S(g) \leq n\},$$

де $l_S(g)$ – довжина елемента g в обраній системі твірних S . Функція зростання періодів має сенс в вузькому класі періодичних скінченно-породжених нескінченних груп, та розглядалася до цього часу для певних p -груп (наприклад, для 2-груп Григорчука, див. [4] та [5]). Тому, цікаво для порівняння розглянути зростання періодів в $\{2, p\}$ -групах.

Відмітимо, що нескінченні, скінченно-породжені групи автоморфізмів дерев виду T_{p_1,p_2} вперше розглядалися в роботах Рожкова (див., наприклад, [6]).

Висновки

В роботі вперше розглядаються нескінченні 2-породжені $\{2, p\}$ -групи. Цікава природа цих груп повинна приглянути увагу до себе. Групи цього класу груп можуть відзначатися новими, притаманними лише їм властивостями. Тому, ці групи надалі потребують більш детального вивчення.

Список використаних джерел

1. Григорчук Р. И. К проблеме Бернсайда о периодических группах / Григорчук Р. И. // Функц. анализ и его прилож. – 1980. – т. 14. – вып. 1. – С. 53–54.
2. Gupta H. Some infinite p -groups / Gupta H., Сидки С. // Алгебра и логика. – 1983. – 22. – № 5. – С. 584–589.
3. Григорчук Р. И. Автоматы, динамические системы и группы / Григорчук Р. И., Некрашевич В. В., Суцанский В. И. // Тр. Матем. Инс-та им. В. А. Стеклова. – 2000. – 231. – С. 134–214.
4. Леонов Ю. Г. Нижняя оценка роста периодов в группах Григорчука / Леонов Ю. Г. // Математичні Студії. Львів. – 1997. – Т. 8. – №. 2. – С. 192–197.
5. Леонов Ю. Г. О точности оценки роста периодов в 2-группах Григорчука / Леонов Ю. Г. // Вопросы Алгебры. Гомель. Беларусь. – 1998. – №. 13. – С. 58–67.
6. Рожков А. В. Условия конечности в группах автоморфизмов деревьев / Рожков А. В. // Дис. докт. физ.-мат. наук. Челябинский гос. ун-т. – 1996.

Надійшла до редколегії 1.11.2010

УДК 512.55

В'ячеслав В. Шви́ров, асистент

Про напівланцюгові справа нетерові зліва напівпервинні кільця

У роботі розглядаються напівланцюгові справа (зліва) нетерові зліва (справа) напівпервинні кільця. Доведено, що такі кільця є скінченним прямим добутком напівланцюгових нетерових спадкових первинних кілець.

Ключові слова: напівланцюгове кільце, нетерове кільце, напівпервинне кільце, спадкове кільце.

E-mail: slavik_asas@mail.ru

Стаття представлена доктором ф.-м. наук, професором Кириченко В. В.

1 Вступ

Структура напівдосконалих нетерових спадкових первинних кілець вивчалася Г. Міхлером у [9]. Напівланцюгові не артинові кільця вперше вивчалися і були описані В.В. Кириченко у [7],[8]. Використовуючи інший підхід до вивчення нетерових напівланцюгових кілець, аналогічні результати о структурі таких кілець в той же час незалежно були отримані і Уорфілдом [11], зокрема, у цих роботах дається повний опис структури напівланцюгових нетерових кілець. У [3] доведено, що нетерове справа напівланцюгове напівпервинне кільце є скінченним прямим добутком нетерових справа і зліва напівланцюгових напівпервинних кілець. Існують приклади напівланцюгових справа нетерових справа напівпервинних кілець, які не є ні напівланцюговими зліва, ні нетеровими зліва (див. [4]). Основним результатом роботи є теорема 1. Аналогічні результати були отримані А.А. Туганбаєвим у [10]. Користуючись результатами В.В. Кириченко [3], [5], [6] вдалося отримати більш коротке доведення.

Теорема 1. Для кільця A наступні умови є еквівалентні

- 1) A – напівланцюгове справа нетерове зліва напівпервинне кільце.
- 2) A – напівланцюгове зліва нетерове справа напівпервинне кільце.
- 3) A – скінченний прямий добуток напівланцюгових нетерових спадкових первинних кілець.

Vjacheslav V. Shvyrov, assistant

On right serial left noetherian semiprime rings

We consider right (left) serial left (right) noetherian semiprime rings. It is proved that such rings are finite direct product of serial noetherian hereditary prime rings.

Key Words: serial ring, noetherian ring, semiprime ring, hereditary rings.

2 Допоміжні визначення і теореми

Означення 2.1. Модуль M називається ланцюговим, якщо ґратка його підмодулів утворює ланцюг, тобто множина всіх підмодулів модулю M є лінійно впорядкована відносно включення.

Означення 2.2. Модуль називається напівланцюговим, якщо він розкладається у скінченну пряму суму ланцюгових модулів.

Означення 2.3. Кільце R називається ланцюговим справа (відповідно зліва) якщо R_R є правим (відповідно ${}_R R$ є лівим) ланцюговим модулем, тобто ґратка правих ідеалів (відповідно лівих ідеалів) є лінійно впорядкована.

Означення 2.4. Кільце R називається напівланцюговим справа (відповідно зліва), якщо R_R (відповідно ${}_R R$) є напівланцюговим модулем. Кільце яке є напівланцюгове справа та зліва називається напівланцюговим.

Означення 2.5. Кільце A називається напівмаксимальним, якщо воно напівдосконале напівпервинне нетерове справа і для будь-якого локального ідемпотенту $e \in A$ eAe – дискретно нормоване кільце, тобто всі $End(P_i)$ – дискретно нормовані кільця.

Означення 2.6. Кільце A називається первинним, якщо 0 є первинний ідеал у A , тобто добуток будь-яких двох ненулевих двосторонніх ідеалів у A не дорівнює нулю.

Означення 2.7. Кільце A називається напівпервинним, якщо 0 є напівпервинний ідеал, тобто A не містить ненулевих нільпотентних ідеалів.

Означення 2.8. Локальна область цілісності O у якій кожен ідеал головний і така, що має єдиний максимальний ідеал називається дискретно нормованим кільцем.

Наступна теорема дає опис напівланцюгових напівпервинних нетерових справа кілець див. [3] теорема 13.5.3.

Теорема 2. Напівланцюгове напівпервинне і нетерове справа кільце може бути розкладено у прямий добуток первинних кілець. Напівланцюгове первинне нетерове справа кільце є нетерове зліва і спадкове з обох боків. У артиновому випадку такі кільця є Моріта еквівалентні тілу і у неартиновому випадку вони Моріта еквівалентні $H_m(O)$, де O – дискретно нормоване кільце. Зворотньо, всі такі кільця є первинні двобічно спадкові і нетерові.

Означення 2.9. Кільце ендоморфізмів нерозкладного проективного модуля над напівдосконалим кільцем називається головним кільцем ендоморфізмів.

У випадку напівдосконалих кілець має місце наступна теорема про декомпозицію таких кілець.

Теорема 3. ([3], теорема 14.4.3.) Напівпервинне напівдосконале кільце є скінченним прямим добутком нерозкладних кілець. Нерозкладне напівпервинне напівдосконале кільце є або артинове просте кільце, або нерозкладне напівпервинне напівдосконале кільце, таке, що всі головні кільця ендоморфізмів є не-артинові.

Теорема 4. ([3], теорема 14.5.1.) Наступні умови еквівалентні для напівдосконалого напівпервинного нетерового справа кільця A :

- (a) кільце A є напівдистрибутивне;
- (b) кільце A є прямим добутком напівпростого артинового кільця і напівмаксимального кільця.

Теорема 5. Кожне напівмаксимальне кільце ізоморфне скінченному прямому добутку первинних кілець наступного виду

$$A = \begin{pmatrix} O & \pi^{\alpha_{12}}O & \dots & \pi^{\alpha_{1n}}O \\ \pi^{\alpha_{21}}O & O & \dots & \pi^{\alpha_{2n}}O \\ \dots & \dots & \dots & \dots \\ \pi^{\alpha_{n1}}O & \pi^{\alpha_{n2}}O & \dots & O \end{pmatrix},$$

де $n \geq 1$, O є дискретно нормованим кільцем з первинним елементом π , і α_{ij} є цілі, такі, що

$$\alpha_{ij} + \alpha_{jk} \geq \alpha_{ik}$$

для всіх i, j, k ($\alpha_{ii} = 0$ для будь-якого i).

Теорема 6. (теорема 3.6.1. див. [5]) Нехай A довільне кільце з ідемпотентом $e^2 = e \in A$. Покладемо $f = 1 - e$, $eAf = X$, $fAe = Y$ і

$$A = \begin{pmatrix} eAe & X \\ Y & fAf \end{pmatrix}$$

відповідна декомпозиція Пірса кільця A . Тоді кільце A є нетеровим справа (артиновим) тоді і тільки тоді, коли кільця eAe і fAf є нетеровими справа (артиновими), X – це скінченнопороджений fAf -модуль і Y – скінченнопороджений eAe -модуль.

Лема 1. Локальне напівпервинне напівланцюгове справа кільце є первинним.

Доведення. Нехай O – локальне кільце з вказаними вище властивостями, тоді, якщо L_1 і L_2 – двобічні ненульові ідеали кільця O , то або L_1 містить L_2 , або L_2 містить L_1 . З рівності $L_1 \cdot L_2 = 0$ (за умови, що $L_2 \subset L_1$) випливає $L_2^2 = 0$, з напівпервинності кільця O випливає, що $L_2 = 0$. Отримано протиріччя, яке доводить, що локальне напівланцюгове справа кільце є первинним кільцем.

Лема 2. (див. [6], лема 1.1) Нехай A локальне нетерове з одного боку кільце з циклічним радикалом $\pi A = A\pi$, причому елемент π не є нільпотентним. Тоді кільце A є дискретно нормованим кільцем.

Наслідок 1. (9.6.6. див. [1]). Напівдосконале напівпервинне нетерове зліва і справа спадкове кільце є напівланцюговим.

3 Доведення теореми 1

Через те, що умова (3) симетрична, достатньо довести $2 \Leftrightarrow 3$. Імплікація $3 \Rightarrow 2$ є очевидною.

Покажемо $2 \Rightarrow 3$. Через те, що напівланцюгові зліва кільця можуть бути розкладені у пряму суму лівих ідеалів, кожен з яких має в точності один максимальний підмодуль, то за

теоремою 6.2.8. ([1]), таке кільце є напівдосконале. За теоремою 2, A є скінченний прямий добуток нерозкладних напівпервинних кілець. Кожне нерозкладне кільце є або просте артинове кільце, або напівпервинне напівдосконале кільце, головні кільця ендоморфізмів якого не артинові.

У першому випадку A ізоморфне $M_n(D)$ і таке кільце є напівланцюговим нетеровим спадковим первинним кільцем.

Нехай

$$A = P_1^{n_1} \oplus P_2^{n_2} \oplus \dots \oplus P_m^{n_m}$$

і

$$1 = e_1 + \dots + e_m$$

відповідна декомпозиція одиниці кільця A у суму попарно ортогональних локальних ідемпотентів.

За умовою, кільце A – нетерове справа. Одже, кільця $e_i A e_i$ за лемою 1 і теоремою 6 ([1]) є локальні напівланцюгові зліва нетерові справа первинні кільця. Радікал кільця $e A e$ є циклічним. За лемою 2, такі кільця є дискретно нормовані. Звідси кільце A є напівмаксимальним. За теоремами 3 та 4 маємо, що кільце A є скінченним добутком первинних напівдосконалих кілець.

Нехай далі кільце A таке, як у теоремі 4. Покажемо, що кільце A нетерове зліва. Нехай

$$1 = e_1 + \dots + e_m$$

– декомпозиція одиниці $1 \in A$ у суму ортогональних локальних ідемпотентів, $A_{ij} = e_i A e_j$ для $i, j = 1, \dots, m$. Позначимо через B_{ij} ($i \neq j$) наступний мінор другого порядку:

$$B_{ij} = \begin{pmatrix} A_{ii} & A_{ij} \\ A_{ji} & A_{jj} \end{pmatrix}$$

Якщо B_{ij} не є приведеним, тоді

$$B_{ij} \simeq M_2(A_{ii})$$

і B_{ij} є нетерове зліва. Якщо B_{ij} є приведене, тоді $A_{ij} a_{ji} \subset A_{ij}$, $\phi_{ji}: A_{ij} \rightarrow A_{ii}$ є мономорфізмом лівих A_{ii} -модулів (для будь-якого ненульового a_{ji} , таким, що $\phi_{ji}(a_{ij}) = a_{ij} a_{ji}$). Якщо A_{ij} не є скінченно породжений, тоді A_{ii} містить не скінченно породжений лівий A_{ii} -підмодуль $A_{ij} a_{ji}$, де $a_{ji} \neq 0$. Отримали протиріччя. Отже, за лемою 13.3.4 (див. [3]), $A_{ij} \simeq A_{ii}$ і B_{ij} є нетеров

зліва, за теоремою 6 (див. [1]). Використовуючи індукцію за числом m і теорему 6, ми отримуємо, що A є нетерове зліва. Отже A є двобічно нетерове.

Через те, що кільце A є напівпервинне первинний радикал $P_r(A) = 0$. Користуючись тим фактом, що сингулярний ідеал міститься у первинному радикалі отримуємо, що сингулярний ідеал (лівий, правий) кільця A , $Sing(A) = 0$. Звідси A є несингулярне (зліва і справа). За твердженням 1.6. (див. [10]), несингулярність еквівалентна напівспадковості (справа і зліва). З того, що A є нетерове справа і зліва випливає, що A є спадкове.

Зважаючи на наслідок 1 (див. [1]), напівдосконале напівпервинне нетерове з обох боків спадкове кільце є напівланцюговим (справа і зліва). Отже, A є напівланцюгове нетерове первинне кільце.

4 Приклади

Наведемо приклади (див. [4]) кілець, що є напівланцюговими справа нетеровими справа, але не є напівланцюговими зліва нетеровими зліва.

1. Нехай Q – поле раціональних чисел і $Q(x)$ – поле раціональних функцій над Q від змінної x . Визначимо мономорфізм

$$\alpha: Q(x) \rightarrow Q(x),$$

вважаючи $\alpha(r(x)) = r(x^2)$. Нехай R складається з правих поліномів від y з коефіцієнтами у $Q(x)$, тобто виразів виду

$$\sum y^i r_i(x),$$

де $r_i(x) \in Q(x)$. Рівність і додавання в R визначаються звичайним чином, множення визначається правилом:

$$r(x)y = y\alpha(r(x)).$$

Легко бачити, R є ланцюговим справа, і, що кожен правий ідеал в R є головним, отже, кільце R – нетерове справа. Але R не є нетеровим зліва.

2. Нехай A – комутативна нетерова область цілісності, що не є полем (наприклад кільце цілих чисел) і нехай F – поле часток для A . Розглянемо кільце R , що утворено матрицями вида

$$\begin{pmatrix} \alpha & \beta \\ 0 & \gamma \end{pmatrix},$$

де $\alpha \in A$, $\beta, \gamma \in F$ з звичайними операціями додавання та множення. Легко бачити, що R – нетерове справа, але не є нетеровим зліва. Такі кільця також є спадковими справа, і не є спадковими зліва. Достатньо розглянути ліві ідеали виду

$$\begin{pmatrix} 0 & M \\ 0 & 0 \end{pmatrix},$$

де M підмодуль A -модуля F , і скористатись тим фактом, що F не є скінченно породженим A -модулем, що суперечить умовам теореми 6.

Клас кілець для яких поняття спадковості справа і спадковості зліва є різними називають кільцями Херстейна-Смолла. Перший приклад таких кілець було сконструйовано І. Капланським. Вище наведені приклади, більш простіші, були пізніше сконструйовані Смоллом. Херстейн [4] використав такі кільця як приклад нетерових справа кілець у яких перетин всіх натуральних степенів радикала Джекобсона не дорівнює нулю.

Висновки. У роботі розглянуті напівланцюгові справа (зліва) і нетерові зліва (справа) напівпервинні кільця. Доведено, що такі кільця є скінченим прямим добутком напівланцюгових нетерових спадкових первинних кілець. Аналогічний результат належить А.А. Туганбаєву. Нами при доведенні були використані результати робіт В.В. Кириченко, А.Г. Завадського, Г. Міхлера, що дозволило скоротити доведення. Також наведені приклади кілець, що є напівланцюговими справа нетеровими справа, але не є напівланцюговими та нетеровими зліва.

Список використаних джерел

1. Gubareni N.M., Kirichenko V.V. Rings and Modules. Czestochowa. // - 2001. - 306 p.

2. Gubareni N.M., Kirichenko V.V. Serial Rings with T-nilpotent Prime Radical. // Algebras and Representations Theory (2006) 9, pp. 147-160.
3. Hazewinkel M., Gubareni N.M., Kirichenko V.V. Algebras, Rings and Modules. Vol.1 // - Kluwer Academic Publishers, 2004.-380p.
4. Herstein I. N., Noncommutative Rings, Cams Mathematical Monographs, No. 15, Mathematical. Association of America, Washington, DC, 1968.
5. Кириченко В. В., О полуцепных наследственных и полунаследственных кольцах, Модули и алгебраические группы, Зап. научн. сем. ЛОМИ, 114, Изд-во "Наука", Ленинград. отд., Л., 1982, 137-147.
6. Завадский А. Г., Кириченко В. В., Полумаксимальные кольца конечного типа, Матем. сб., 103(145):3(7) (1977), 323-345.
7. Kirichenko V.V., Generalized uniserial r-ngs. Preprint IM-75-1, Kiev, 1975.
8. Kirichenko V.V., Generalized uniserial r-ngs. Mat. sb. v.99(141), N4 (1975), p. 559-581.
9. Michler G., Structure of semi-perfect hereditary noetherian rings. // J. Algebra, v. 13, N.3, 1069, p.327-344.
10. Tuganbaev A.A., Modules over serial rings. Mat. Zam. Vol. 65, no 6, pp. 880-892, 1999.
11. Warfield R.B., Serial rings and finitely presented modules. J. Algebra, v. 37 (1975), p.187-222.

Надійшла до редколегії 7.11.10

**ДИФЕРЕНЦІАЛЬНІ
РІВНЯННЯ,
МАТЕМАТИЧНА ФІЗИКА
ТА МЕХАНІКА**

УДК 620.179

Богданов В.Р., к.ф.-м.н., с.н.с., Сулим Г.Т., д.ф.-м.н., проф.

Динамічний розвиток тріщини у компактному зразку за пружно-пластичною моделлю плоского напруженого стану.

З використанням різницевих методів досліджується плоский напружений стан тонкого компактного зразка для визначення в'язкості руйнування (тріщиновитримності) в нестационарній пружно-пластичній постановці з урахуванням підростання тріщини за навантаження, яке прикладене до локалізованої області та змінюється з часом за лінійним законом. Тріщина підростає за умови відсутності максимальних напружень в області вістря тріщини.

Ключові слова: компактний зразок, пластична деформація, ріст тріщини.

*E-mail: vladislav_bogdanov@hotmail.com

Статтю представив д.ф.-м.н. Мелешко В.В.

1. Вступ. У праці [5] було запропоновано для аналізу процесів руйнування застосувати поряд із експериментальними також і розрахункові методи із використанням квазістатичної пружно-пластичної моделі матеріалу. Ці результати було узагальнено у публікаціях [2, 3], де визначальні співвідношення для пружно-пластичного деформування було поєднано із динамічними рівняннями у припущенні, що тріщина є нерухомою. Запропоновані моделі дали можливість істотно підвищити рівень адекватності отриманих теоретичних підходів, однак їхнє застосування можна вважати коректним лише у разі дуже повільного підростання тріщини. В публікації [6] визначається напружено-деформований стан жорсткопластичної криволінійної пластини змінної товщини з довільним отвором при динамічному навантаженні.

У цьому дослідженні методика праці [3] поширена на той випадок, коли тріщина динамічно просувається у пружно-пластичному зразку після його динамічного навантаження.

2. Математичне формулювання задачі.

Розглядається деформування балкового зразка (компактного зразка) у формі прямокутника $\Sigma = L \times B$ ($0 \leq y \leq B$; $-L/2 \leq x \leq L/2$) з пропилом-тріщиною початкової довжини $l = l_0$ уздовж

V. R. Bogdanov*, PhD, G. T. Sulym, Professor.

The Crack Growing in Compact Specimen by Plastic-Elastic Model of Planar Stress State

On the base of developed method of solving problems of planar stress state in non-stationary plastic-elastic model the problem of crack growing was solved using method of finite differences when non-stationary pressure is applied. The non-stationary pressure time dependence is linear and area of that pressure has not changed. The crack is growing when maximal stresses in the area of the top of crack are absent.

Key Words: compact specimen, plastic deformation, crack growing.

відрізка $\{x = 0; 0 \leq y \leq l_0\}$ і контактує з двома нерухомими опорами уздовж $\{L \leq |x| \leq L + a; y = 0\}$. Товщина w зразка вважається настільки малою, щоб можна було використовувати залежності плоского напруженого стану. Зверху на тіло падає абсолютно жорсткий ударник, що контактує уздовж відрізка $\{|x| \leq A; y = B\}$. Його дію на тіло замінимо рівномірно розподіленим в області контакту нормальним напруженням $-P$, що змінюється з часом як лінійна функція $P = p_{01} + p_{02}t$. З огляду на симетрію процесу деформування відносно лінії $x = 0$ далі розглядається лише права частина поперечного перерізу тіла (рис. 1а). Вважатимемо, що матеріал є пружно-пластичним із зміцненням, причому розрахунок полів напружень, деформацій та їхніх приростів, зокрема й приростів інтенсивності пластичних $d\epsilon_i^p$, а також параметра Одквіста $\kappa = \int d\epsilon_i^p$ будемо нижче здійснювати на основі числового розв'язування відповідної динамічної пружно-пластичної задачі. При здійсненні розрахунків використаємо відомі методики дослідження квазістатичної пружно-пластичної [5] моделі, що враховують нестационарність навантаження і застосовують числове інтегрування, зrealізоване

в розрахунку динамічної пружної моделі [10]. При розрахунку динамічних полів напружень і деформацій не враховуємо взаємодії хвильових полів, відбиття від межі тіла і можливої при цьому контактної взаємодії між берегами розрізу. Розглядаються рівняння плоскої динамічної теорії. Компоненти вектора зміщень $\mathbf{u} = (u_x, u_y)$ пов'язані з компонентами тензора деформацій співвідношеннями Коші. Рівняння руху середовища густиною ρ мають вигляд

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} = \rho \frac{\partial^2 u_x}{\partial t^2}, \quad \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} = \rho \frac{\partial^2 u_y}{\partial t^2}.$$

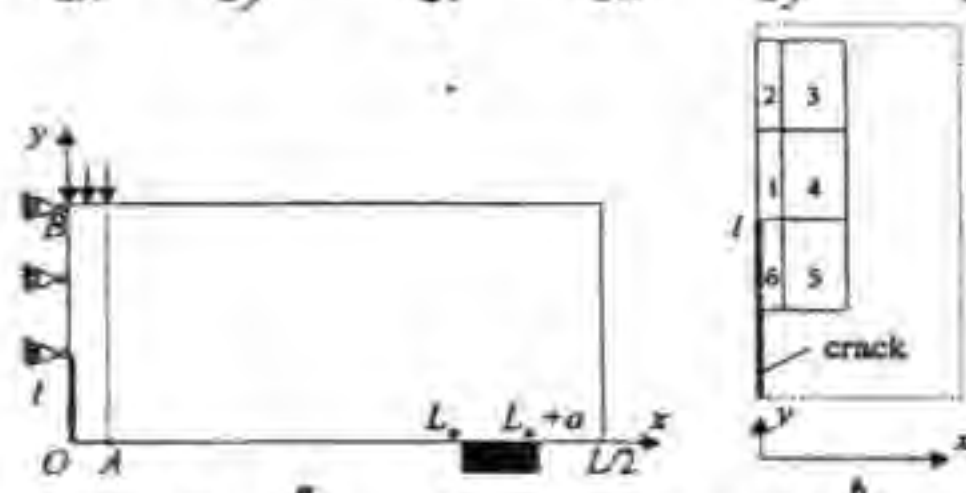


Рис. 1. Геометрична схема задачі (а) і сітка розбиття біля вістря тріщини (б)

Початкові і крайові умови задачі, які враховують зміну довжини тріщини, однак виходять із припущення про незмінність області прикладання реакції, розташування опор, а також визначення опорних реакцій за допомогою методів статистики, запишуться так:

$$\begin{aligned} x=0, \quad 0 < y < l: & \quad \sigma_{xx} = 0, \quad \sigma_{xy} = 0; \\ x=0, \quad l < y < B: & \quad u_x = 0, \quad \sigma_{xy} = 0; \\ x=L/2, \quad 0 < y < B: & \quad \sigma_{xx} = 0, \quad \sigma_{xy} = 0; \\ y=0, \quad 0 < x < L: & \quad \sigma_{yy} = 0, \quad \sigma_{xy} = 0; \\ y=0, \quad L < x < L+a: & \quad u_y = 0, \quad \sigma_{xy} = 0; \\ y=0, \quad L+a < x < L/2: & \quad \sigma_{yy} = 0, \quad \sigma_{xy} = 0; \\ y=B, \quad 0 < x < A: & \quad \sigma_{yy} = -P, \quad \sigma_{xy} = 0; \\ y=B, \quad A < x < L/2: & \quad \sigma_{yy} = 0, \quad \sigma_{xy} = 0, \end{aligned}$$

$$u_{|x=0} = 0, \quad \dot{u}_{|x=0} = 0, \quad l_{|t=0} = l_0. \quad (1)$$

В основу визначальних співвідношень механічної моделі покладено теорію неізотермічного пластичного течіння середовища із зміцненням за умови текучості Губера - Мізеса. Ефектами повзучості і температурним розширенням нехтуємо [8]. Відповідно цієї моделі рівняння зв'язку між напруженнями і деформаціями мають вид [1]:

$$\varepsilon_{ij} = \varepsilon_{ij}^e + \varepsilon_{ij}^p, \quad \varepsilon_{ij}^e = \frac{1}{2G} s_{ij} + K\sigma, \quad d\varepsilon_{ij}^p = s_{ij} d\lambda. \quad (2)$$

тут $s_{ij} = \sigma_{ij} - \delta_{ij}\sigma$ - компоненти дівіатора тензора напружень; δ_{ij} - символ Кронекера; G - модуль зсуву; $K_1 = (1-2\nu)/(3E)$, $K = 3K_1$ - модуль об'ємного стиску, що зв'язує у співвідношенні $\varepsilon = K\sigma$ об'ємне розширення 3ε ; E - модуль пружності; $\sigma = (\sigma_{xx} + \sigma_{yy} + \sigma_{zz})/3$ - середнє напруження; ν - коефіцієнт Пуассона; $d\lambda$ - деяка скалярна функція, що визначається умовою пластичності (формою поверхні навантаження) і квадратично залежить від компонент дівіатора напружень s_{ij} [8].

$$d\lambda = \begin{cases} 0 & (f \equiv \sigma_i^2 - \sigma_s^2(T) < 0), \\ \frac{3d\varepsilon_i^p}{2\sigma_i} & (\|f, df\| = 0) \end{cases}$$

Матеріал зміцнюється з коефіцієнтом зміцнення η_* [5]:

$$\begin{aligned} \sigma_s(T) &= \sigma_{02}(T_0) \left(1 + \kappa(T)/\varepsilon_0\right)^{\eta_*}, \\ \varepsilon_0 &= \sigma_{02}(T_0)/E, \quad T_0 = 20^\circ\text{C}, \end{aligned} \quad (3)$$

де $\sigma_s(T)$ - межа текучості після зміцнення матеріалу за температури T .

3. Схема розв'язування задачі.

Нехай нестационарна взаємодія відбувається в інтервалі часу $t \in [0, t_*]$. Тоді для кожного моменту часу t з цього інтервалу:

$$\begin{aligned} \varepsilon_{|x,y|}^e &= \frac{\sigma_{|x,y|} - \sigma}{2G} + K\sigma, \quad \frac{d\varepsilon_{|x,y|}^p}{dt} = \left(\sigma_{|x,y|} - \sigma\right) \frac{d\lambda}{dt}, \\ \varepsilon_{xx}^e &= -\frac{\nu}{1-\nu} (\varepsilon_{xx}^e + \varepsilon_{yy}^e), \quad \frac{d\varepsilon_{xy}^p}{dt} = \sigma_{xy} \frac{d\lambda}{dt}, \quad \varepsilon_{xy}^e = \frac{\sigma_{xy}}{2G}, \\ \varepsilon_{xx}^p &= -\varepsilon_{xx}^e - \varepsilon_{yy}^p. \end{aligned} \quad (4)$$

Для числового інтегрування за часом використовувався квадратурна формула Грегорі [9] порядку $m_1 = 3$ з коефіцієнтами D_n . Після рівномірної дискретизації за часом з вузлами $t_k = k\Delta t \in [0, t_*]$ ($k = \overline{0, K}$) запишемо:

$$\begin{aligned} \Delta\varepsilon_{xx,k} &= B_1\sigma_{xx,k} + B_2\sigma_{yy,k} - b_{xx}, \quad \Delta\varepsilon_{xy,k} = B_3\sigma_{xy,k} - b_{xy}, \\ \Delta\varepsilon_{yy,k} &= B_2\sigma_{xx,k} + B_1\sigma_{yy,k} - b_{yy}, \quad B_3 = \frac{1}{2G} + D_0\Delta\lambda_k, \\ \Delta\varepsilon_{xx,k} &= B_4(\sigma_{xx,k} + \sigma_{yy,k}) - \Delta\varepsilon_{xx,k} - \Delta\varepsilon_{yy,k} - b_{xx}, \\ B_1 &= \frac{1}{3} \left(K + \frac{1}{G} + 2D_0\Delta\lambda_k \right), \quad B_4 = \frac{1-2\nu}{3(1-\nu)} \left(2K + \frac{1}{2G} \right), \\ B_2 &= \frac{1}{3} \left(K - \frac{1}{2G} - D_0\Delta\lambda_k \right), \quad b_{xx} = B_4(\sigma_{xx,k-1} + \sigma_{yy,k-1}), \\ b_{ij} &= \frac{1}{2G} \sigma_{ij,k-1} + \delta_{ij} \left(K - \frac{1}{2G} \right) \sigma_{k-1} - \end{aligned}$$

$$\sum_{n=1}^{m-1} D_n (\sigma_{ij,k-n} - \delta_{ij} \sigma_{k-n}) \Delta \lambda_{k-n} \quad (i, j = x, y). \quad (5)$$

Функція $\psi = 1/(2G) + \Delta \lambda$, що характеризує умову текучості, з урахуванням (4) дорівнює:

$$\psi = \left\{ \frac{1}{2G} \quad (f < 0), \quad \frac{1}{2G} + \frac{3\Delta \varepsilon_i^p}{2\sigma_i} \quad (f = 0, \quad df = 0) \right\}$$

$$\sigma_i = \frac{1}{\sqrt{2}} \left((\sigma_{xx} - \sigma_{yy})^2 + (\sigma_{xx})^2 + (\sigma_{yy})^2 + 6(\sigma_{xy})^2 \right)^{1/2}$$

$$\Delta \varepsilon_i^p = \frac{\sqrt{2}}{3} \left((\Delta \varepsilon_{xx}^p - \Delta \varepsilon_{yy}^p)^2 + (\Delta \varepsilon_{xx}^p)^2 + (\Delta \varepsilon_{yy}^p)^2 + \right. \quad (6)$$

$$\left. (\Delta \varepsilon_{yy}^p - \Delta \varepsilon_{xx}^p)^2 + 6(\Delta \varepsilon_{xy}^p)^2 \right)^{1/2}, \quad \Delta \varepsilon_{xx}^p = -\Delta \varepsilon_{xx}^p - \Delta \varepsilon_{yy}^p,$$

$$\varepsilon_{[xx,yy]}^e = \frac{1}{2G} \sigma_{[xx,yy]} + \left(K - \frac{1}{2G} \right) \sigma, \quad \varepsilon_{xy}^e = \frac{1}{2G} \sigma_{xy},$$

$$\Delta \varepsilon_{[xx,yy,xy]}^p = \Delta \varepsilon_{[xx,yy,xy]} - \Delta \varepsilon_{[xx,yy,xy]}^e.$$

Для урахування фізичної нелінійності, що міститься в залежностях (6), застосовується метод послідовних наближень, який дає можливість нелінійну задачу звести до послідовності лінійних задач [5]:

$$\psi^{(n+1)} = \left\{ \psi^{(n)} p + \frac{1-p}{2G} \quad (Q_i < -Q); \quad \psi^{(n)} \quad (|Q_i| < Q); \right. \\ \left. \psi^{(n)} \frac{\sigma_i^{(n)}}{\sigma_s(T)} \quad (Q_i > Q) \right\}, \quad Q_i = \sigma_i^{(n)} - \sigma_s(T). \quad (7)$$

де Q – найбільше відхилення інтенсивності напружень від зміщеної межі текучості; стала $0 \leq p \leq 1$.

Розв'язок системи (5) дає вирази для компонент тензора напружень на кожному кроці:

$$\sigma_{xx,k} = A_1 \Delta \varepsilon_{xx,k} + A_2 \Delta \varepsilon_{yy,k} + Y_{xx}, \quad Y_{xx} = A_1 b_{xx} + A_2 b_{yy},$$

$$\sigma_{yy,k} = A_2 \Delta \varepsilon_{xx,k} + A_1 \Delta \varepsilon_{yy,k} + Y_{yy}, \quad Y_{yy} = A_2 b_{xx} + A_1 b_{yy},$$

$$\sigma_{xy,k} = A_3 \Delta \varepsilon_{xy,k} + Y_{xy}, \quad Y_{xy} = A_3 b_{xy}, \quad A_3 = 1/B_3,$$

$$A_1 = B_1 / (B_1^2 - B_2^2), \quad A_2 = -B_2 / (B_1^2 - B_2^2). \quad (8)$$

Інтенсивність напружень і деформацій, що використовуються вище, визначалися для кожної елементарної комірки із числового розв'язку.

Оскільки в механіці руйнування в'язкість руйнування (тріщиновитримність) переважно отримують у квазістатичних експериментах та зіставляють її із граничним значенням коефіцієнту інтенсивності напружень (КІН) K_I , отриманим із лінійно пружного розв'язку, то для опису зміни окремих характеристик у багатьох залежностях в ролі незалежного параметра (змінної) виберемо наближене значення КІН K_I^e (нижче називатимемо його пружним КІН) для багатьох залежностей для пружної задачі трітточкового згину балки з тріщиною [7]:

$$K_I^e = 12F\sqrt{l} \left(1,93 - 3,07(l/B) + 14,53(l/B)^2 - 25,11(l/B)^3 + 25,8(l/B)^4 \right) / \sqrt{Bh}. \quad (9)$$

Більш вірогідним є використання локального критерію руйнування [2]. Для визначення актуальної довжини l тріщини у кожний розглядуваний момент часу використовувалася такий алгоритм. Якщо найбільші значення будь-яких напружень з'являлися не в комірці 1, а в комірці 2, тоді тріщина просувалася на одну комірку вгору. Комірка 2 ставала коміркою 1. Всі напруження перераховувалися для збільшеної тріщини. Так робилося поки всі найбільші напруження не знаходилися в комірці 1 знов.

4. Числова реалізація. Для розрахунків математичної моделі компактного зразка із сталі 15Х2НМФА застосовано метод скінчених різниць зі змінюваним кроком розбиття уздовж осей Ox і Oy (N і M елементів). Крок між точками розбиття був найменшим в околі вершини тріщини і на межах зразка. Характерний розмір [2] комірок в радіусі 1-2 мм від вершини тріщини дорівнює середньому розміру зерна випробуваного металу (0,05 мм). Використання методу скінчених різниць обґрунтовується в [4], причому забезпечується точність розрахунків з похибкою не більше ніж $O((\Delta x)^2 + (\Delta y)^2 + (\Delta l)^2)$.

Для порівнювання з [5] розрахунки проводилися у припущенні відсутності пластичних деформацій ε_{xx}^p . Результати розрахунків середніх напружень у комірках поблизу вістря тріщини за відносно невеликих навантажень, коли у дискретизованій задачі пластичні деформації відсутні і розв'язок задачі є суто лінійно пружним, зіставлялися із розрахованими для центру комірки на основі класичних одночленних асимптотичних залежностей [7, п. 1.2] з використанням виразу (9) для КІН пружного розв'язку. На рис. 2 і 3 відображено результати обчислення величин за таких значень параметрів: $\eta_s = 0,05$; $L = 60$ мм; $B = 10$ мм; $l_0 = 3$ мм; $\Delta t = 5 \cdot 10^{-4}$ с; $A = 2,5$ мм; $p_{01} = 8$ МПа; $p_{02} = 10$ МПа; $M = 60$; $N = 77$, $T = 50^\circ$.

Графіки розрахованої залежності напружень біля вістря тріщини (комірка 1, рис. 1б) (рис. 2) свідчать, що з розвитком процесу деформування зразка в разі перевищення пружним КІН K_I^e рівня $K_I^e = K_I^e = 90,5$ МПа $\sqrt{м}$, напруження у цій

точці монотонний характер збільшення змінюють на коливний.

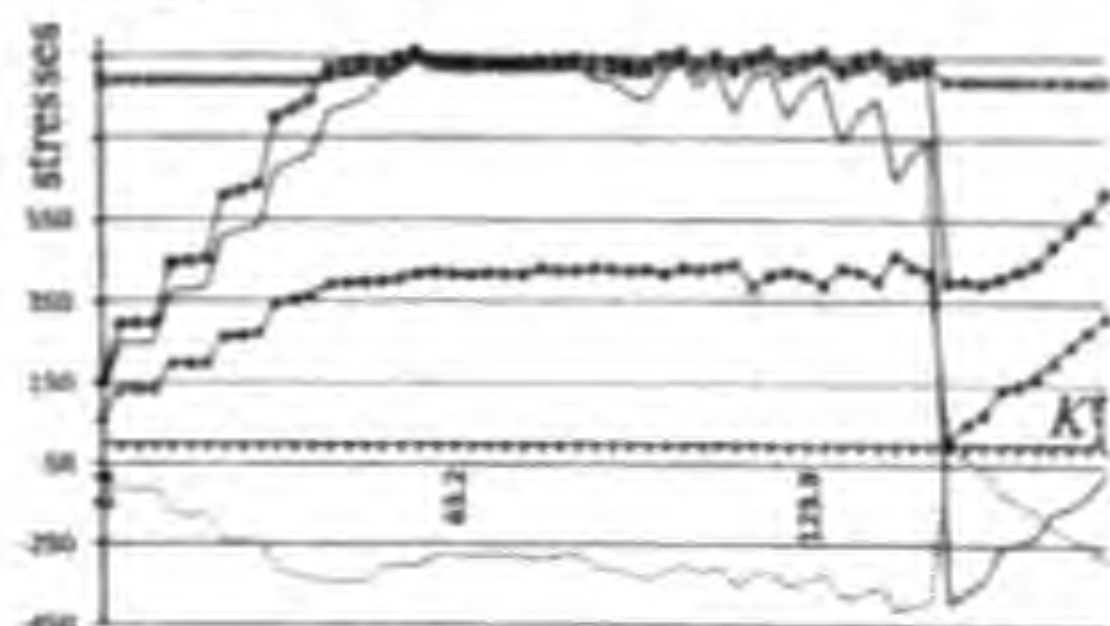


Рис. 2. Залежність напруження у комірці I на продовженні осі тріщини від K_I^*

Суцільна, суцільна з трикутником, пунктирна, суцільна з хрестиком і суцільна з кружечком лінії стосуються відповідно напружень σ_{xx} , σ_{yy} , σ_{zz} , напруження текучості σ_s та інтенсивність напружень σ_i . Обчислення виявили (рис. 3а), що тріщина починала збільшуватися в разі перевищення пружним КІН рівня $K_{Ic}^* = K_{Ic}^* = 51 \text{ МПа}\sqrt{\text{м}}$. Після досягнення пружним КІН значення $K_I^* = K_{Ic}^* = 148,2 \text{ МПа}\sqrt{\text{м}}$ параметр Одквіста (рис. 3б) в комірці I зменшується до нуля, а довжина тріщини сягає $l = 8,35 \text{ мм}$.

Список використаних джерел

1. Аркулис Г.Э., Дорогобид В.Г. Теория пластичности / Г.Э. Аркулис, В.Г. Дорогобид – М.: Металлургия, 1987. – 352 с.
2. Богданов В.Р. Визначення в'язкості руйнування матеріалу на основі чисельного моделювання плоского напруженого стану / В.Р. Богданов // Вісник Київського нац. університету. Серія фіз.-мат. науки. – 2008. – № 3. – С. 51–56.
3. Богданов В.Р. Тривимірна динамічна задача концентрації пластичних деформацій і напружень біля вершини тріщини / В.Р. Богданов // Вісник Київського нац. університету. Серія фіз.-мат. науки. – 2009. – № 2. – С. 51–56.
4. Зюкина Е.Л. Консервативные разностные схемы на неравномерных сетках для двумерного волнового уравнения / Е.Л. Зюкина // Труды математич. центра имени Н.И. Лобачевского, – Казань, – т.26. – 2004 – С. 151–160.

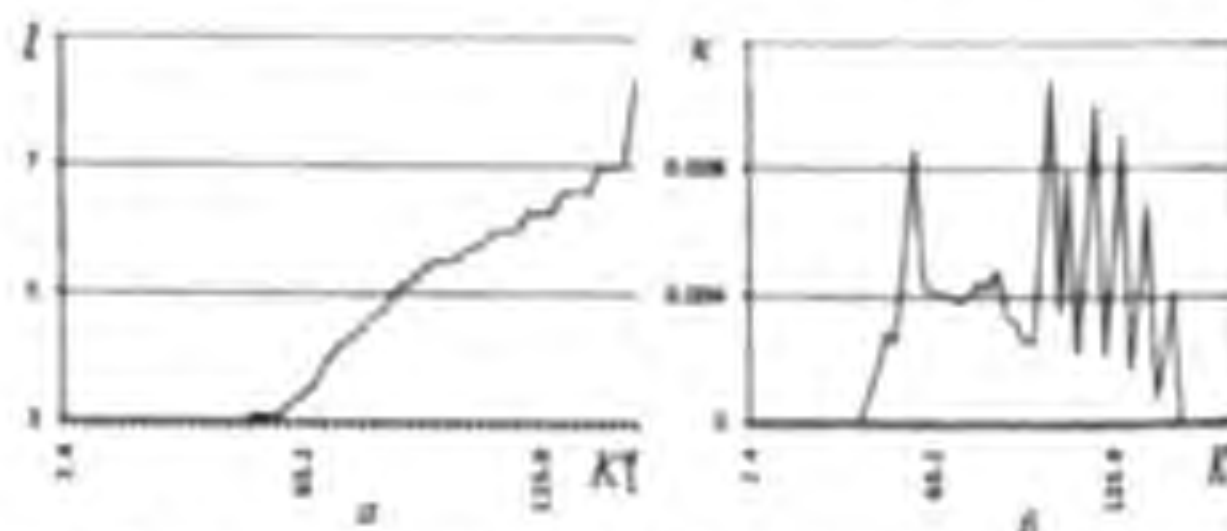


Рис. 3. Залежність від КІН K_I^* а). довжини тріщини l ; б). параметра Одквіста k у комірці I.

Це означає, що за такої довжини тріщини перед її вістрям матеріал не є пластично деформованим і подальше доломлювання зразка відбувається практично пружно.

Висновок. Розв'язування задачі плоского напруженого стану для компактного зразка для визначення в'язкості руйнування на тріщинковий згин у динамічному пружно-пластичному формулюванні з урахуванням підростання тріщини за умови відсутності максимальних напружень на вістрі тріщини дає можливість набагато точніше визначити поля пластичних деформацій і напружень, аніж при розв'язуванні квазістатичної пружно-пластичної задачі плоского напруженого стану, а також дає можливість адекватно моделювати процес розповсюдження тріщини.

5. Махненко В.И. Расчетные методы исследования кинетики сварочных напряжений и деформаций / В.И. Махненко – Киев: "Наукова думка", 1976. – 320 с.
6. Немировский Ю.В., Романова Т.П. Динамика жесткопластической криволинейной пластины переменной толщины с произвольным отверстием / Ю.В. Немировский, Т.П. Романова // Прикл. механика – 2010. – 46, №3. – С. 70 – 76.
7. Саврук М.П. Механика разрушения и прочность материалов / М.П. Саврук // Т.2. Коэффициенты интенсивности напряжений в телах с трещинами. – Киев: "Наукова думка", 1988. – 620 с.
8. Сборник: Теория пластичности. – М.: ИЛ, 1948. – 460 с.
9. Хемминг Р.В. Численные методы / Р.В. Хемминг // – М.: Наука, 1972. – 399 с.
10. Kubenko V.D., Bogdanov V.R. Planar problem of the impact of a shell on an elastic half-space / V.D. Kubenko, V.R. Bogdanov // International Applied Mechanics. – Vol. 31. – № 6. – 1995. – P. 483–490.

Надійшла до редколегії 15.09.10

УДК 539.3

Олександр Я. Григоренко*, д.ф.-м.н., професор
Тетяна Л. Єфімова, к.ф.-м.н., старший науковий
співробітник
Інна В. Власова**, аспірант

Власні коливання прямокутних анізотропних пластин змінної товщини

Досліджуються вільні коливання прямокутних
анізотропних пластин змінної товщини при
різних граничних умовах. Для розрахунку частот
застосовано чисельно-аналітичний підхід, який
базується на застосуванні сплайн-апроксимації
та методу дискретної ортогоналізації разом з
методом покрокового пошуку.

Ключові слова: анізотропна прямокутна
пластина змінної товщини, вільні коливання,
сплайн-апроксимація.

Aleksandr Ya. Grigorenko, Dr Sci. (Phys.-Math.),
Professor
Tatiana L. Efimova, PhD (Phys.-Math.), senior
researcher
Inna V. Vlasova, PhD student

Natural vibrations of rectangular anisotropic plats of variable thickness

An investigation of the problem on natural
vibrations of anisotropic rectangular plates of
variable thickness under complicated boundary
conditions is carried out on the base of classic
theory. The problems are solved by the spline-
approximation and collocation by the steady-state
numerical method of discrete orthogonalization
with incremental search.

Key Words: anisotropic rectangular plates, natural
vibrations, spline-approximation.

*E-mail: ayagrigorenko@yandex.ru

**E-mail: inna-2011@hotmail.com

Статтю представив член редколегії д.ф.-м.н., професор Мелешко В.В.

Вступ

Пластини змінної товщини з анізотропних
матеріалів знаходять широке застосування в
конструкціях різного типу [1, 3], дослідження їх
динамічних характеристик, спектра власних
частот і форм, має велике значення як для
подальших фундаментальних досліджень, так і
вирішення технічних проблем. У випадку вільних
коливань ортотропних прямокутних пластин
постійної товщини з шарнірно опертими двома
протилежними сторонами можна побудувати
розв'язання в замкненому вигляді. Однак, в
інших випадках граничних умов, зміні товщини в
двох напрямках та при повороті осей ортотропії
відносно координатних осей отримати
аналітичне розв'язання неможливо. В літературі
наведені різні підходи [1, 4] до знаходження
частот власних коливань прямокутних
ізотропних та ортотропних (з осями ортотропії,
що співпадають з координатними осями) пластин
постійної товщини з різними граничними
умовами. Однак пластини змінної товщини
майже не вивчалися.

В даній статті розглядаються вільні коливання
анізотропних (з однією площиною пружної
симетрії) прямокутних пластин в рамках теорії

Кірхгофа-Лява. Для розв'язання поставленої
задачі використовується чисельно-аналітичний
підхід, який базується на зведенні двовимірної
крайової задачі до системи звичайних
диференціальних рівнянь з допомогою
застосування методів сплайн-апроксимації та
колокації в одному з координатних напрямків.
Отримана одновимірна крайова задача на власні
значення розв'язується стійким методом
дискретної ортогоналізації разом з методом
покрокового пошуку.

Постановка задачі

Розглянемо в декартовій системі координат
прямокутну пластину $0 \leq x \leq a$, $0 \leq y \leq b$,
 $-\frac{h}{2} \leq z \leq \frac{h}{2}$ змінної в двох координатних
напрямках товщини $h(x, y)$ (координатна
площина xOy є серединною площиною
пластини). Пластина є анізотропною з однією
площиною пружної симетрії.

Задача о вільних коливаннях такої пластини
формулюється в рамках теорії Кірхгофа-Лява.
Рівняння усталених коливань (множник $e^{i\omega t}$ тут і
надалі опущений) записуються у вигляді

$$\frac{\partial Q_x}{\partial x} + \frac{\partial Q_y}{\partial y} + \rho h \omega^2 w = 0, \quad (1)$$

де w – прогин пластини, ω – кругова частота власних коливань, ρ – густина матеріалу. Для моментів M_x , M_y , M_{xy} і перерізуючих зусиль Q_x та Q_y мають місце співвідношення:

$$Q_x = \frac{\partial M_x}{\partial x} + \frac{\partial M_{xy}}{\partial y};$$

$$Q_y = \frac{\partial M_y}{\partial y} + \frac{\partial M_{xy}}{\partial x};$$

$$M_x = - \left(D_{11} \frac{\partial^2 w}{\partial x^2} + D_{12} \frac{\partial^2 w}{\partial y^2} + 2D_{16} \frac{\partial^2 w}{\partial x \partial y} \right); \quad (2)$$

$$M_y = - \left(D_{12} \frac{\partial^2 w}{\partial x^2} + D_{22} \frac{\partial^2 w}{\partial y^2} + 2D_{26} \frac{\partial^2 w}{\partial x \partial y} \right);$$

$$M_{xy} = - \left(D_{16} \frac{\partial^2 w}{\partial x^2} + D_{26} \frac{\partial^2 w}{\partial y^2} + 2D_{66} \frac{\partial^2 w}{\partial x \partial y} \right),$$

в яких жорсткісні характеристики пластини D_{ij} визначаються за формулами:

$$D_{ij} = \frac{B_{ij} h^3 (x, y)}{12}, \quad (3)$$

$$B_{11} = \frac{a_{22}a_{66} - a_{26}^2}{\Delta}, \quad B_{22} = \frac{a_{11}a_{66} - a_{16}^2}{\Delta},$$

$$B_{66} = \frac{a_{11}a_{22} - a_{12}^2}{\Delta}, \quad B_{12} = \frac{a_{11}a_{66} - a_{16}a_{26}}{\Delta},$$

$$B_{16} = \frac{a_{12}a_{26} - a_{22}a_{16}}{\Delta}, \quad B_{26} = \frac{a_{12}a_{16} - a_{11}a_{26}}{\Delta},$$

$$\Delta = (a_{11}a_{22} - a_{12}^2)a_{66} - 2a_{12}a_{16}a_{26} - a_{11}a_{26}^2 - a_{22}a_{16}^2,$$

a_{ij} – коефіцієнти матриці податливості матеріалу з однією площиною пружної симетрії.

Якщо матеріал пластини ортотропний, але осі ортотропії повернуті відносно координатних осей на деякий кут φ , то матеріал в новій системі координат має одну площину пружної симетрії, і його жорсткісні коефіцієнти B'_{ij} можна отримати з використанням відповідних

жорсткісних коефіцієнтів в системі координат, пов'язаний з осями ортотропії, за формулами [2]:

$$B'_{11} = B_{11} \cos^4 \varphi + 2(B_{12} + 2B_{66} \sin^2 \varphi \cos^2 \varphi) + B_{22} \sin^4 \varphi, \quad (4)$$

$$B'_{22} = B_{22} \cos^4 \varphi + 2(B_{12} + 2B_{66} \sin^2 \varphi \cos^2 \varphi) + B_{11} \sin^4 \varphi,$$

$$B'_{12} = B_{12} + (B_{11} + B_{22} - 2(B_{12} + 2B_{66})) \sin^2 \varphi \cos^2 \varphi,$$

$$B'_{66} = B_{66} + (B_{11} + B_{22} - 2(B_{12} + 2B_{66})) \sin^2 \varphi \cos^2 \varphi,$$

$$B'_{16} = 0.5 [B_{22} \sin^2 \varphi - B_{11} \cos^2 \varphi + (B_{12} + 2B_{66}) \cos 2\varphi] \sin 2\varphi,$$

$$B'_{26} = 0.5 [B_{22} \cos^2 \varphi - B_{11} \sin^2 \varphi + (B_{12} + 2B_{66}) \cos 2\varphi] \sin 2\varphi.$$

З рівнянь (1), (2) отримаємо одне рівняння відносно прогину w , де вигляд оператора L має досить громіздкий, але якісний вигляд:

$$L(w) + \rho h \omega^2 w = 0. \quad (5)$$

На краях пластини задаються граничні умови, в термінах прогину. Для сторін пластини $y = 0$, $y = b$ граничні умови можуть бути такі:

1) контур жорстко закріплений при $y = 0$, $y = b$

$$w = 0, \quad \frac{\partial w}{\partial y} = 0; \quad (6)$$

2) контур шарнірно опертий при $y = 0$

$$w = 0, \quad \frac{\partial^2 w}{\partial y^2} = 0. \quad (7)$$

Аналогічні граничні умови задаються при $x = 0$ та $x = a$.

Рівняння (5) разом з відповідними граничними умовами на краях пластини утворюють двовимірну крайову задачу на власні значення відносно частоти ω .

Метод розв'язання

Для розв'язування двовимірної задачі на власні значення застосуємо метод сплайн-апроксимації. Прогин подамо у вигляді

$$w(x, y) = \sum_{i=0}^N w_i(x) \varphi_i(y). \quad (8)$$

У виразі (8) $w_i(x)$ – невідомі функції, $\varphi_i(y)$ ($i = 0, \dots, N$) – лінійні комбінації B -сплайнів п'ятого степеня на рівномірній сітці, що задовольняють задані граничні умови на краях.

Підставляючи вирази (8) у розв'язувальне рівняння (5) і граничні умови, вимагаємо, щоб рівняння точно виконувалося в $\bar{N} = N + 1$ точці колокації $\xi_k \in [0, L]$, $k = \overline{0, \bar{N}}$. Вибір точок колокації детально описано в роботі [1]. Як результат отримуємо одновимірну крайову задачу, яку можна подати у вигляді

$$\begin{aligned} \frac{d\bar{Y}}{dx} &= A(x, \omega) \bar{Y}, \\ B_1 \bar{Y} &= \bar{0} \text{ для } x = 0, \\ B_2 \bar{Y} &= \bar{0} \text{ для } x = a, \end{aligned} \quad (9)$$

де $\bar{Y} = \{w, w', w'', w'''\}$, $\bar{w} = \{w_0, w_1, \dots, w_N\}$, $\bar{w}' = \{w'_0, w'_1, \dots, w'_N\}$, $\bar{w}'' = \{w''_0, w''_1, \dots, w''_N\}$, $\bar{w}''' = \{w'''_0, w'''_1, \dots, w'''_N\}$, A – квадратна матриця порядку $4(N+1) \times 4(N+1)$, B_1 і B_2 – прямокутні матриці граничних умов порядку $2(N+1) \times 4(N+1)$. Одновимірну крайову задачу (9) на власні значення розв'язується стійким чисельним методом дискретної ортогоналізації разом з застосуванням метода покрокового пошуку [1, 3].

Результати розрахунків

Для оцінки точності та достовірності результатів розрахунків по запропонованій методиці порівнювалися безрозмірні частоти $\bar{\omega} = \frac{a^2}{\pi^2} \omega \sqrt{\frac{\rho h}{D}}$ вільних коливань квадратних

ізотропних шарнірно опертих пластин постійної товщини [5, 6], знайдених за допомогою методу сплайн-колокації (МСК) для різної кількості точок колокації $\bar{N}$ та аналітично (табл.1). Для вказаних пластин вихідне рівняння спрощується до виду

$$\frac{\partial^4 w}{\partial x^4} + 2 \frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} + \rho h \frac{\omega^2}{D} w = 0. \quad (10)$$

В даному випадку можливе представлення прогину у вигляді

$$w = W \sin \frac{k\pi x}{a} \sin \frac{m\pi y}{a}, \quad (11)$$

яке приводить до формули для обчислення безрозмірних частот

$$\bar{\omega} = k^2 + m^2. \quad (12)$$

Таблиця 1

$\bar{\omega}$	Аналітичний розв'язок	МСК $\bar{N}=10$	МСК $\bar{N}=20$	МСК $\bar{N}=30$
$\bar{\omega}_1$	2.0	1.96	1.98	1.99
$\bar{\omega}_2$	5.0	4.96	4.98	4.99
$\bar{\omega}_3$	5.0	5.25	5.15	5.03
$\bar{\omega}_4$	8.0	8.21	8.13	8.04
$\bar{\omega}_5$	10.0	10.13	10.09	10.05

З аналізу частот, вміщених в таблицю, можна зробити висновок про можливість застосування даної методики на вказаному класі задач.

Розв'язувалася задача про власні коливання ортотропної квадратної пластини зі стороною a постійної товщини $h/a = 0.1$ з осями ортотропії, повернутими відносно координатних осей на деякий кут φ . Механічні параметри матеріалу: $E_1 = 4.76E_0$, $E_2 = 2.07E_0$, $G = 0.532E_0$, $E_0 = 10^4 \text{ ГПа}$, $\nu_1 = 0.149$, $\nu_2 = 0.0647$, $\rho = 7800 \text{ кг/м}^3$ (E_1, E_2 – модулі пружності в напрямках x, y відповідно, G – модуль зсуву в площині xy , ν_1, ν_2 – коефіцієнти Пуассона). Розрахунки частот проводилися для різних значень кута повороту осей ортотропії φ у випадках різних граничних умов на краях. Перші три безрозмірні частоти $\bar{\omega} = \omega a^2 \sqrt{\frac{\rho_0 h}{D_0}}$

($D_0 = E_0 h_0^3$, $\rho_0 = 1 \text{ г/см}^3$, $h_0 = 1$) наведені в табл.2 для випадків: I – всі торці пластини шарнірно оперті, II – всі торці пластини жорстко закріплені.

Таблиця 2

Тип граничних умов	$\bar{\omega}$	$\bar{\varphi} = 0$	$\bar{\varphi} = \pi/6$	$\bar{\varphi} = \pi/4$
I	$\bar{\omega}_1$	0.03563	0.03563	0.03563
	$\bar{\omega}_2$	0.11368	0.11368	0.11368
	$\bar{\omega}_3$	0.29836	0.29836	0.29836
II	$\bar{\omega}_1$	2.0291	1.9734	1.9532
	$\bar{\omega}_2$	3.5664	3.6652	3.7336
	$\bar{\omega}_3$	4.7190	4.3201	4.1582

З наведених результатів можна зробити висновок, що при зростанні кута повороту осей ортотропії φ від 0 до $\pi/4$ перша та третя

частоти спадають, а друга зростає як для випадку жорсткого закріплення всіх торців пластини, так і для шарнірно опертих торців.

Досліджувалися власні коливання ортотропної жорстко закріпленої по торцях квадратної пластини зі стороною a змінної за законом $h = (\alpha(6x^2 - 6x + 1) + 1)h_1$, $h_1 = 0.1$ товщини з осями ортотропії, повернутими відносно координатних осей на кут φ . Механічні параметри матеріалу вказані вище. Результати обчислення першої безрозмірної частоти

$$\bar{\omega}_1 = \omega_1 a^2 \sqrt{\frac{\rho_0 h_0}{D_0}} \quad (D_0 = E_0 h_0^3, \rho_0 = 1 \text{ т/м}^3, h_0 = 1)$$

для різних значень параметра α та різних значень кута φ наведено в табл. 3.

Таблиця 3

α	$\bar{\omega}$	$\bar{\varphi} = 0$	$\bar{\varphi} = \pi/6$	$\bar{\varphi} = \pi/4$
0	$\bar{\omega}_1$	2.0291	1.9734	1.9532
0.1	$\bar{\omega}_1$	2.0808	2.0124	1.9952
0.2	$\bar{\omega}_1$	2.1324	2.0508	2.0167

При збільшенні значень параметра α перша власна частота пластини зростає. При фіксованому значенні параметра α і при збільшенні кута φ – перша частота спадає.

Висновки

1. Розроблено чисельно-аналітичну методику розрахунку частот вільних коливань ортотропних прямокутних пластин з анізотропного матеріалу (з однією площиною

пружних властивостей) з застосуванням метода сплайн-апроксимації, колокації, метода дискретної ортогоналізації та покрокового пошуку.

2. Досліджено поведінку власних частот ортотропної квадратної пластини при збільшенні значень кута φ повороту осей ортотропії відносно координатних осей у випадках різних граничних умов на торцях (шарнірно оперті та жорстко закріплені всі торці пластини). При зростанні кута повороту осей ортотропії φ від 0 до $\pi/4$ перша та третя частоти спадають, а друга зростає як для розглянутих типів граничних умов.

3. Досліджено поведінку частот ортотропної з кутом повороту осей ортотропії φ відносно координатних осей квадратної жорстко закріпленої пластини змінної товщини (квадратичний закон зміни $h = (\alpha(6x^2 - 6x + 1) + 1)h_1$). При різних значеннях параметра α та при різних значеннях кута φ . При збільшенні значень параметра α перша власна частота пластини зростає. При фіксованому значенні параметра α при збільшенні кута φ перша частота спадає.

Список використаних джерел

1. Григоренко А.Я., Ефимова Т.Л. Применение метода сплайн-аппроксимации для решения задач о свободных колебаниях прямоугольных пластин переменной толщины // Прикл. механика. 2005. – 41, №10. – С. 90–99.

2. Лехницкий С.Г. Анизотропные пластинки. – М.:Гос. изд-во технико-теорет. литературы, 1957. – 464с.

3. Григоренко О.Я., Ефимова Т.Л., Пузирьов С.В. Дослідження вільних коливань прямокутних ортотропних пластин лінійно змінної товщини // Математичні методи та фізико-механічні поля. – 2006. – 49, №3. – С. 153–161.

4. Курпа Л.В., Чистилина А.В. Исследование собственных колебаний многослойных пологих оболочек и пластин сложной формы в плане // Пробл. прочности. – 2003. – №2. – С. 112–123.

5. Chopra I., Durvasula S. Vibration of simply- supported trapezoidal plates. Part 1. Symmetric trapezoids // J. Sound and Vibr.–1971. – 19, No. 4. – P. 379–392.

6. Chopra I., Durvasula S. Vibration of simply- supported trapezoidal plates. Part 2. Unsymmetric trapezoids // J. Sound and Vibr.–1972. – 20, No. 2. – P. 125–134.

Надійшла до редколегії 11.02.11

УДК 539.3

Олександр Л. Кіпніс*, студент

Згин пружної клиноподібної пластини зосередженою силою

В роботі з використанням методу інтегрального перетворення Мелліна побудовані точні розв'язки задач про згин тонкої пружної клиноподібної пластини зосередженою силою для різних способів закріплення її сторін. Визначено прогин, згинаючі і крутильні моменти та поперечні зусилля. Досліджено локальну поведінку та встановлено характер особливостей характеристик біля вершини.

Ключові слова: клиноподібна пластинка, прогин, метод інтегрального перетворення Мелліна, кутова точка.

* E-mail: a.l.kipnis@gmail.com

Статтю представив член редколегії д.ф.-м.н., професор Мелешко В.В.

1. Вступ

Задачі про згин тонких ізотропних пластин різноманітної форми нормальними навантаженнями при різних умовах закріплення контуру належать до основних задач теорії пластин [1]. Суттєвий інтерес викликає питання про локальну поведінку низки силових характеристик (згинаючі моменти та поперечні зусилля) в класичній моделі Кірхгофа в околі точки зміни типу граничних умов на гладкому контурі [2,3] або точки зламу контуру [4].

Для типового випадку круглої пластини підхід, що базується на використанні методу інтегральних рівнянь [5,6], після складних викладок авторів робіт [2,3] привів до певних висновків відносно локальної поведінки поперечного зусилля в околі точки зміни типу умов. Для випадку шарнірно опертої та вільної межі [2] особливість мала вигляд $\ln x / \sqrt{x}$ при $x \rightarrow 0$. Такий незвичайний тип поведінки стимулював інтерес до розгляду більш простих задач про згин клиноподібної пластини з різними, але однотипними умовами на її сторонах. Задачі допускають точний аналітичний розв'язок, викладений для ряду частинних випадків у відомому задачнику [7]. Частинний випадок клину, розгорнутого у півплощину, і визначить локальну поведінку величин для випадку круглої пластини.

Alexander L. Kipnis*, student

Deflection of elastic wedge-shaped plate by a concentrated force

The problem on the deflection of elastic wedge-shaped plate by a concentrated force is considered. The different cases of fixing of the sides are studied. An exact solution of the corresponding boundary value problem is constructed by Mellin's integral transform method. The deflection, the bending and torsion moments and the shearing forces are determined. The local behavior of these characteristics near the corner point is investigated.

Key words: wedge-shaped plate, deflection, Mellin's integral transform method, corner point..

2. Постановка задачі

Тонка ізотропна пружна пластинка у вигляді клину з кутом розхилу α займає область $0 \leq r < \infty$, $0 \leq \theta \leq \alpha$ в полярних координатах (r, θ) і навантажена в точці (r_0, θ_0) зосередженою силою P . В рамках класичної теорії пластин Кірхгофа прогин пластини $w(r, \theta)$ визначається із неоднорідного бігармонічного рівняння

$$D \Delta \Delta w = \frac{P}{r} \delta(r - r_0) \delta(\theta - \theta_0), \quad (1)$$

де $\Delta = \frac{\partial^2}{\partial r^2} + \frac{1}{r} \frac{\partial}{\partial r} + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2}$ – двовимірний

оператор Лапласа в полярних координатах, D – стала циліндрична жорсткість пластини.

Відомі вирази [1] для згинаючого моменту M_θ і узагальненої перерізуючої сили V_θ

$$M_\theta = -D \left(\nu \frac{\partial^2 w}{\partial r^2} + \frac{1}{r} \frac{\partial w}{\partial r} + \frac{1}{r^2} \frac{\partial^2 w}{\partial \theta^2} \right), \quad V_\theta = Q_\theta - \frac{\partial M_{r\theta}}{\partial r} \\ Q_\theta = -D \frac{1}{r} \frac{\partial}{\partial \theta} (\Delta w), \quad M_{r\theta} = D(1 - \nu) \frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial w}{\partial \theta} \right) \quad (2)$$

(ν – коефіцієнт Пуассона) дають можливість сформулювати граничні умови для рівняння (1) в термінах прогину w .

У випадку жорстко защемленої сторони $\theta = 0$ і вільної від навантаження сторони $\theta = \alpha$ граничні умови мають вигляд:

$$w|_{\theta=0}=0, \quad \frac{\partial w}{\partial \theta}|_{\theta=0}=0, \quad M_\theta|_{\theta=\alpha}=0, \quad V_\theta|_{\theta=\alpha}=0. \quad (3)$$

Якщо сторона $\theta=0$ жорстко защемлена, а сторона $\theta=\alpha$ вільно оперта

$$w|_{\theta=0}=0, \quad \frac{\partial w}{\partial \theta}|_{\theta=0}=0, \quad w_\theta|_{\theta=\alpha}=0, \quad M_\theta|_{\theta=\alpha}=0. \quad (4)$$

Для вільної від навантаження сторони $\theta=0$ і вільно опертої сторони $\theta=\alpha$

$$M_\theta|_{\theta=0}=0, \quad V_\theta|_{\theta=0}=0, \quad w|_{\theta=\alpha}=0, \quad M_\theta|_{\theta=\alpha}=0. \quad (5)$$

Ставиться задача визначення прогину w , згинаючих моментів M_r і M_θ , крутильного моменту $M_{r\theta}$, поперечних зусиль Q_r , Q_θ та дослідження поведінки цих характеристик біля вершини клиноподібної пластини.

3. Побудова розв'язку

Для побудови розв'язку задачі використовується інтегральне перетворення Мелліна [8, 9]. До розгляду вводиться функція перетвореного прогину

$$\bar{w}(p, \theta) = \int_0^\infty w(r, \theta) r^{p-2} dr, \quad (6)$$

де $p = \sigma + i\tau$ ($-\sigma_1 < \sigma < 0$) – комплексний параметр. Тоді рівняння (1) з врахуванням затухання прогину в нулі прийме вигляд:

$$\frac{d^4 \bar{w}}{d\theta^4} + [(p-1)^2 + (p+1)^2] \frac{d^2 \bar{w}}{d\theta^2} + (p-1)^2 (p+1)^2 \bar{w} = \frac{Pr_0^{p+1}}{D} \delta(\theta - \theta_0). \quad (7)$$

Розв'язок звичайного неоднорідного диференціального рівняння (7) знаходиться методом варіації довільних сталих. Виконавши нульові граничні умови, після використання формули обернення отримаємо вираз для прогину:

$$w(r, \theta) = \frac{Pr_0 r}{8D\pi i} \int_{-\sigma-i\infty}^{\sigma+i\infty} \frac{\Omega_{c-f}(p, \theta)}{p(p^2-1)\Delta_{c-f}(p)} \left(\frac{r}{r_0}\right)^{-p} dp, \quad (8)$$

де

$$\Delta_{c-f}(p) = \nu^2 + 2\nu + 5 + (1-\nu)(3+\nu) \times \left[\cos(2p\alpha) - \frac{2(1-\nu)}{3+\nu} p^2 \sin^2 \alpha \right]. \quad (9)$$

Чисельник $\Omega_{c-f}(p, \theta)$, що містить величини ν, α і θ_0 в якості параметрів, через свою громіздкість тут не приводиться. Інтегрування здійснюється вздовж довільної прямої L , паралельної уявній осі, що лежить у смугі аналітичності $-\sigma_1 < \text{Re } p < 0$.

Аналітичні представлення розв'язків для граничних умов (4) і (5) мають вигляд, аналогічний (8), де в знаменниках підінтегрального виразу стоять відповідно функції

$$\Delta_{c-s}(p) = \sin(2p\alpha) - p \sin(2\alpha) \quad (10)$$

і

$$\Delta_{f-s}(p) = (3+\nu) \sin(2p\alpha) + p(1-\nu) \sin(2\alpha). \quad (11)$$

4. Аналіз розв'язку

Застосовуючи до інтегралу (8) теорему про лишки отримаємо представлення прогину (а отже й моментів та зусиль) в області $r < r_0$ через нескінченні степеневі ряди. Аналіз їх головних членів при $r \rightarrow 0$ дозволяє зробити наступні висновки:

1. У випадку жорстко защемленої та вільної від навантаження сторін пластини при кутах розхилу $\alpha < 180^\circ$ згинаючий момент M_θ та поперечне зусилля Q_θ мають при $r \rightarrow 0$ складну осцилюючу поведінку типу $r^\lambda [A \sin(\omega \ln r) + B \cos(\omega \ln r)]$ з показниками λ_1 та $\lambda_2 = \lambda_1 - 1$ відповідно. Тут A, B, ω і λ є відомі постійні величини, що залежать від α і ν . Показник λ_1 пов'язаний з найближчим зліва до уявної осі коренем p_{-1} трансцендентного рівняння $\Delta_{c-f}(p) = 0$ таким співвідношенням:

$$\lambda_1 = -\text{Re } p_{-1} - 1. \quad (12)$$

Залежність показників λ_1 і λ_2 від кута α при $\nu = 0,3$ наведена на рис. 1а. Встановлено існування критичних кутів $\alpha^*(\nu)$, $\alpha^{**}(\nu)$, таких, що при $\alpha > \alpha^*$ вершина клина є особливою точкою. Для кутів розхилу $\alpha^* < \alpha < \alpha^{**}$ біля вершини клину спостерігається затухаюча осциляція моментів та необмежена за амплітудою осциляція поперечних зусиль. Для кутів $\alpha^{**} < \alpha < \pi$ значення моментів і значення поперечних зусиль є необмеженими за амплітудою біля вершини клину. Для кутів $\alpha < \alpha^*$ вершина клину є звичайною точкою: біля неї значення моментів і значення поперечних зусиль є обмеженими. Залежність критичних кутів α^* , α^{**} від коефіцієнту Пуассона ν наведено на рис 2а і 2б відповідно.

У випадку півплощини ($\alpha = 180^\circ$) рівняння $\Delta_{c-f}(p) = 0$ допускає аналітичний розв'язок:

$$p_{\pm} = k + \frac{1}{2} \pm \frac{i}{2\pi} \omega_0, \quad \omega_0 = \ln \frac{1-\nu}{3+\nu}. \quad (13)$$

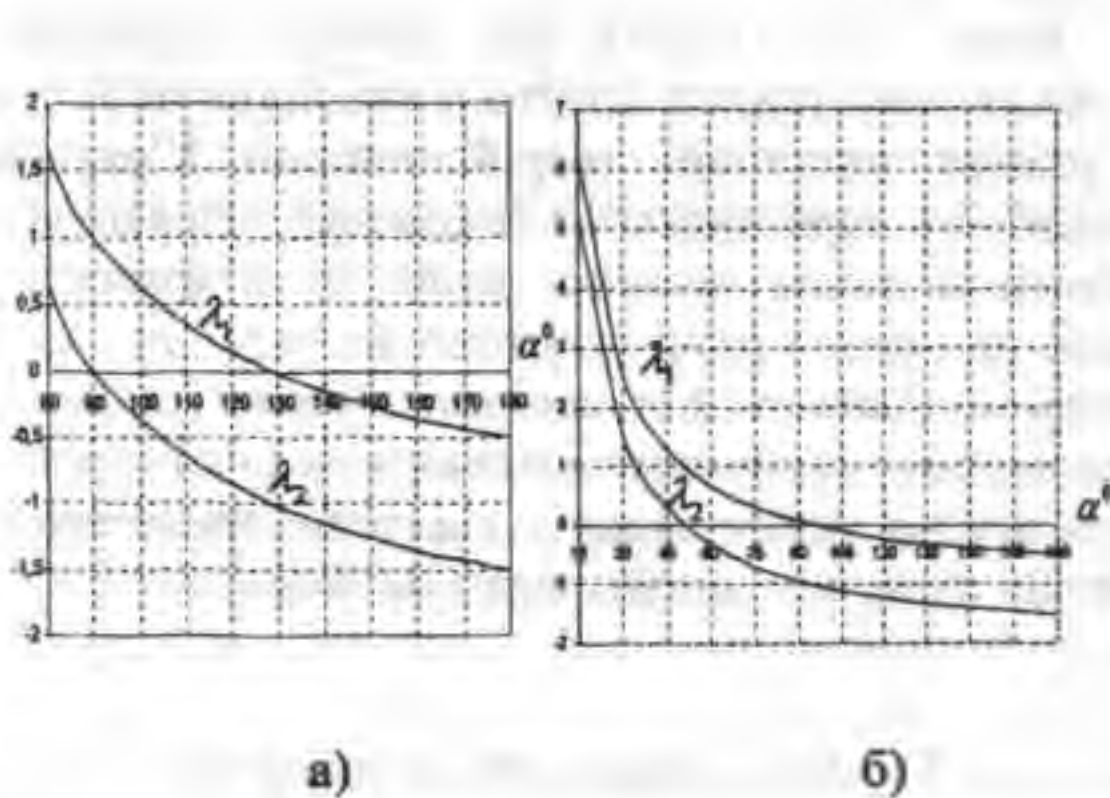


Рис.1 Поведінка степеневих показників для локальних особливостей в згинаючому моменті (λ_1) та поперечному зусиллі (λ_2) в залежності від кута пластини α ; а) жорстко защемлений та вільний від навантаження край; б) жорстко защемлений та шарнірно опертий край

Таким чином, вираз для згинаючого моменту M_θ при $r < r_0$ має вигляд

$$M_\theta(r, \theta) = M \sum_{k=1}^{\infty} \left(\frac{r}{r_0} \right)^{k-3/2} \left[b_k(\theta) \cos(\omega \ln \frac{r}{r_0}) - a_k(\theta) \sin(\omega \ln \frac{r}{r_0}) \right],$$

де $M = P \frac{1-\nu}{4(1+\nu)\pi}$, а функції $a_k(\theta)$ і $b_k(\theta)$

мають громоздкий вигляд, який тут не наведено, та містять величини θ_0 і ν в якості параметрів.

Аналізуючи перші члени цього ряду і аналогічних рядів для прогину і поперечного зусилля, робимо висновок, що біля точки зміни типу граничних умов мають місце осциляції виду:

$$\begin{aligned} w(r, \theta) &\sim r^{3/2} [A_w \sin(\omega_0 \ln r) + B_w \cos(\omega_0 \ln r)], \\ M_\theta(r, \theta) &\sim r^{-1/2} [A_M \sin(\omega_0 \ln r) + B_M \cos(\omega_0 \ln r)], \\ Q_\theta(r, \theta) &\sim r^{-3/2} [A_Q \sin(\omega_0 \ln r) + B_Q \cos(\omega_0 \ln r)]. \end{aligned}$$

2. У випадку жорстко защемленої і шарнірно опертої сторін пластини при куті розхилу $\alpha < 180^\circ$, найближчий до уявної осі корінь рівняння $\Delta_{\text{с-м}}(p) = 0$ є дійсним для довільного кута розхилу пластини α . Тому, згинаючий момент M_θ і поперечне зусилля Q_θ мають при

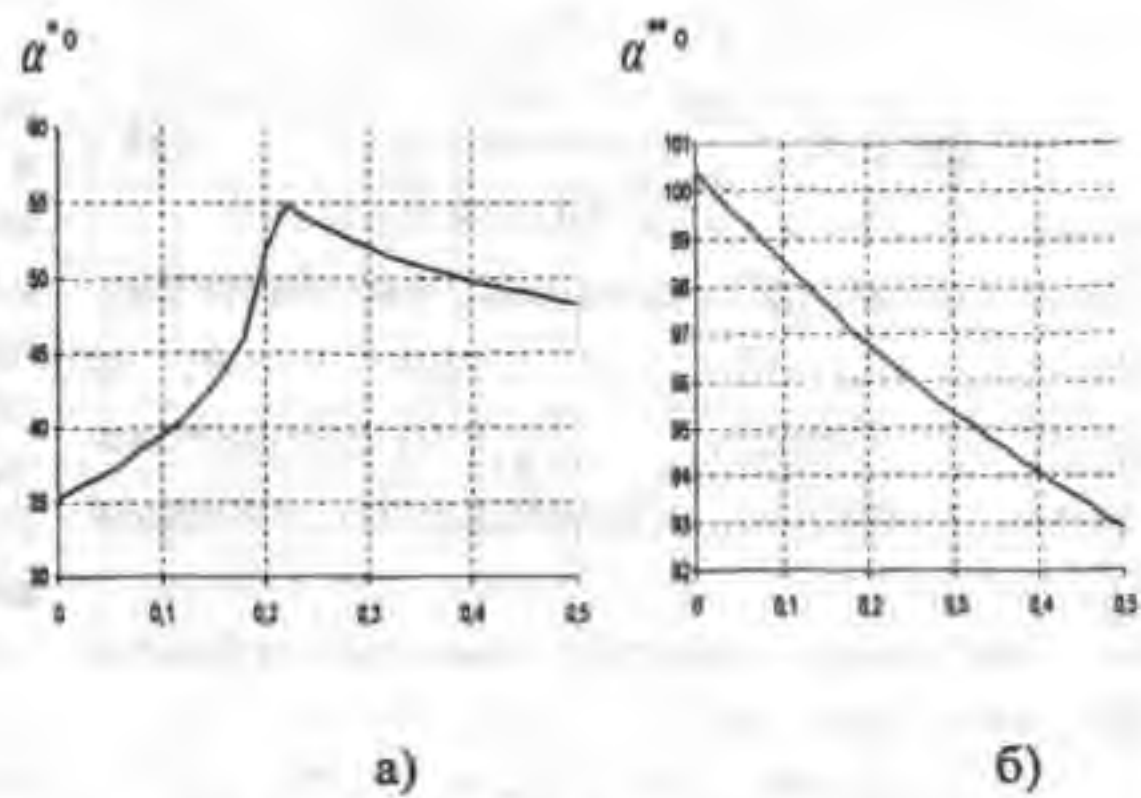


Рис.2 Залежність критичних кутів α^* , α^{**} від коефіцієнту Пуассона ν

$r \rightarrow 0$ степеневу поведінку типу Ar^λ і Br^λ з показниками λ_1 і $\lambda_2 = \lambda_1 - 1$ відповідно. Тут A , B є відомі постійні величини, що залежать від α . Залежність показників λ_1 і λ_2 від кута розхилу α наведена на рис.1б. Зауважимо, що в цьому випадку показники поведінки силових характеристик від коефіцієнту Пуассона ν не залежать. Видно, що при $\alpha > 90^\circ$ поперечне зусилля має особливість, а при $\alpha > 129^\circ$ особливість спостерігається також в згинаючому моменті.

Для частинного випадку півплощини ($\alpha = 180^\circ$), коли сила P прикладена на бісектрисі ($\theta_0 = 90^\circ$), прогин має поведінку $w(r, \theta) \sim Cr^{3/2}$ при $r \rightarrow 0$. Прогин вздовж цієї лінії в області $r < r_0$ виражається формулою

$$w\left(r, \frac{\pi}{2}\right) = \frac{Pr_0^2}{2\pi D} \left[\rho \operatorname{arctg} \sqrt{\rho} - \rho^2 \operatorname{arctg} \rho + \frac{1}{4}(1-\rho^2) \ln \frac{1-\rho}{1+\rho} + \frac{1}{2}\rho \right], \quad \rho = \frac{r}{r_0}. \quad (14)$$

Залежність від r згинаючого моменту та поперечного зусилля на защемлені $\theta = 0^\circ$ при $r < r_0$ має вигляд

$$M_\theta(r, 0) = -\frac{\sqrt{2}P}{2\pi} \frac{\sqrt{r/r_0} + \sqrt{r_0/r}}{1 + (r/r_0)^2}, \quad (15)$$

$$Q_\theta(r, 0) = \frac{2P}{\pi r_0} \frac{1}{(1 + (r/r_0)^2)^2}. \quad (16)$$

Таким чином, на защемленні має степеневу особливість $M_\theta(r, 0) \sim Ar^{-1/2}$ при $r \rightarrow 0$, а поперечне зусилля $Q_\theta(r, 0)$ залишається скінченним і досягає максимального значення $0,637P/r_0$ при $r = 0$.

На шарнірно опертій частині границі $\theta = 180^\circ$ маємо при $r < r_0$

$$Q_\theta(r, \pi) = \frac{\sqrt{2}P}{4\pi r_0} \frac{\rho^{-3/2} + \rho^{-1/2} + 5\rho^{1/2} - 3\rho^{3/2} - 4\sqrt{2}}{(1 + \rho^2)^2}.$$

Отже, поперечне зусилля має степеневу особливість $Q_\theta(r, \pi) \sim Br^{-3/2}$ при $r \rightarrow 0$.

3. У випадку, коли одна із сторін пластини ($\theta = 0$) вільна від навантаження, а інша ($\theta = \alpha$) шарнірно оперта, найближчий до вісі корінь рівняння $\Delta_{f-\alpha}(\rho) = 0$ для довільного α є дійсними для всіх значень коефіцієнта Пуассона, тому осциляції в даній задачі не спостерігаються. При $\nu = 0,3$ особливість у поперечного зусилля виникає при кутах $\alpha > 50^\circ$, а у згинаючих моментах – при кутах $\alpha > 90^\circ$.

У випадку півплощини ($\alpha = 180^\circ$) степенева поведінка згинаючого моменту і поперечного зусилля має вигляд $M_\theta \sim Ar^{-1/2}$, $Q_\theta \sim Br^{-3/2}$ при $r \rightarrow 0$ відповідно. Таким чином, точка зміни типу граничних умов є особливою.

5. Висновки

В роботі методом інтегрального перетворення Мелліна побудовані точні розв'язки задач про згин тонкої пружної клиноподібної пластини зосередженою силою для різних випадків закріплення її сторін.

Порівнюючи отримані результати локальної поведінки згинаючого моменту і поперечного зусилля для півплощини як частинного випадку клиноподібної пластини з різними умовами закріплення її сторін з результатами робіт [2,3] для круглої пластини, не було отримано повної відповідності. Особливість виду $\ln x / \sqrt{x}$ при $x \rightarrow 0$ локальної поведінки поперечного зусилля для випадку шарнірно опертої і вільної від навантажень границі [2] не була одержана. Це питання потребує подальшого вивчення шляхом більш уважного розгляду складного шляху

побудови розв'язку інтегрального рівняння у вказаній статті.

Крім того, підхід до аналізу локальних особливостей деяких інтегральних характеристик в рамках класичної теорії пластин Кірхгофа потребує серйозного уточнення. Правильна поведінка таких величин може бути отримана лише при розв'язанні просторових задач лінійної теорії пружності для клиноподібного шару зі змішаними граничними умовами на його гранях. Розв'язання таких задач і сьогодні представляє виклик сучасним аналітичним методам.

Список використаних джерел

1. Тимошенко С.П. Пластинки и оболочки / С.П. Тимошенко, С. Войновский-Кригер. – М.: Наука, 1963. – 636 с.
2. Monegato G. On the form of the contact reaction in a solid circular plate simply supported along two antipodal edge arcs and deflected by a central transverse concentrated force / G. Monegato, A. Strozzi // J. Elasticity. – 2002. – Vol.68. – P.13–35.
3. Strozzi A. Solid circular plate clamped along two antipodal edge arcs and deflected by a central transverse concentrated force / A. Strozzi, G. Monegato // J. Elasticity. – 2009. – Vol.97. – P.155–171.
4. Meleshko V.V. Normal reaction in a clamped elastic rectangular plate / V.V. Meleshko, A.M. Gomilko, A.A. Gourjii // J. Eng. Math. – 2001. – Vol.40. – P.377–398.
5. Шерман Д.И. Об изгибе круглой пластинки, частично защемленной и частично опертой вдоль контура / Д.И. Шерман // Докл. АН СССР. – 1955. – Т.101, Вып.3. – С.623–626.
6. Шерман Д.И. Об изгибе круглой пластинки, частично опертой и частично свободной вдоль контура / Д.И. Шерман // Докл. АН СССР. – 1955. – Т.105, Вып.6. – С.1180–1183.
7. Лебедев Н.Н. Сборник задач по математической физике / Н.Н. Лебедев, И.П. Скальская, Я.С. Уфлянд. – М.: Гостехтеоретиздат, 1955. – 420с.
8. Трантер К.Дж. Интегральные преобразования в математической физике / К.Дж. Трантер. – М.: Гостехиздат, 1956. – 204с.
9. Уфлянд Я.С. Интегральные преобразования в задачах теории упругости / Я.С. Уфлянд. – Л.: Наука, 1967. – 402 с.

Надійшла до редколегії 31.01.11

УДК 539.376

Тетяна С. Краснопольська, д.ф.-м.н.
Євгеній Д. Печук*, аспірант

Побудова еволюційних рівнянь по даним вихідного сигналу

Реалізовано і порівняно два основні методи побудови еволюційних рівнянь динамічних систем по вихідному сигналу. Перший – метод послідовного диференціювання, другий – метод затримки. Вперше для знаходження часу оптимальної затримки в другому методі реконструкції використано сталий проміжок залежності дивергенції реконструйованої системи від параметра затримки.

Ключові слова: еволюційне рівняння, реконструкція, динамічна система, послідовне диференціювання, затримка.

*E-mail: uzuzun@i.ua

Статтю представив член редколегії д.ф.-м.н., професор Мелешко В.В.

Вступ

Вивчення складних часових сигналів представляє важливу і актуальну задачу сучасного природознавства [4]. Особливо вагомим є її розв'язок, коли зрозуміти динаміку фізичного явища і прогнозувати його еволюцію з плином часу, можливо лише, з аналізу сигналів, що фіксуються на його виході. Так, наприклад, аналіз роботи серця людини проводиться на основі дослідження електрокардіограм, рух об'єктів в астрофізиці – на основі сигналів випромінювання з космосу, в акустиці – по сигналам тіл, що рухаються. Отримані таким чином сигнали можуть бути достатньо складними. Дана робота присвячена апробації і подальшій розробці двох методів побудови еволюційних рівнянь, тобто побудови динамічної системи, по часовому ряду вихідного сигналу [1, 2, 3, 6]. Метою нашої роботи є знаходження детермінованих систем, розв'язки яких представляють собою детерміновані періодичні, квазіперіодичні та хаотичні усталені процеси.

В цій роботі порівняно два основні методи реконструкції еволюційних рівнянь по одновимірному часовому ряду – розв'язку конкретної динамічної системи третього порядку. Тому, розв'язок цієї системи буде порівнюватись

Tatyana S. Krasnopol'skaya, Dr.Sc.
Evgeniy D. Pechuk*, PhD student

Reconstruction of the evolution equations from data of the output signal

Two main methods of reconstruction of the evolution equations from data of the output signal have been used and compared. First one is the method of successive differentiation and the second is based on delay coordinates. It was firstly suggested to choose a time delay parameter from the stable region of a divergence of the reconstructed system.

Key Words: equation of the evolution, reconstruction, dynamical system, successive differentiation, delay.

з розв'язками побудованих по ньому систем також третього порядку.

Методи побудови динамічних систем

Нехай, в результаті експерименту, маємо залежність змінної $a = a(t)$ від часу, задану на скінченному проміжку часу t_0 з кроком (дискретизацією) Δt : $a(i\Delta t) = a_i$, $i = 1, \dots, N$, $N = [t_0 / \Delta t]$. Припустимо, що $a(t)$ являє собою одновимірну проекцію фазової траєкторії деякої динамічної системи. Метою задачі відновлення динамічної системи є побудова системи еволюційних рівнянь, розв'язок якої достатньо точно співпадає з $a(t)$. Таким чином, реконструкція належить до класу обернених задач з неоднозначним розв'язком.

Позначимо координати стану реконструйованої системи через $\bar{x}(t) = (x_1, x_2, x_3)$. Існує два основні методи реконструкції. Ідея першого метода – метода послідовного диференціювання [1, 2, 3] полягає в тому, що фазові змінні системи записуються наступним чином:

$$\bar{x}(t) = (a(t), \dot{a}(t), \ddot{a}(t)) \quad (1)$$

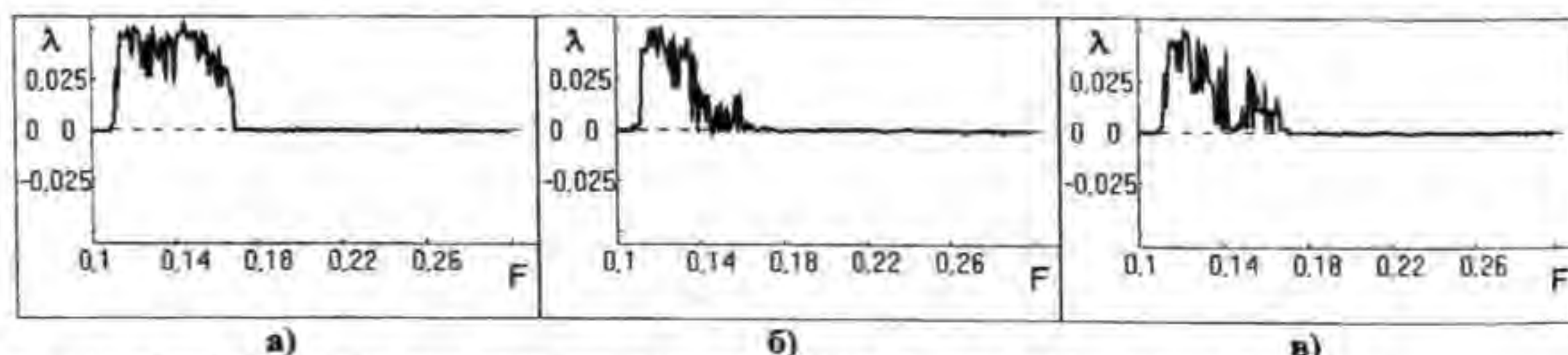


Рис. 1 Старший Ляпуновський показник а) системи (6) та систем, побудованих б) методом диференціювання і в) методом затримки.

Відповідна математична модель, в загальному випадку, має наступний вигляд:

$$\begin{cases} \dot{x}_1 = x_2; \\ \dot{x}_2 = x_3; \\ \dot{x}_3 = F(x_1, x_2, x_3), \end{cases} \quad (2)$$

де F – невідома нелінійна функція. Для одержання конкретного вигляду системи (2) функція F представляється у вигляді розкладу за деякою базою функцій, де число членів розкладу залежить від необхідної точності апроксимації. В нашій роботі в якості базових функцій обрано поліноми наступного вигляду:

$$F(\vec{x}) = c_0 + \sum_{i=1}^3 c_i x_i + \dots + \sum_{a, m, n, k, j, l=1}^3 c_{amknjl} x_a x_m x_n x_k x_j x_l, \quad (3)$$

де c_i – невідомі коефіцієнти. Оскільки всі координати вектора $\vec{x}(t)$ відомі, то для кожного a_i , система (2) представляє собою лінійне рівняння відносно невідомих коефіцієнтів c_i . Тому, для знаходження конкретного вигляду математичної моделі, необхідно розв'язати систему з N лінійних алгебраїчних рівнянь з числом невідомих, що дорівнює числу невідомих коефіцієнтів в розкладі функції F . Як правило, це число значно менше за N , тому відповідну систему розв'язують методом найменших квадратів.

Другий метод реконструкції називається методом затримки [1, 2, 3, 6]. При його застосуванні відповідні просторові змінні будуються за допомогою вибору деякого фіксованого проміжку часу $\tau \ll t_0$:

$$\vec{x}(t) = (a(t), a(t + \tau), a(t + 2\tau)). \quad (4)$$

Відповідна математична модель має наступний вигляд:

$$\begin{cases} \dot{x}_1 = F_1(x_1, x_2, x_3); \\ \dot{x}_2 = F_2(x_1, x_2, x_3); \\ \dot{x}_3 = F_3(x_1, x_2, x_3), \end{cases} \quad (5)$$

де F_i – нелінійні функції. Якщо в якості нелінійних функцій обрано поліноми вигляду (3), то для кожного a_i , система (5) представляє собою три лінійні рівняння відносно невідомих коефіцієнтів розкладу математичної моделі, необхідно знайти розв'язок трьох систем з N лінійних алгебраїчних рівнянь. Основною складністю цього метода є вибір значення затримки τ . Її значення, наприклад, може бути отримано, як час досягнення першого нуля автокореляційної функції [1, 2], побудованої по сигналу $a(t)$. Однак, при такому виборі затримки, в деяких випадках, спостерігається порушення дисипативності реконструйованої системи. Враховуючи той факт, що реальні системи неконсервативні їх дивергенція від'ємна, то час затримки потрібно брати з проміжка, де значення дивергенції реконструйованої системи в усталеному режимі від'ємне.

В якості прикладу, на якому ми будемо тестувати застосування методів відновлення динамічної системи візьмемо плоский фізичний маятник при обмеженому збуренні [5]. Нелінійні коливання маятника $\alpha(t)$ відшукуються у вигляді [5]:

$$\alpha(t) = (a/l)^{1/3} [y_1(t) \cos \Theta + y_2(t) \sin \Theta].$$

Після проведення процедури усереднення по швидкому часу Θ і переході до похідних по повільному часу, отримують наступну нелінійну динамічну систему [5]:

$$\begin{cases} \frac{dy_1}{dt} = -0.1y_1 - y_2y_3 - \frac{1}{8}(y_1^2y_2 + y_2^3) \\ \frac{dy_2}{dt} = -0.1y_2 + y_1y_3 - \frac{1}{8}(y_2^2y_1 + y_1^3) + 1, \\ \frac{dy_3}{dt} = -0.5y_2 - 0.61y_3 + F \end{cases} \quad (6)$$

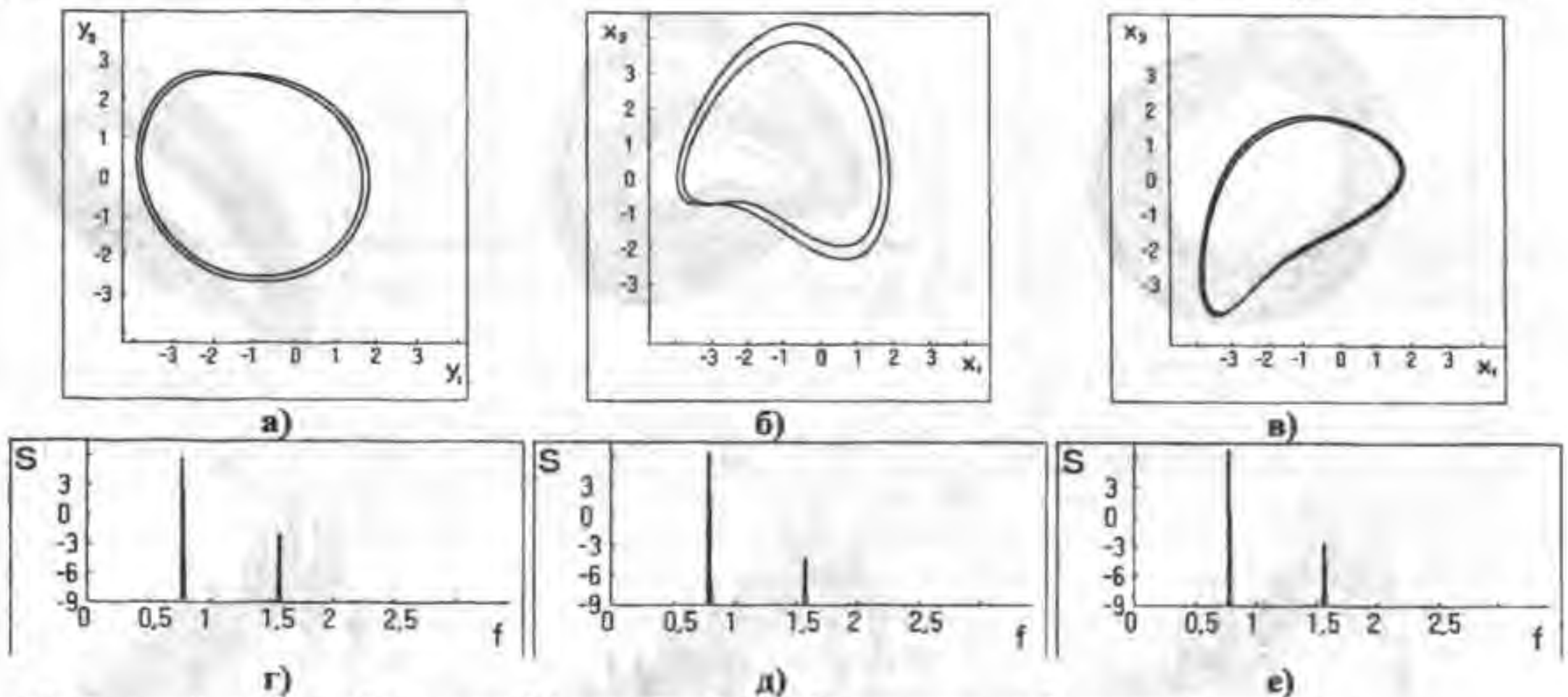


Рис.2 Фазовий портрет а) системи (6) при $F = 0.257$ б), системи, побудованої методом диференціювання і в) методом затримки та їх спектри потужності г) – е).

де F – біфуркаційний параметр.

Для отримання конкретних вихідних сигналів, брались значення параметра з проміжку $0.1 \leq F \leq 0.3$ і кроком 0.001. Потім отримані системи було розв'язано методом Рунге – Кути п'ятого порядку з використанням коректуючої процедури Дормана – Принса. Сигнали знімалися з перших координат розв'язків в усталеному режимі через $\Delta t = 0.004$ і з $N = 2^{18}$. В результаті отримано масив з 200 сигналів, кожному з яких відповідає значення з проміжку $0.1 \leq F \leq 0.3$.

Порівняння методів реконструкції динамічних систем

Для кожного з отриманих вихідних сигналів було побудовано систему еволюційних рівнянь, використовуючи перший і другий методи реконструкції. Для вибору часу затримки $\tau = n\Delta t = n \cdot 0.004$ в другому методі використано наступну методику. По вихідному сигналу будемо його автокореляційну функцію і знаходимо її перший корінь a_0 . Потім, для кожного $0 < n \leq [a_0 / \Delta t]$ будемо систему методом затримки і знаходимо її дивергенцію в усталеному режимі. На графіку залежності отриманої дивергенції від n обираємо потрібне значення з області, де дивергенція неперервна, від'ємна і має майже нульову похідну. Так обираємо значення $n = 240$ для регулярного і $n = 120$ для хаотичного сигналу.

Залежність старшого Ляпуновського показника системи (6) від значень біфуркаційного параметра F зображено на рис.

1а). На рис. 1б) і рис. 1в) зображено значення старшого Ляпуновського показника систем, побудованих за першим і другим методом відповідно, по вихідним сигналам, що відповідають своїм значенням біфуркаційного параметра F . З рисунків випливає, що області хаотичності (показник додатній) в реконструйованих системах менші за відповідні області вихідної системи. Системи, побудовані за другим методом більш точно передають вихідний динамічний режим (вікна періодичності в області хаотичності мають менший розмір), ніж системи, побудовані за першим методом.

На рис. 2 зображено двовимірні проекції фазових портретів та спектри потужності вихідної системи та реконструйованих систем в усталеному режимі. Як видно з рисунку, в усіх трьох випадках проекцією атрактора є двооборотний цикл, а спектри потужностей відповідних часових реалізацій дискретні з двома основними частотами. Обчислення показують, що метод диференціювання відновлює сигнал, спектр потужності якого співпадає зі спектром вихідного сигналу з точністю більше 95%, метод затримки – більше 97%.

На рис. 3 зображено аналогічні характеристики для хаотичного вихідного сигналу системи (6) в усталеному режимі. Проекції фазових портретів вихідної та реконструйованих систем представляють собою проекції хаотичних атракторів стрічкової структури. Відповідні спектри потужності неперервні і можуть бути апроксимовані однією лінійною функцією затухання $S = -6.75 - 8.5f$.

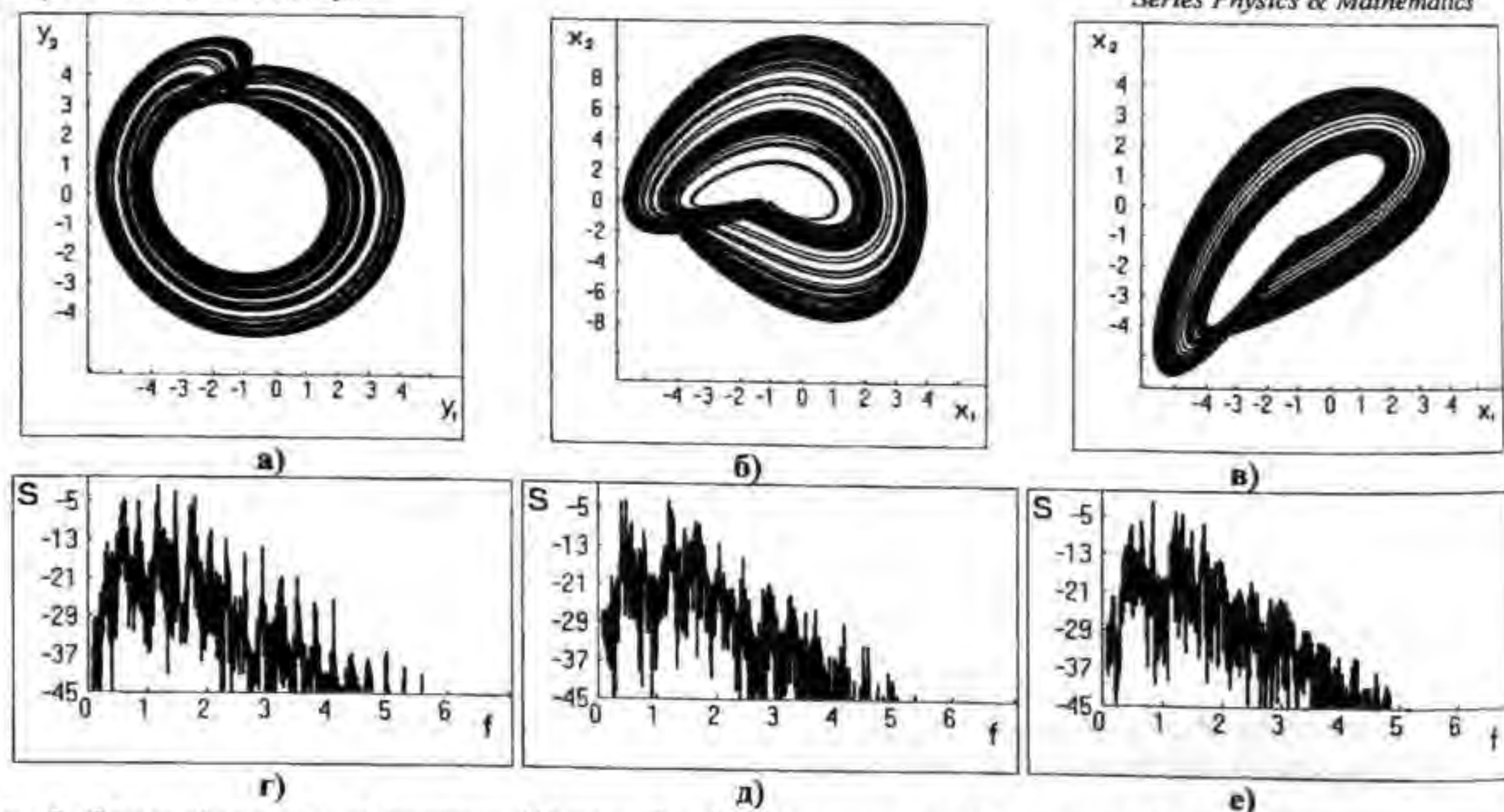


Рис. 3 Фазовий портрет а) системи (6) при $F = 0.114$, системи, побудованої б) методом диференціювання і в) методом затримки та їх спектри потужності г) – е).

Висновки

Таким чином, в роботі реалізовано і порівняно два основні методи побудови динамічних систем по вихідному сигналу.

Враховуючи неконсервативність реальних динамічних систем, для визначення часу оптимальної затримки в другому методі реконструкції використано лінійний проміжок на залежності від'ємної дивергенції реконструйованої системи від параметра затримки. На конкретному прикладі вихідних

сигналів системи, що описує коливання плоского фізичного маятника, протестовано методи реконструкції динамічної системи. При дослідженні систем та сигналів використано спектральні методи, побудову фазових портретів та старший Ляпуновський показник. В результаті дослідження виявлено, що для побудови еволюційних рівнянь по хаотичним вихідним сигналам, слід віддавати перевагу методу затримки з коректним вибором параметра затримки.

Список використаних джерел

1. Crutchfield J.P. Equations of Motion from a Data Series / J.P. Crutchfield, B.S. McNamara. // Complex Systems – 1987. – 1, N 14. – P. 417–452.
2. Анищенко В. С. Знакомство с нелинейной динамикой / В. С. Анищенко. – М.-Ижевск: Институт компьютерных исследований, 2002. – 142 с.
3. Болотин Ю. Л. Конструктивный хаос / Ю. Л. Болотин, А. В. Тур, В. В. Яновский. – Харьков: Институт монокристаллов, 2005. – 420 с.
4. Грінченко В. Т. Основи акустики / В. Т. Грінченко, І. В. Вовк, В. Т. Маципура. – Київ: Наукова думка, 2007. – 640 с.
5. Краснопольская Т. С. Регулярная и хаотическая динамика систем с ограниченным возбуждением / Т. С. Краснопольская, А. Ю. Швеце. – М.-Ижевск: Институт компьютерных исследований, 2008. – 278 с.
6. Яновский В. В. Лекции о нелинейных явлениях. Том 1 / В. В. Яновский. – Харьков: Институт монокристаллов, 2006. – 456 с.

Надійшла до редколегії 11.01.2011

УДК 539.3

Наталія І. Ободан*, д.т.н., професор
Наталія А. Гук**, к.ф.-м.н., доцент

Обернена задача визначення зовнішніх навантажень при деформації тонкостінних оболонок

Розглядається задача відновлення функції зовнішнього навантаження, діючого на циліндричну оболонку, з урахуванням особливостей нелінійного напружено-деформованого стану. Підхід дозволяє відновити невідомі параметри у всьому діапазоні дії навантаження.

Ключові слова: ідентифікація, циліндрична оболонка, нелінійний напружено-деформований стан, обернена задача, початкове наближення.

Natalia I. Obodan*, Professor
Natalia A. Guk**, PhD

Inverse problem of determination of external load at deformation of the thin-walled shells

The problem of identification of external loading, account the nonlinear stress-strain state of cylinder shell, is considered. Approach allows to identify the function of external influence in all of range of loading action. Solutions of the load parameters recovery boundary problem are presented.

Keywords: identification, cylinder shell, nonlinear stress-strain state, inverse problem, initial approaching.

* E-mail: vobodan@mail.ru

** E-mail: nguk@farlep.dp.ua

Статтю представив член редколегії д.ф.-м.н., професор Мелешко В.В.

1 Вступ

Розв'язки обернених задач механіки деформівного твердого тіла знаходять широке застосування в машинобудуванні, енергетиці, медицині [1]. Їх дослідженню присвячена велика кількість робіт, огляд яких приведений в [2].

Дана робота присвячена розв'язанню однієї з важливих обернених задач теорії оболонок – реконструкції зовнішнього навантаження в діапазоні, що містить значення, які є близькими до критичних, за наслідками спостереження за напружено-деформованим станом (НДС).

2 Постановка задачі

Вектор-функція стану $U(X)$ тонкостінного оболонкового елемента в обмеженій просторовій області $\Omega = \{X : X \in \Omega\}$, який знаходиться під дією зовнішнього поверхневого або (та) зосередженого навантаження, описується системою нелінійних диференціальних рівнянь:

$$\begin{aligned} \nabla_j (D_1^{ijkl} \nabla_k u_l) - \nabla_j \left[D_1^{ijkl} \left(B_{kl} w - \frac{1}{2} w_{,k} w_{,l} \right) \right] + R_1 &= 0 \\ \nabla_{ij} (D_2^{ijkl} \nabla_{kl} w) - D_1^{ijkl} \varepsilon_{kl} (B_{ij} + \nabla_{ij} w) - R_3 &= 0, \end{aligned} \quad (1)$$

де $U(X) = (u_1, u_2, w)$ – вектор-функція переміщень в напрямках x_1, x_2 та нормалі; $X = \{x_1, x_2\}$ – вектор координат;

$R(X) = (R_1, R_2, R_3)$ – вектор-функція, що описує зовнішній вплив; ε_{ij} – тангенціальна складова узагальненої деформації серединної поверхні оболонки; B_{kl} , B_{ij} – кривизни серединної поверхні; D_1^{ijkl} , D_2^{ijkl} – функції, що характеризують жорсткість оболонки на розтягування та вигин; $\nabla_{ij}(\cdot) = \partial^2(\cdot)/\partial x_i \partial x_j$; $\nabla_i(\cdot) = \partial(\cdot)/\partial x_i$, $i, j, k, l = 1, 2$. Прийнята угода про підсумовування по індексах, що повторюються.

На границі Γ області Ω сформульовані умови:

$$L_\Gamma(U(X), R(X)) = 0, \quad (2)$$

де $L_\Gamma(\cdot)$ – заданий диференціальний оператор.

За наслідками спостережень за деформуванням оболонки, які описуються вектор-функцією

$$\bar{\sigma}(X, R) = \sigma(X, R)|_{X \in \bar{\Omega}} = \sigma^*(\bar{X}), \quad (3)$$

$\bar{X}$ – область, де спостерігаються значення $\sigma^*(\bar{X})$, необхідно визначити вектор-функцію $R(X)$. Для отримання розрахункових значень $\bar{\sigma}(X, R)$ використовується математична модель деформування тонкостінної оболонки (1), (2).

Вектор-функція невідомих задачі з областю визначення $\bar{R} = \{0 \leq R_1 \leq P_1; 0 \leq R_2 \leq P_2; 0 \leq R \leq P_3\}$ містить в якості компонент функції, що описують довільний розподіл зовнішніх навантажень, P_p – задані константи, $p = 1, 3$.

Вказана обернена задача формулюється як варіаційна [5], тобто здійснюється перехід до задачі мінімізації відстані $\rho_{L_2\Omega}$ між вектор-функцією $\bar{\sigma}(X, R)$, що характеризує НДС, і вектор-функцією вимірювань $\sigma^*(X)$. Тут $\bar{\sigma}(X, R) = \{\varepsilon_{ij}(R), \chi_{ij}(R)\}^T$ – вектор-функція, компоненти якої визначаються з розв'язку прямої задачі нелінійної теорії оболонок (1), (2) при відомих значеннях $R(X)$; $\varepsilon_{ij}, \chi_{ij}$ – тангенціальна і вигиниста складові узагальненої деформації середньої поверхні оболонки, виражені через компоненти вектора переміщень $U(X)$; $\sigma^* = \{\varepsilon_{ij}^*, \chi_{ij}^*\}^T$ – значення відповідних функцій, що спостерігаються.

Відстань $\rho_{L_2\Omega}$ обирається як

$$\rho_{L_2\Omega} = \int_{\Omega} (\bar{\sigma}(X, R) - \sigma^*)^T (\bar{\sigma}(X, R) - \sigma^*) d\Omega.$$

Тоді $R(X)$ визначається з умови, аналогічної [1]

$$R(X) = \arg \min_{R \in \bar{R}} \rho_{L_2\Omega}(\bar{\sigma}(X, R), \sigma^*(X)), \sigma^* \in \bar{U}, \quad (4)$$

де $\bar{R}$ – компактна множина, яка відповідає класу коректності задачі (1) – (3); $\bar{U}$ – множина можливих НДС оболонки.

3 Математична модель

Для опису m -ого стану оболонки введемо вектор-функцію

$$\{(U^m(X, R^m), R^m(X)) \mid R^m(X) \in \bar{R}, U^m(X, R^m) \in \bar{U}\},$$

компонентами якої є $U^m(X, R^m)$ – переміщення точок середньої поверхні оболонки, визначені при $R^m(X)$ із розв'язку прямої задачі (1), (2) $R^m(X)$ – можливий розв'язок оберненої задачі.

Наближений розв'язок для m -ого стану оболонки представимо у вигляді [3]

$$R^m(X) = R_0 + \Delta R^m,$$

$$U^m(X, R^m) = U_0(X, R_0) + \Delta U^m(X, R_0, \Delta R^m),$$

де $\Delta U^m(X, R_0, \Delta R^m) = (\Delta u_1^m, \Delta u_2^m, \Delta w^m)$ – приращення вектор-функції переміщень, викликане зміною функції зовнішнього навантаження ΔR^m ;

$U_0(X, R_0)$ – розв'язок прямої задачі при відомому навантаженні R_0 .

Для опису додаткового НДС будемо використовувати узагальнений розв'язок. Вектор-функція ΔU^m (надалі індекс m буде опущений) повинна задовольняти інтегральній тотожності для будь-якої вектор-функції $\Phi(\varphi_1, \varphi_2, \varphi)$

$$(\Delta U, \Phi)_{\Psi} = \int_{\Omega} \left[B_{kl} \Delta w - \frac{1}{2} (w_{0,kl} \Delta w_{,kl}) \right] D_1^{ijkl} \nabla_j \varphi_i + \\ + D_1^{ijkl} \left(\Delta \varepsilon_{kl} + \frac{1}{2} (w_{0,kl} \Delta w_{,kl} + w_{0,kl} \Delta w_{,kl}) \right) \cdot \\ \cdot (B_{ij} \varphi - \Delta w_{,ij} \varphi_{,ij}) + \Delta R \varphi dx_1 dx_2, \quad i, j, k, l = 1, 2, \quad (5)$$

де Ψ – спеціальний функціональний простір [3].

Вважаємо, що виконані умови $w|_{\Gamma_0} = 0$ на деякій частині контуру Γ_0 , решта умов на границі Γ є природними [3].

У роботі [3] показано, що, якщо напружено-деформований стан $U_0(X, R_0)$ визначається функцією навантаження R_0 і розв'язок для цього стану є єдиним, то при зміні функції R_0 на $R_0 + \Delta R$ при $\|\Delta R\|_{W_{2\Omega}^{\beta}} < \mu$, $\mu < 1$, де $W_{2\Omega}^{\beta}$ – функціональний простір Соболева, що складається з функцій, похідні яких порядку $\leq \beta$, належать простору $L_{2\Omega}$, розв'язуюче рівняння (1) для $U_0 + \Delta U$ має єдиний розв'язок. Звідси витікає, що при відомому неособливому стані $U_0(X, R_0)$ розв'язок задачі (1) – (4) є коректним при $\mu < 1$ [3].

Для визначення класу коректності, для якого виконується умова $\mu < 1$, перетворимо простір $\bar{R}$ невідомих функцій задачі в безперервний простір $\bar{R} = \{\delta_1 \leq \delta_1 \leq \bar{\delta}_1; \delta_2 \leq \delta_2 \leq \bar{\delta}_2\}$, такий, що кожній точці $R = (R_1, \dots, R_p) \in \bar{R}$ відповідає точка $\delta = (\delta_1, \delta_2) \in \bar{R}$, де $\delta_1 = \|R\|_{W_{2\Omega}^0}$, $\delta_2 = \|R\|_{W_{2\Omega}^1}$.

Використовуючи факт безпосереднього зв'язку між $\|\Delta R\|_{W_{2\Omega}^{\beta}}$ та δ_1, δ_2 [2], перейдемо від безкінечномірної задачі оптимізації функціоналу ρ по функції R , до скінченномірної задачі оптимізації по змінним δ_1, δ_2 . Якщо покроково задавати компоненти $\delta = (\delta_1, \delta_2)$ у вигляді $\delta_{ik} = \delta_{i,k-1} + \Delta \delta_i$ ($i = 1, 2$), де приращення $\Delta \delta_i$ на k -ому кроці є такими, що $\|\Delta R\|_{W_{2\Omega}^{\beta}} < \mu$, то кожне

значення $\Delta\delta_i$ визначає клас M станів оболонки, для яких виконується умова $\mu < 1$.

З умови, аналогічної (4), отримаємо

$$\delta^* = \arg \min_{\delta \in \bar{R}} \rho(\|\sigma(X, \delta)\|_{W_{2\Omega}^0}, \|\sigma^*\|_{W_{2\Omega}^0}). \quad (7)$$

Для відшукування розв'язків δ^* , що задовольняють умові (7), проведемо дискретизацію простору станів оболонки, замінюючи область $\bar{R}$ об'єднанням прямокутних областей, розміри яких визначаються кроком по параметрам δ_1, δ_2 .

Процедура визначення розв'язків R оберненої задачі зводиться до послідовного виконання двох етапів:

1. виділення класу коректності розв'язків оберненої задачі, що задовольняють умові (7);

2. визначення розв'язку оберненої задачі з використанням будь-якого методу локальної оптимізації на множині M розв'язків, яка виступає у ролі множини коректності, причому в якості початкового наближення можна обрати будь-який m -ий розв'язок з числа M .

4 Метод розв'язання

Параметричне представлення невідомих функцій задачі та функції спостереження здійснюється шляхом їх апроксимації за допомогою методу скінчених елементів (МСЕ).

Тоді замість (5) отримуємо систему лінійних алгебраїчних рівнянь у вигляді:

$$K(U_0)\Delta U = Q(\Delta R), \quad (8)$$

де $K(U_0)$ – матриця жорсткості, що залежить від розв'язку U_0 ; $\Delta U = \{\Delta U_v\}$, $\Delta R = \{\Delta R_z\}$ – вектори вузових значень прирощень переміщень і компонент шуканого вектора оберненої задачі; $Q = \{Q_v\}$ – вектор вузових значень правих частин; $v = 1, V$, $z = 1, Z$ – параметри сіток.

Розв'язок прямої задачі (8) дає можливість сформулювати вектор $\{\sigma_v\}$, $v = 1, V$ у вузлах сітки при фіксованому значенні R_0 і заданому прирощенні ΔR , відповідному $\Delta\delta_i$, причому для кожного фіксованого $\sigma_0(R_0)$ перевіряється умова рівності нулю визначника матриці системи (8). У разі її виконання фіксується точка $(\sigma_{\text{крит}}, R_{\text{крит}})$. Функція спостереження задається у вигляді вектору $\sigma^* = \{\sigma_s^*\}$, де $s = 1, S$ – вузли спостережень, $S \subset V$, відповідно формується вектор $\bar{\sigma} = \{\sigma_s\}$, $\{\sigma_s\} \subset \{\sigma_v\}$.

Умова (4) в дискретній формі матиме вигляд:

$$R = \arg \min_{R \in \bar{R}} (\bar{\sigma} - \sigma^*)^T (\bar{\sigma} - \sigma^*), \quad \sigma^* \in \bar{U}. \quad (9)$$

Для визначення початкового наближення $\{R_z^0\}$ організується ітераційний процес:

1. Задається початковий розподіл функції навантаження R_0 у вигляді $\{R_{0z}\}$, значення $\rho_{h_2}(\|\sigma(\delta)\|_{h_2}, \|\sigma^*\|_{h_2})$ обчислюються з кроком

$\Delta\delta_1 = \mu$, $\Delta\delta_2 = \mu$ із застосуванням методу продовження по параметрам δ_1, δ_2 [4]. На кожному k -му кроці методу продовження в якості U_0 виступає значення U_{k-1} , $k = 1, K$.

2. Здійснюється пошук всіх локальних екстремумів $\rho_{h_2}^d(\|\sigma(\delta)\|_{h_2}, \|\sigma^*\|_{h_2})$ ($d = 1, 2, \dots$), визначає номер точки локального мінімуму функціоналу).

3. Визначається початкове наближення, параметри розподілу якого відповідають

$$\delta^* = \arg \inf_d \rho_{h_2}^d(\|\sigma(\delta)\|_{h_2}, \|\sigma^*\|_{h_2}). \quad (10)$$

Після виконання описаної процедури, вектор значень $\{R_z^0\}$, відповідний $\delta^* = (\delta_1^*, \delta_2^*)$, обирається у якості початкового наближення для виконання чисельної мінімізації функціоналу (9) по компонентам $\{R_z\}$ методом Ньютона. Для уточнення компонент вектору навантаження $\{R_z\}$ організується ітераційний процес:

$$\Delta R^{(i+1)} = -A^{-1}(R^{(i)}) (\sigma(R^{(i)}) - \sigma^*), \quad (11)$$

де $A(R^{(i)})$ – матриця Фреше вектора нев'язки $(\sigma(R^{(i)}) - \sigma^*)$ по компонентам вектору параметрів $\{R_z\}$; i – номер кроку ітераційного процесу методу Ньютона.

5 Аналіз чисельних результатів

Запропонований підхід був застосований для відновлення локалізованих навантажень, що діють на тонкостінну оболонку, яка має геометричні параметри $L_0/D_0 = 4$, $D_0/h_0 = 200$ (L_0, D_0, h_0 – довжина, радіус, товщина оболонки). Розподіл нерівномірного поперечного зовнішнього тиску, що підлягає реконструкції, відповідає законам:

$$1) R = \lambda(a + b \cos x_2)^g; \quad 2) R = \lambda e^{-(\eta_1 x_1 + \eta_2 x_2^2)}.$$

У якості початкових наближень обирались розподіли вигляду 1), 2) та 3) $R = \lambda(a + b \cos \pi x_2)$. Для вказаних видів розподілу величини $\lambda, a, b, g, \eta_1, \eta_2, n$ є параметрами, при цьому λ

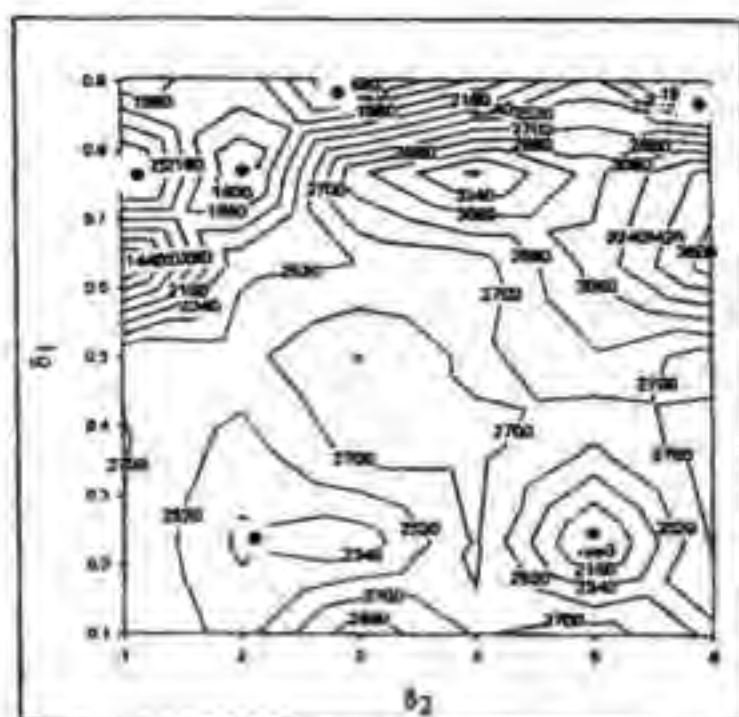
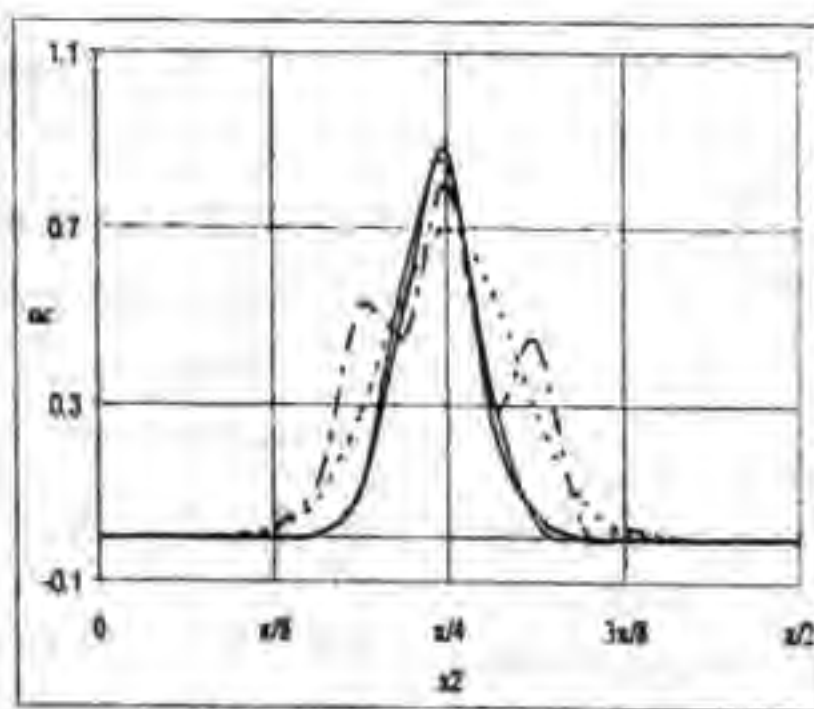
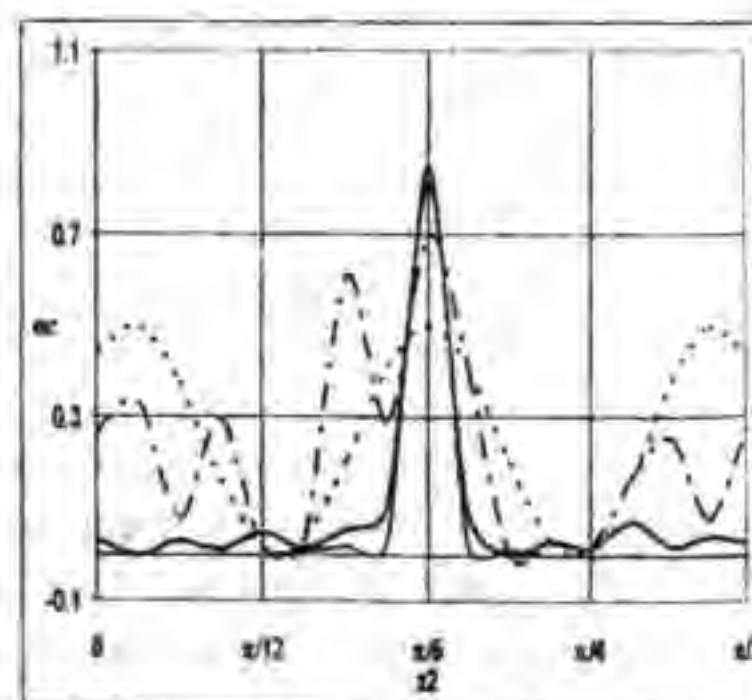


Рис. 1 – Вигляд функціоналу ρ_{12}



а



б

Рис. 2 – Процедура відновлення функції навантаження

характеризує рівень навантаження ($\lambda = \bar{\lambda}/\lambda_{\text{крит}}$, де $\lambda_{\text{крит}}$ – власні значення краєвої задачі (1), (2)). Аналіз був проведений для граничних умов, відповідних шарнірному опору торців оболонки $w = 0$, $u_2 = 0$, $M_{11} = 0$.

Розглянемо основні особливості реконструкції функції локалізованого навантаження, пов'язані з немонотонністю залежності значень $\|\sigma\|_{1/2}$ від значень δ_1 , δ_2 . На рис. 1 в лініях рівнів представлено поверхню, що відображає вигляд функціоналу $\rho_{12}(\|\sigma(\delta)\|_{1/2}, \|\sigma^*\|_{1/2})$, який відповідає стану оболонки при навантаженні вигляду 1) і σ^* , визначеному при $\lambda = 0.9$, в залежності від значень δ_1 , δ_2 (залежності побудовані для $\Delta\delta_1/\delta_1^0 = \mu$, $\Delta\delta_2/\delta_2^0 = \mu$, $\mu = 0.5$). На рис. 1 точками позначені локальні мінімуми $\rho_{12}(\|\sigma(\delta)\|_{1/2}, \|\sigma^*\|_{1/2})$. Необхідно відзначити, що розподіл функції R_0 впливає на збіжність методу Ньютона, але у всіх випадках, коли параметри розподілу початкового наближення визначаються з умови (10), збіжність спостерігається.

На рис. 2 а, б приведені етапи, що ілюструють збіжність ітераційної процедури відновлення функції навантаження (тонкою суцільною лінією позначений дійсний розподіл функції навантаження; пунктиром – початкове наближення; штрих-пунктиром – результат 1 ітерації; жирною суцільною лінією – результат відновлення) з використанням різних початкових наближень (на рис. 2, а дійсне навантаження відповідає розподілу 1), початкове наближення – розподілу 1); на рис. 2, б дійсне навантаження відповідає розподілу 2, початкове наближення – розподілу 3)).

6 Висновки

Запропонований підхід дозволяє ідентифікувати функцію зовнішнього навантаження, враховуючи особливості нелінійного НДС. Регуляризація задачі здійснюється за рахунок вибору множини коректності у вигляді μ -околу відомого розв'язку. Порівняльний аналіз результатів обчислювального експерименту показав, що відновлення параметрів можливе у всьому діапазоні зміни навантаження, навіть при наближенні до критичних значень.

Список використаних джерел

1. Тихонов А.Н. Математическое моделирование технологических процессов и метод обратных задач в машиностроении / А. Н. Тихонов, В.Д. Кальнер, В.Б. Гласко. – М.: Машиностроение, 1990. – 263 с.
2. Ватульян А.О. Обратные задачи в механике деформируемого твердого тела / А.О. Ватульян – М.: Физматлит, 2007. – 222 с.
3. Ворovich И.И. Математические проблемы нелинейной теории оболочек / И.И. Ворovich. – М.: Наука, 1989. – 373 с.
4. Шалашилин В.И. Метод продолжения по параметру и наилучшая параметризация / В.И. Шалашилин, Е. Б. Кузнецов. – М.: Эдиториал УРСС, 1999. – 222 с.
5. Гук Н.А. Выбор критерия идентификации в обратных задачах теории оболочек / Н.А. Гук, Н.И. Ободан, Г.М. Гавеля // Проблемы обчислювальної механіки і міцності конструкцій. – Д.: Наука і освіта. – 2010. – вип. 14. – С. 123–133.

Надійшла до редколегії 11.11.10

УДК 519.9

Євген Ю. Таран*, д.ф.-м.н., професор,
Роман Я. Кондрат, к.ф.-м.н.

Вплив мікродинаміки недеформівних ланцюгових макромолекул у розведеному розчині на його макрореологію

Демонструється, що відмінність макрореологічної поведінки розведених розчинів недеформівних вільнопротічних ланцюгових макромолекул у течіях одновісного розтягу і простого зсуву є наслідком різної мікродинаміки зважених макромолекул під дією гідродинамічних сил, які діють з боку низькомолекулярного розчинника.

Ключові слова: макрореологія, розведений розчин, недеформівні ланцюгові макромолекули, течія одновісного розтягу, течія простого зсуву

*E-mail: taran@univ.kiev.ua

Статтю представив проф. Мелешко В.В.

1 Вступ

Як показують експерименти, неньютонівська поведінка рідких середовищ у градієнтних течіях може залежати від багатьох чинників [1]. У рідких середовищах з мікроструктурою, зокрема у суспензіях та розчинах полімерних макромолекул, така поведінка залежить від динаміки елементів мікроструктури рідкого середовища, яка може змінюватися при зміні величини градієнта швидкості течії, або при зміні геометрії течії.

У даній роботі вивчається вплив мікродинаміки недеформівних вільнопротічних ланцюгових макромолекул у розведених розчинах з низькомолекулярними розчинниками на макрореологію розчинів. Для цього у роботі досліджується динаміка зважених макромолекул у течіях одновісного розтягу і простого зсуву таких розчинів та знаходяться аналітичний вираз і числові значення для ефективної в'язкості розчинів у течії одновісного розтягу. Для одержання цих результатів використовуються реологічні рівняння розведених розчинів недеформівних ланцюгових макромолекул, виведені у роботі [2].

Вивчення у [2] реологічної поведінки розведених розчинів недеформівних ланцюгових макромолекул у течії простого зсуву за

Ye.Yu. Taran*, D.Sci.(Phys.-Math.), Full Professor,
R.Ya. Kondrat, PhD (Phys.-Math.)

The influence of microdynamics of undeformable chain macromolecules in dilute solution on its macrorheology

It is demonstrated that a difference in a macrorheological behavior of dilute solutions of undeformable free-drained chain macromolecules in an uniaxial extension flow and in a simple shear flow is a consequence of the different microdynamics of suspended macromolecules under the action of hydrodynamic forces acting from side of a low-molecular solvent.

Key words: macrorheology, dilute solution, undeformable chain macromolecules, uniaxial extension flow, simple shear flow

допомогою одержаних реологічних рівнянь дозволило виявити пружнов'язку поведінку таких розчинів – псевдопластичне зменшення ефективної в'язкості розчинів зі збільшенням швидкості зсуву та ефект Вайссенберга, притаманний пружнов'язким рідким середовищам.

Ці результати досліджень стали можливими завдяки використанню в роботі [2] структурно-феноменологічного методу [3] при побудові реологічних рівнянь розведених розчинів недеформівних ланцюгових макромолекул. Цей метод дозволив виразити макрореологічні характеристики розчинів через їхні осереднені мікрохарактеристики.

У свою чергу це дозволило пояснити неньютонівську поведінку розведених розчинів недеформівних ланцюгових макромолекул їх кінематичною орієнтацією у простій зсувній течії під дією гідродинамічних сил з боку розчинника та протидією цій кінематичній орієнтації броунівським рухом, в якому беруть участь зважені макромолекули.

Закладений у реологічні рівняння розведеного розчину недеформівних ланцюгових макромолекул у роботі [2] зв'язок між мікродинамікою макромолекул і макрореологічними характеристиками розчину у цілому дозволив у даній роботі виявити відмінність динаміки макромолекул у течіях простого зсуву та одновісного розтягу їх розведених розчинів і, як наслідок, різну макрореологічну поведінку розчинів у цих течіях.

2 Реологічне моделювання розведених розчинів недеформівних ланцюгових макромолекул

Макромолекули полімерів, розведені розчини яких вивчаються у даній роботі, мають дві суперечливі якості – вони проникні для розчинника і мають високий ступінь жорсткості. До таких макромолекул відносяться макромолекули целюлози та деяких її похідних – тринітроцелюлози, етилцелюлози [4, 5]. Підвищена жорсткість цих макромолекул ґрунтується на специфічних взаємодіях різної природи між сегментами макромолекул [4]. Ці взаємодії призводять до утворення внутрішньо-молекулярної недеформівної мікроструктури.

Співвідношення між характерними розмірами макромолекул полімеру з великою молекулярною вагою і молекул низькомолекулярного розчинника є таким, що взаємодію розчинника із зваженими макромолекулами можна вважати гідродинамічною [4, 5]. Для гідродинамічного моделювання недеформівних ланцюгових макромолекул у даній роботі, як і в роботі [2], використовується недеформівний „складаний метр” – „перлинне намисто” Сайто [6], недеформівне у градієнтних течіях розчину. Ця модель складається із $2n+1$ однакових бусинок – „перлинок”, – які пронумеровані від $-n$ до n , та з'єднані в жорстку систему за допомогою $2n$ сегментів однакової довжини. Модельна макромолекула взаємодіє з розчинником через бусинки, які мають коефіцієнт тертя ξ при їх поступальному русі у розчиннику. Сегменти ж, на кінцях яких бусинки розташовані, з розчинником не взаємодіють.

Модельні макромолекули Сайто характеризуються величинами R, P, Q ($P=Q$) [6]

$$R = \sum_{k=-n}^{+n} \rho_k^2, \quad P = \sum_{k=-n}^{+n} \eta_k^2, \quad Q = \sum_{k=-n}^{+n} \zeta_k^2,$$

де $(\rho_k, \eta_k, \zeta_k)$ – координати k -ої бусинки модельної макромолекули в системі координат $O\rho\eta\zeta$, яка зв'язана з головними осями інерції макромолекули.

Для вивчення у даній роботі реологічної поведінки розведеного розчину таких модельних макромолекул використовується реологічна модель, яка одержана у [2] в рамках структурно-феноменологічного методу [3].

Згідно з [2] обертальна динаміка модельних макромолекул Сайто під дією гідродинамічних сил у довільній градієнтній течії розчину визначається рівнянням

$$\frac{dn_i}{dt} = \omega_{ik} n_k + \Lambda (d_{ik} n_k - d_{km} n_k n_m n_i), \quad (2.1)$$

де $n_i (i = \{x, y, z\})$ – одиничний вектор, який характеризує орієнтацію модельної макромолекули Сайто в лабораторній системі координат $Oxyz$; напрям вектора n_i співпадає з напрямом осі $O\rho$; ω_{ik} – тензор вихору швидкості, $\omega_{ik} = 1/2(v_{i,k} - v_{k,i})$; d_{ij} – тензор швидкостей деформації, $d_{ij} = 1/2(v_{i,j} + v_{j,i})$; v_i – вектор швидкості; t – час; Λ – сталий параметр, який визначає геометрію модельної макромолекули Сайто,

$$\Lambda = \frac{R - P}{R + P}. \quad (2.2)$$

Згідно з припущеннями у [2, 6] на модельну макромолекулу Сайто, крім гідродинамічних сил, діє обертальний броунівський рух. У змаганні дій гідродинамічних сил і дії обертального броунівського руху орієнтація макромолекул визначається ймовірно за допомогою функції розподілу F макромолекул по кутових положеннях, яка є розв'язком рівняння [2]

$$\frac{\partial F}{\partial t} + \frac{\partial}{\partial n_i} \left(F \frac{dn_i}{dt} \right) = D_r \left(\Delta F - 2n_k \frac{\partial F}{\partial n_k} + n_k n_m \frac{\partial^2 F}{\partial n_k \partial n_m} \right). \quad (2.3)$$

У (2.3) Δ – оператор Лапласа; D_r – коефіцієнт обертальної броунівської дифузії модельної макромолекули Сайто [6],

$$D_r = \frac{kT}{\xi(R + P)},$$

де k – стала Больцмана; T – абсолютна температура.

Використання у [2] структурно-феноменологічного методу [3] дозволило отримати у тензорному вигляді реологічне рівняння для напруження T_{ij} , яке виникає у довільних градієнтних течіях розведеного розчину модельних макромолекул Сайто

$$T_{ij} = -p\delta_{ij} + 2\mu_0 \left(1 + n_0 \frac{\xi P}{2\mu_0} \right) d_{ij} + 3n_0 \xi (R - P) D_r \times$$

$$\times \left(\langle n_i n_j \rangle - \frac{1}{3} \delta_{ij} \right) + n_0 \xi \frac{(R-P)^2}{R+P} d_{km} \langle n_k n_m n_i n_j \rangle + \\ + n_0 \xi \frac{P(R-P)}{2(R+P)} (d_{ik} \langle n_k n_j \rangle + d_{jk} \langle n_k n_i \rangle). \quad (2.4)$$

У (2.4) p – тиск; δ_{ij} – символ Кронекера; μ_0 – ньютонівська в'язкість низькомолекулярного розчинника; n_0 – кількість макромолекул розчиненого полімеру в одиниці об'єму розчину; кутові дужки $\langle \rangle$ означають осереднення у просторі орієнтацій вектора n_i за допомогою функції розподілу F , яка є розв'язком рівняння (2.3).

3 Динаміка модельних макромолекул Сайто у течіях одновісного розтягу та простого зсуву

Для вивчення обертальної динаміки макромолекули Сайто у течії одновісного розтягу $v_x = -\frac{q_1}{2}x$, $v_y = -\frac{q_1}{2}y$, $v_z = q_1 z$, $q_1 = \text{const}$, (3.1) та течії простого зсуву

$$v_x = Ky, \quad v_y = 0, \quad v_z = 0, \quad K = \text{const}. \quad (3.2)$$

під дією гідродинамічних сил, які діють на макромолекулу з боку розчинника, у даній роботі використовується визначальне рівняння (2.1), одержане в роботі [2].

У сферичній системі координат (φ, θ) (Рис.1) компоненти одиничного вектора n_i , який характеризує орієнтацію макромолекули Сайто, визначаються співвідношеннями

$$n_x = \cos \varphi \sin \theta, \quad n_y = \sin \varphi \sin \theta, \quad n_z = \cos \theta, \quad (3.3)$$

а визначальне рівняння обертальної динаміки (2.1) окремої макромолекули під дією гідродина-

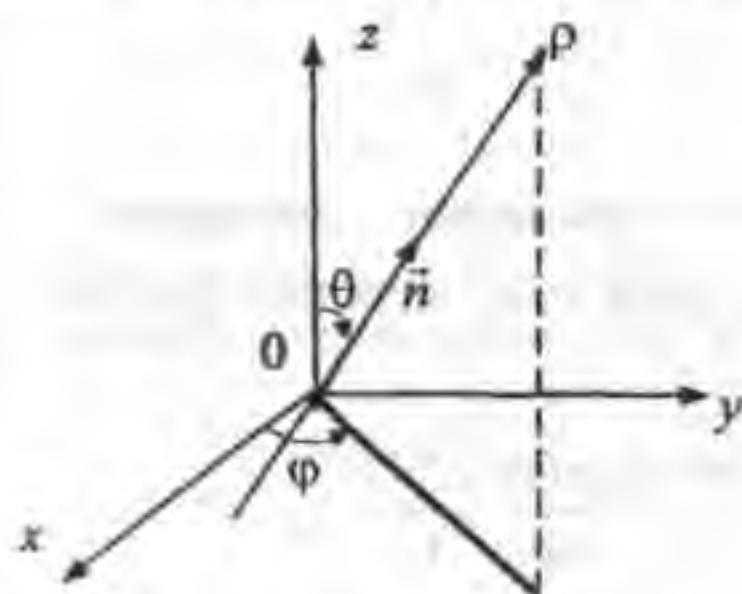


Рис. 1 Лабораторна система координат, початок якої співпадає з центром мас модельної макромолекули

мічних сил у течії (3.1) набирає вигляду

$$\frac{d\varphi}{dt} = 0, \\ \frac{d\theta}{dt} = -\frac{3}{4} \Lambda q_1 \sin 2\theta.$$

Дослідження розв'язків цих рівнянь

$$\varphi = \varphi_0,$$

$$\lg \theta = \lg \theta_0 \exp \left\{ -\frac{3}{2} \Lambda t \right\}$$

показує, що при будь-яких початкових кутових положеннях φ_0, θ_0 макромолекули широтний кут θ прямує до нуля при $t \rightarrow \infty$, тобто макромолекули розчину орієнтуються стаціонарно уздовж осі Oz – осі розтягу у течії (3.1).

Для порівняння одержаного результату розглянемо обертальну динаміку недеформівних макромолекул Сайто під дією гідродинамічних сил у течії простого зсуву (3.2).

Згідно (2.1) з врахуванням заміни (3.3) обертальний рух окремої недеформівної макромолекули Сайто у течії (3.2) розведеного розчину таких макромолекул визначається диференціальними рівняннями

$$\frac{d\varphi}{dt} = \frac{K}{2} (1 + \Lambda \cos 2\varphi), \quad (3.4)$$

$$\frac{d\theta}{dt} = \frac{K}{4} \Lambda \sin 2\varphi \sin 2\theta. \quad (3.5)$$

Розв'язки цих рівнянь мають вигляд

$$\lg \varphi = \sqrt{\frac{R}{P}} \lg \frac{2\pi}{T} t, \\ \lg^2 \theta = \frac{CR}{R \cos^2 \varphi + P \sin^2 \varphi},$$

де

$$T = \frac{2\pi R+P}{K \sqrt{RP}} \quad (3.6)$$

– період обертання макромолекули навколо осі Oz ; C – стала інтегрування, яка повинна визначатися із початкових умов для широтного кута θ .

Отже, на відміну від течії одновісного розтягу (3.1), в якій макромолекули орієнтуються стаціонарно уздовж осі Oz , у простій зсувній

течії (3.2) макромолекули Сайто обертаються періодично навколо осі Oz з періодом (3.6).

Згідно (3.4), (3.5) кутова швидкість макромолекули $\vec{\omega} = \{\dot{\varphi}; \dot{\theta}\}$ залежить від її кутового положення. Тому можна говорити про кінематичну орієнтацію макромолекули під дією гідродинамічних сил у течії (3.2) у кутовому положенні $(\varphi; \theta)$, в якому $\vec{\omega}$ має найменше значення.

Проведені дослідження демонструють принципову різну динаміку зважених макромолекул Сайто у течіях (3.1) і (3.2) їх розведеного розчину.

У змаганні з гідродинамічними силами, які діють на недеформівні макромолекули у градієнтних течіях розчину, обертальний броунівський рух, у якому беруть участь зважені макромолекули, протидіє орієнтуючій дії гідродинамічних сил, але зі збільшенням паралельного градієнта швидкості q_1 течії (3.1) та поперечного градієнта швидкості K течії (3.2) дія гідродинамічних сил на макромолекулу перемагає.

4 Реологічна поведінка розведеного розчину недеформівних макромолекул Сайто у течії одновісного розтягу

Ефективна в'язкість розведеного розчину модельних макромолекул Сайто, яка є єдиною його макрореологічною характеристикою у течії одновісного розтягу (3.1), обчислюється у даній роботі за формулою [3]

$$\mu_a^1 = \frac{T_{ij} d_{ij}}{2d_{km} d_{km}} \quad (4.1)$$

Необхідні при обчисленні μ_a^1 елементи T_{xx}, T_{yy}, T_{zz} тензора напружень T_{ij} отримуються із (2.4) з врахуванням заміни (3.3)

$$\begin{aligned} T_{xx} = & -p - q_1 \mu_0 \left(1 + n_0 \frac{\xi P}{2\mu_0} \right) + 3n_0 \xi (R - P) D_r \times \\ & \times \left(\langle \cos^2 \varphi \sin^2 \theta \rangle - \frac{1}{3} \right) + n_0 \xi \frac{(R - P)^2}{R + P} \times \\ & \times q_1 \langle \cos^2 \varphi \sin^2 \theta (1 - \frac{3}{2} \sin^2 \theta) \rangle - \\ & - n_0 \xi \frac{P(R - P)}{2(R + P)} q_1 \langle \cos^2 \varphi \sin^2 \theta \rangle; \end{aligned} \quad (4.2)$$

$$\begin{aligned} T_{yy} = & -p - q_1 \mu_0 \left(1 + n_0 \frac{\xi P}{2\mu_0} \right) + 3n_0 \xi (R - P) D_r \times \\ & \times \left(\langle \sin^2 \varphi \sin^2 \theta \rangle - \frac{1}{3} \right) + n_0 \xi \frac{(R - P)^2}{R + P} \times \\ & \times q_1 \langle \sin^2 \varphi \sin^2 \theta (1 - \frac{3}{2} \sin^2 \theta) \rangle - \\ & - n_0 \xi \frac{P(R - P)}{2(R + P)} q_1 \langle \sin^2 \varphi \sin^2 \theta \rangle; \end{aligned} \quad (4.3)$$

$$\begin{aligned} T_{zz} = & -p + 2q_1 \mu_0 \left(1 + n_0 \frac{\xi P}{2\mu_0} \right) + 3n_0 \xi (R - P) D_r \times \\ & \times q_1 \langle \cos^2 \theta (1 - \frac{3}{2} \sin^2 \theta) \rangle + \\ & + n_0 \xi \frac{P(R - P)}{R + P} q_1 \langle \cos^2 \theta \rangle. \end{aligned} \quad (4.4)$$

При осередненні у (4.2) – (4.4) використовується функція розподілу орієнтацій макромолекул по кутових положеннях у течії одновісного розтягу (3.1) їх розведеного розчину, яка є розв'язком рівняння

$$\begin{aligned} \frac{d^2 F}{d\theta^2} + \left(\operatorname{ctg} \theta + \frac{3}{4} \alpha_1 \Lambda \sin 2\theta \right) \frac{dF}{d\theta} + \\ + \frac{3}{2} \alpha_1 \Lambda (3 \cos^2 \theta - 1) F = 0, \end{aligned} \quad (4.5)$$

що отримується з (2.3) з урахуванням заміни (3.3). У (4.5) $\alpha_1 = q_1 / D_r$; параметр Λ визначається співвідношенням (2.2); згідно з [6] $\Lambda > 0$.

У даній роботі функція $F(\theta)$ отримується у вигляді подвійного ряду

$$F(\theta) = \sum_{n=0}^{\infty} \sum_{l=0}^{\infty} a_{nl} \left(\alpha_1 \frac{R - P}{R + P} \right)^l P_{2n}(\cos \theta), \quad (4.6)$$

де $P_{2n}(\cos \theta)$ – поліноми Лежандра. Для коефіцієнтів a_{nl} ряду (4.6) знайдено рекурентне співвідношення

$$\begin{aligned} 3n(2n+1)a_{nl} = & \frac{(2n-1)2n(2n+1)}{(4n-3)(4n-1)} a_{n-1,l-1} + \\ & + \left[\frac{(2n+1)^2(2n+3)}{(4n+1)(4n+3)} - \frac{(2n-2)(2n)^2}{(4n-1)(4n+1)} - 1 \right] a_{n,l-1} - \end{aligned}$$

$$\frac{2n(2n+1)(2n+2)}{(4n+3)(4n+5)} a_{n+1, i-1}.$$

При осередненні виразів (4.2) – (4.4) у даній роботі використовуються три перших члена ряду (4.6)

$$F(\theta) = \frac{1}{4\pi} \left\{ 1 + \frac{1}{4} \alpha_1 \Lambda (3 \cos^2 \theta - 1) + \frac{9}{4} (\alpha_1 \Lambda)^2 \times \right. \\ \left. \times \left[\frac{1}{126} (3 \cos^2 \theta - 1) + \frac{1}{280} (35 \cos^4 \theta - 30 \cos^2 \theta + 3) \right] \right\}.$$

Виконання осереднення за формулою

$$\langle a(\varphi, \theta) \rangle = \frac{1}{4\pi} \int_0^{2\pi} \int_0^\pi a(\varphi, \theta) F(\theta) \sin \theta d\theta d\varphi,$$

де $a(\varphi, \theta)$ – осереднювані величини у виразах (4.2) – (4.4) і використання отриманих результатів у (4.1) дозволяє знайти ефективну в'язкість $\mu_a^{\parallel}$ розведеного розчину модельних макромолекул Сайто у течії одновісного розтягу (3.1)

$$\mu_a^{\parallel} = \frac{T_y d_y}{2 d_{km} d_{km}} = 3\mu_0 + \frac{3}{2} n_0 \xi P + n_0 \xi (R - P) \times \\ \times \left\{ \frac{83R - 63P}{80(R + P)} + \frac{399R^2 - 952RP + 553P^2}{2240(R + P)^2} \alpha_1 - \right. \\ \left. - \frac{R^3 - 5R^2P + 7RP^2 - 3P^3}{560(R + P)^3} \alpha_1^2 \right\}. \quad (4.7)$$

Вивчаючи реологічну поведінку розведеного розчину недеформівних ланцюгових макромолекул у течії простого зсуву (3.2), Сайто помітив [6], що динаміка зважених макромолекул під дією гідродинамічних сил у цій течії розчину при відповідному виборі параметра Λ (2.2) нагадує динаміку непротічного еліпсоїда обертання, який використовується у [7] як гідродинамічна модель непротічних для розчинника недеформівних ланцюгових макромолекул. У роботі [2] встановлено параметри такого еліпсоїда. Так при

$$R = \frac{2}{3} \frac{\pi \alpha^3 \mu_0 g(p_0)}{\xi} \left(1 + \frac{p_0^2 - 1}{p_0^2 + 1} \right), \quad (4.8)$$

$$P = \frac{2}{3} \frac{\pi \alpha^3 \mu_0 g(p_0)}{\xi} \left(1 - \frac{p_0^2 - 1}{p_0^2 + 1} \right), \quad (4.9)$$

$$\text{де } g(p_0) = \frac{p_0^4 - 1}{p_0^4 \left(\frac{2p_0^2 - 1}{2p_0 \sqrt{p_0^2 - 1}} \ln \left(\frac{p_0 + \sqrt{p_0^2 - 1}}{p_0 - \sqrt{p_0^2 - 1}} \right) - 1 \right)};$$

$p_0 = a/b$; a і b – півосі еквівалентного еліпсоїда обертання, параметр Λ (2.2) набирає вигляду

$$\Lambda = \frac{p_0^2 - 1}{p_0^2 + 1}, \quad (4.10)$$

а рівняння обертальної динаміки (2.1) модельної макромолекули Сайто під дією гідродинамічних сил співпадає з відповідним рівнянням для зваженого модельного еліпсоїда обертання

$$\frac{dn_i}{dt} = \omega_{ik} n_k + \frac{p_0^2 - 1}{p_0^2 + 1} (d_{ik} n_k - d_{km} n_k n_m n_i).$$

Визначення мікрохарактеристик R і P модельних макромолекул Сайто у (4.8), (4.9) через параметри, які характеризують геометрію еквівалентного еліпсоїда обертання та його взаємодію з ньютонівським розчинником, дозволяє отримати придатний для розрахунків вираз для ефективної в'язкості $\mu_a^{\parallel}$ (4.7) у вигляді

$$\mu_a^{\parallel} = 3\mu_0 + \frac{3}{2} \mu_0 \phi p_0^2 g(p_0) (1 - \Lambda) + \frac{1}{4} \mu_0 \phi p_0^2 g(p_0) \Lambda \times \\ \times \left\{ \frac{73\Lambda + 10}{20} + \frac{476\Lambda - 77}{140} \Lambda \alpha_1 - \frac{16\Lambda - 8}{70} \Lambda^2 \alpha_1^2 \right\}, \quad (4.11)$$

де Λ – параметр, який визначається формулою (4.10); ϕ – об'ємна концентрація модельних макромолекул – еліпсоїдів обертання – у розчині.

На Рис. 2 наведена залежність від α_1 приведеного інкременту

$$[\mu_a^{\parallel}] = \frac{\mu_a^{\parallel} - \mu_a^{\parallel}(0)}{\mu_0 \phi} \quad (4.12)$$

ефективної в'язкості $\mu_a^{\parallel}$ розведеного розчину модельних ланцюгових макромолекул Сайто у течії одновісного розтягу (3.1); $\mu_a^{\parallel}(0)$ у (4.12) визначається з (4.11) при $\alpha_1 = 0$.

Одержані результати показують, що ефективна в'язкість розведеного розчину недеформівних макромолекул Сайто суттєво залежить від геометрії течії розчину. Згідно з [2] розведений розчин таких макромолекул виявляє

псевдопластичне зменшення ефективної в'язкості $\mu_o^\perp$ у течії простого зсуву (3.2) при збільшенні градієнта швидкості K , а в течії одновісного розтягу (3.1) ефективна в'язкість розчину $\mu_a^\parallel$ зростає зі збільшенням градієнта швидкості q_1 цієї течії (Рис. 2). Така відмінність макрореологічної поведінки розведеного розчину недеформівних ланцюгових макромолекул у течіях (3.1), (3.2) є наслідком різної динаміки макромолекул у цих течіях.

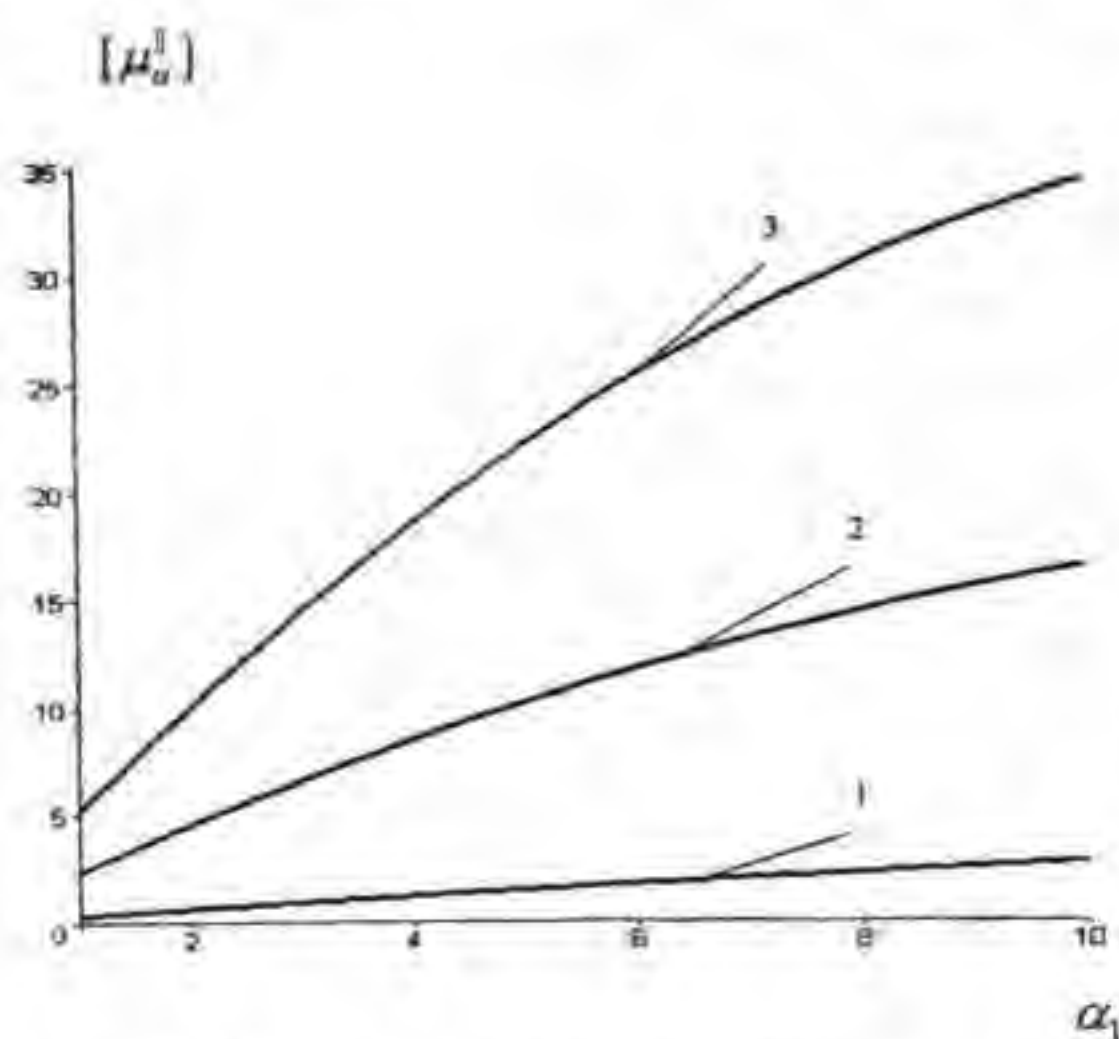


Рис. 2. Залежність $[\mu_a^\parallel]$ розведеного розчину модельних макромолекул Сайто від $\alpha_1 = q_1 / D_r$; криві 1,2,3 відповідають значенням $p_0 = 2, 4, 6$

Висновки

1. Вивчення течій одновісного розтягу (3.1) і простого зсуву (3.2) розведеного розчину недеформівних ланцюгових макромолекул дозволило виявити різну динаміку зв'язаних макромолекул під дією гідродинамічних сил, які діють на молекули з боку низькомолекулярного розчинника. Такі молекули стаціонарно орієнтуються уздовж осі розтягу у течії (3.1) і здійснюють періодичний обертальний рух у течії (3.2) розчину.

2. Наслідком різної динаміки зв'язаних макромолекул у течіях (3.1) і (3.2) є різна макрореологічна поведінка їх розведених розчинів у цих течіях. Виявлено, що ефективна в'язкість $\mu_a^\parallel$ розведених розчинів недеформівних ланцюгових макромолекул Сайто у течії одновісного розтягу (3.1) збільшується зі збільшенням паралельного градієнта швидкості

q_1 течії, тоді, як згідно [2], ефективна в'язкість $\mu_a^\perp$ таких розчинів у течії (3.2) зменшується зі збільшенням поперечного градієнта швидкості K цієї течії.

3. Дослідження, які проведені у даній роботі, стали можливими завдяки використанню структурно-феноменологічних реологічних рівнянь розведених розчинів недеформівних ланцюгових макромолекул Сайто, одержаних у роботі [2]. Ці визначальні рівняння поєднують макрореологічні властивості розчинів з мікроструктурними характеристиками зв'язаних макромолекул і особливостями їх взаємодії з розчинниками у градієнтних течіях розчинів.

Список використаних джерел

1. Эйрих Ф. Р. Реология. Теория и приложения / под ред. Эйриха Ф. Р. – Москва: ИЛ, 1962. – 824 с.
2. Таран Є. Ю. Структурно-феноменологічна реологія розведених розчинів недеформівних вільнопротічних ланцюгових макромолекул / Таран Є. Ю., Кондрат Р. Я. // Вісник Київського університету. Серія : фіз.-мат. науки. – 2009. – Вип. 3. – С. 97 – 102.
3. Таран Е. Ю. Структурно-феноменологическая реология разбавленных суспензий: дис. ... доктора физ.-мат. наук : спец. 01.02.05 / Таран Евгений Юрьевич. – Харьков, 1994. – 346 с.
4. Цветков В. Н. Структура макромолекул в растворах / Цветков В. Н., Эскин В. Е., Френкель С. Я. – Москва: Наука, 1964. – 720 с.
5. Моравец Г. Макромолекулы в растворе / Моравец Г. – Москва: Мир, 1967. – 398 с.
6. Saito N. The effect of the Brownian motion on the viscosity of solutions of macromolecules. II. Rigid chain molecules / Saito N. // J. Phys. Soc. Japan. – 1951. – Vol. 6, № 2. – P. 302 – 314.
7. Saito N. The effect of the Brownian motion on the viscosity of solutions of macromolecules. I. Ellipsoid of revolution / Saito N. // J. Phys. Soc. Japan. – 1951. – Vol. 6, № 2. – P. 297 – 302.

Надійшла до редколегії 10.02.2011

КОМП'ЮТЕРНІ НАУКИ ТА ІНФОРМАТИКА

**(The Proceedings of International
Conference "Theoretical and Applied
Aspects of Program Systems
Development")**

THE J. H. HARRIS FOUNDATION
AT
TAMMAMBOH

THE J. H. HARRIS FOUNDATION

AT TAMMAMBOH

AT TAMMAMBOH

AT TAMMAMBOH

УДК 517.929.4

Діблік Й.^{*}, д. ф.-м. н.,
Кухаренко О.В., аспірант,
Хусаїнов Д.Я., д. ф.-м. н., проф.

Зображення розв'язку першої крайової задачі системи рівнянь із сталим запізнюванням

У даній роботі розглянуто першу крайову задачу для системи лінійних стаціонарних рівнянь в частинних похідних другого порядку з сталим запізнюванням. Знайдено аналітичний розв'язок системи у вигляді ряду, для випадку, коли матриці коефіцієнтів системи комутативні, а їх власні числа дійсні й різні.

Ключові слова: система рівнянь в частинних похідних, перша крайова задача, запізнення аргументу.

^{*} Was supported by the grant pP201/11/0768 of the Czech Grant Agency (Pradue) and by the Council of Czech Government MSM 0021630503

E-mail: akukharenko@ukr.net, diblic@feec.vutbr.cz, khusainov@unicyb.kiev.ua
Статтю представив д.т.н., проф. Гаращенко Ф.Г.

Вступ

У запропонованій роботі розглянуті системи лінійних стаціонарних диференціальних рівнянь другого порядку з сталим запізнюванням

$$\begin{cases} \frac{\partial u(x,t)}{\partial t} = a_{11} \frac{\partial^2 u(x,t-\tau)}{\partial x^2} + a_{12} \frac{\partial^2 v(x,t-\tau)}{\partial x^2} + b_{11}u(x,t) + b_{12}v(x,t), \\ \frac{\partial v(x,t)}{\partial t} = a_{21} \frac{\partial^2 u(x,t-\tau)}{\partial x^2} + a_{22} \frac{\partial^2 v(x,t-\tau)}{\partial x^2} + b_{21}u(x,t) + b_{22}v(x,t) \end{cases} \quad (0.1)$$

Передбачається, що матриці $A = \{a_{ij}\}$, $B = \{b_{ij}\}$, $i, j = \overline{1,2}$ задовольняють умові комутативності. Отримано розв'язок першої крайової задачі для випадку, коли власні числа цих матриць дійсні та різні.

Система із запізнюванням

Розглянемо систему виду (0.1), визначену при $0 \leq x \leq l$, $t \geq -\tau$. Початкові і крайові умови мають вигляд

$$u(x,t) = \varphi(x,t), \quad v(x,t) = \psi(x,t), \quad (1.1)$$

$$0 \leq x \leq l, \quad -\tau \leq t \leq 0,$$

$$\begin{aligned} u(0,t) &= \mu_1(t), \quad u(l,t) = \mu_2(t), \quad v(0,t) = \theta_1(t), \\ v(l,t) &= \theta_2(t), \quad t \geq -\tau. \end{aligned} \quad (1.2)$$

Крім того, виконані «умови узгодженості»

J. Diblic, Dr.Sc.,
O.V. Kukharenko, PhD,
D.Ya. Khusainov, full prof., Dr.Sc.

Representation of solution of the first boundary value problem for delay systems

The first boundary value problem for a system of linear stationary delay partial differential equations of the second order has been solved. A solution is presented in an analytical form of formal series for the case, when matrixes of coefficients are commutative and their eigenvalues are real and different.

Key Words: partial differential system, first boundary value problem, time delay argument

$$\begin{aligned} \mu_1(t) &= \varphi(0,t), \quad \mu_2(t) = \varphi(l,t), \quad \theta_1(t) = \psi(0,t), \\ \theta_2(t) &= \psi(l,t), \quad -\tau \leq t \leq 0. \end{aligned}$$

Якщо матриці A і B комутативні, то існує неособлива матриця S , що одночасно зводить ці матриці до жорданової форми [1]. Тому перетворенням

$$\begin{pmatrix} u(x,t) \\ v(x,t) \end{pmatrix} = S \begin{pmatrix} \xi(x,t) \\ \eta(x,t) \end{pmatrix}, \quad S = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix}$$

систему (0.1) можна звести до виду

$$\begin{pmatrix} \frac{\partial \xi(x,t)}{\partial t} \\ \frac{\partial \eta(x,t)}{\partial t} \end{pmatrix} = \Lambda_1 \begin{pmatrix} \frac{\partial^2 \xi(x,t-\tau)}{\partial x^2} \\ \frac{\partial^2 \eta(x,t-\tau)}{\partial x^2} \end{pmatrix} + \Lambda_2 \begin{pmatrix} \xi(x,t) \\ \eta(x,t) \end{pmatrix}, \quad (1.3)$$

де Λ_1 і Λ_2 – жорданові форми матриць A і B відповідно.

Одержимо розв'язок першої крайової задачі, для випадку, коли корені λ_1 , λ_2 , ζ_1 , ζ_2 характеристичних рівнянь матриць A і B дійсні й різні. Тоді після лінійного перетворення

система розщеплюється на два незалежних рівняння

$$\begin{aligned}\frac{\partial \xi(x,t)}{\partial t} &= \lambda_1 \frac{\partial^2 \xi(x,t-\tau)}{\partial x^2} + \varsigma_1 \xi(x,t), \\ \frac{\partial \eta(x,t)}{\partial t} &= \lambda_2 \frac{\partial^2 \eta(x,t-\tau)}{\partial x^2} + \varsigma_2 \eta(x,t).\end{aligned}\quad (1.4)$$

Початкові й крайові умови при цьому будуть мати вигляд

$$\begin{aligned}\xi(0,t) &= \bar{\mu}_1(t), \quad \xi(l,t) = \bar{\mu}_2(t), \quad \eta(0,t) = \bar{\theta}_1(t), \\ \eta(l,t) &= \bar{\theta}_2(t), \quad t \geq -\tau, \\ \xi(x,t) &= \bar{\varphi}(x,t), \quad \eta(x,t) = \bar{\psi}(x,t), \quad 0 \leq x \leq l, \\ &\quad -\tau \leq t \leq 0, \\ \begin{pmatrix} \bar{\mu}_1(t) \\ \bar{\theta}_1(t) \end{pmatrix} &= S^{-1} \begin{pmatrix} \mu_1(t) \\ \theta_1(t) \end{pmatrix}, \quad \begin{pmatrix} \bar{\mu}_2(t) \\ \bar{\theta}_2(t) \end{pmatrix} = S^{-1} \begin{pmatrix} \mu_2(t) \\ \theta_2(t) \end{pmatrix}, \\ \begin{pmatrix} \bar{\varphi}(t) \\ \bar{\psi}(t) \end{pmatrix} &= S^{-1} \begin{pmatrix} \varphi(t) \\ \psi(t) \end{pmatrix}.\end{aligned}$$

Розглянемо перше рівняння

$$\frac{\partial \xi(x,t)}{\partial t} = \lambda_1 \frac{\partial^2 \xi(x,t-\tau)}{\partial x^2} + \varsigma_1 \xi(x,t) \quad (1.5)$$

з початковими й крайовими умовами виду

$$\begin{aligned}\xi(0,t) &= \bar{\mu}_1(t), \quad \xi(l,t) = \bar{\mu}_2(t), \quad t \geq -\tau, \\ \xi(x,t) &= \bar{\varphi}(x,t), \quad 0 \leq x \leq l, \quad -\tau \leq t \leq 0.\end{aligned}$$

Розв'язок шукаємо у вигляді суми

$$\begin{aligned}\xi(x,t) &= \xi_0(x,t) + \xi_1(x,t) + \bar{\mu}_1(t) + \\ &\quad + \frac{x}{l} [\bar{\mu}_2(t) - \bar{\mu}_1(t)],\end{aligned}\quad (1.6)$$

• $\xi_0(x,t)$ – розв'язок однорідного рівняння

$$\frac{\partial \xi(x,t)}{\partial t} = \lambda_1 \frac{\partial^2 \xi(x,t-\tau)}{\partial x^2} + \varsigma_1 \xi(x,t), \quad (1.7)$$

з нульовими крайовими $\xi(0,t) = 0$, $\xi(l,t) = 0$ і

ненульовими початковими умовами $\xi(x,t) = \Phi(x,t)$, $-\tau \leq t \leq 0$, $0 \leq x \leq l$,

$$\Phi(x,t) = \bar{\varphi}(x,t) - \bar{\mu}_1(t) - \frac{x}{l} [\bar{\mu}_2(t) - \bar{\mu}_1(t)];$$

• $\xi_1(x,t)$ – розв'язок неоднорідного рівняння

$$\frac{\partial \xi(x,t)}{\partial t} = \lambda_1 \frac{\partial^2 \xi(x,t-\tau)}{\partial x^2} + \varsigma_1 \xi(x,t) + F(x,t),$$

$$\begin{aligned}F(x,t) &= \varsigma_1 \left\{ \bar{\mu}_1(t) + \frac{x}{l} [\bar{\mu}_2(t) - \bar{\mu}_1(t)] \right\} - \\ &\quad - \dot{\bar{\mu}}_1(t) - \frac{x}{l} \left[\dot{\bar{\mu}}_2(t) - \dot{\bar{\mu}}_1(t) \right]\end{aligned}\quad (1.8)$$

з нульовими крайовими $\xi(0,t) = 0$, $\xi(l,t) = 0$, $t \geq -\tau$ і нульовими початковими умовами $\xi(x,t) = 0$, $-\tau \leq t \leq 0$, $0 \leq x \leq l$.

Однорідне рівняння. Для знаходження розв'язку $\xi_0(x,t)$ будемо використовувати метод розділення змінних, тобто будемо шукати розв'язок у вигляді добутку двох функцій $\xi_0(x,t) = X(x)T(t)$. Після підстановки в рівняння маємо

$$X(x)T'(t) = \lambda_1 X''(x)T(t-\tau) + \varsigma_1 X(x)T(t).$$

Розділивши змінні, одержимо

$$\frac{T'(t) - \varsigma_1 T(t)}{\lambda_1 T(t-\tau)} = \frac{X''(x)}{X(x)} = -k^2.$$

Розіб'ємо даний вираз на два рівняння

$$\begin{aligned}T'(t) - \varsigma_1 T(t) + \lambda_1 k^2 T(t-\tau) &= 0, \\ X''(x) + k^2 X(x) &= 0.\end{aligned}\quad (1.9)$$

Розв'язки другого з рівнянь (1.9), що задовольняють умовам $X(0) = 0$, $X(l) = 0$, мають

$$\text{вигляд } X_n(x) = A_n \sin \frac{\pi n}{l} x, \quad k_n^2 = \left(\frac{\pi n}{l} \right)^2, \quad n = 1, 2, \dots$$

Розглянемо перше з рівнянь (1.9)

$$T'_n(t) = \varsigma_1 T_n(t) - \lambda_1 \left(\frac{\pi n}{l} \right)^2 T_n(t-\tau). \quad (1.10)$$

Для одержання початкових умов розкладемо функцію $\Phi(x,t)$ в ряд по розв'язкам другого рівняння

$$\Phi(x,t) = \sum_{n=1}^{\infty} \Phi_n(t) \sin \frac{\pi n}{l} x, \quad (1.11)$$

$$\begin{aligned}\Phi_n(t) &= \frac{2}{l} \int_0^l \bar{\varphi}(s,t) \sin \frac{\pi n}{l} s ds + \\ &\quad + \frac{2}{\pi n} [(-1)^n \bar{\mu}_2(t) - \bar{\mu}_1(t)], \quad n = 1, 2, \dots\end{aligned}$$

Одержимо розв'язок задачі Коші для кожного з рівнянь (1.10) з умовами (1.11) в аналітичному вигляді.

Попередньо розглянемо лінійне однорідне рівняння із чистим сталим запізнюванням виду [2]

$$\dot{x}(t) = bx(t - \tau) \quad (1.12)$$

і початковою умовою $x(t) = \beta(t)$, $-\tau \leq t \leq 0$.

Означення 1. [3] Запізнюючимся експоненціалом $\exp_{\tau}\{b, t\}$ назвемо функцію, що має вигляд

$$\exp_{\tau}\{b, t\} = \begin{cases} 0, & \text{якщо } -\infty < t < -\tau, \\ 1, & \text{якщо } -\tau \leq t < 0, \\ 1 + b \frac{t}{1!}, & \text{якщо } 0 \leq t < \tau, \\ \dots & \dots \\ 1 + b \frac{t}{1!} + b^2 \frac{(t - \tau)^2}{2!} + \dots + \\ + b^k \frac{[t - (k - 1)\tau]^k}{k!}, & \text{якщо } (k - 1)\tau \leq t < k\tau. \end{cases}$$

полінома степеня k на проміжках $(k - 1)\tau < t \leq k\tau$, «склеєного» у вузлах $t = k\tau$, $k = 0, 1, 2, \dots$.

Теорема 1. Розв'язок рівняння (1.12), що задовольняє початковим умовам $x(t) = \beta(t)$, $-\tau \leq t \leq 0$, може бути представлений у вигляді

$$x(t) = \exp_{\tau}\{b, t\} \beta(-\tau) + \int_{-\tau}^0 \exp_{\tau}\{b, t - \tau - s\} \beta'(s) ds.$$

Далі розглянемо диференціальне рівняння

$$\dot{x}(t) = ax(t) + bx(t - \tau) \quad (1.13)$$

з початковою умовою $x(t) = \beta(t)$, $-\tau \leq t \leq 0$ [4].

Теорема 2. Розв'язок рівняння (1.13), що задовольняє початковим умовам $x(t) = \beta(t)$, $-\tau \leq t \leq 0$, може бути представлений у вигляді

$$x(t) = \exp_{\tau}\{b_1, t\} e^{a(t+\tau)} \beta(-\tau) + \int_{-\tau}^0 \exp_{\tau}\{b_1, t - \tau - s\} \times \\ \times e^{a(t-s)} [\beta'(s) - a\beta(s)] ds, \quad b_1 = be^{-a\tau}.$$

На підставі наведених теорем можна сформулювати твердження, що розв'язок $\xi_0(x, t)$ однорідного рівняння (1.7), який задовольняє

нульовим крайовим $\xi(0, t) = 0$, $\xi(l, t) = 0$ і ненульовим початковим умовам $\xi(x, t) = \Phi(x, t)$, $-\tau \leq t \leq 0$, $0 \leq x \leq l$ має вигляд

$$\xi_0(x, t) = \sum_{n=1}^{\infty} \left\{ \exp_{\tau} \left\{ -\lambda_1 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_1 \tau}, t \right\} e^{\zeta_1(t+\tau)} \times \right. \\ \times \Phi_n(-\tau) + \int_{-\tau}^0 \exp_{\tau} \left\{ -\lambda_1 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_1 \tau}, t - \tau - s \right\} \times \\ \times e^{\zeta_1(t-s)} [\Phi'_n(s) - \zeta_1 \Phi_n(s)] ds \Big\} \sin \frac{\pi n}{l} x, \quad n = 1, 2, \dots, \\ \Phi_n(t) = \frac{2}{l} \int_0^l \varphi(s, t) \sin \frac{\pi n}{l} s ds + \frac{2}{\pi n} [(-1)^n \bar{\mu}_2(t) - \bar{\mu}_1(t)].$$

Неоднорідне рівняння. Далі розглянемо неоднорідне рівняння

$$\frac{\partial \xi(x, t)}{\partial t} = \lambda_1 \frac{\partial^2 \xi(x, t - \tau)}{\partial x^2} + \zeta_1 \xi(x, t) + F(x, t)$$

з нульовими крайовими $\xi(0, t) = 0$, $\xi(l, t) = 0$, $t \geq -\tau$ і нульовими початковими умовами $\xi(x, t) = 0$, $-\tau \leq t \leq 0$, $0 \leq x \leq l$. Розв'язок будемо шукати у вигляді розкладу по власних функціях попередньої задачі, тобто у вигляді ряду

$$\xi_1(x, t) = \sum_{n=1}^{\infty} T_n(t) \sin \frac{\pi n}{l} x.$$

Підставивши ряд у рівняння (1.8) і прирівнявши коефіцієнти при однакових членах, одержимо систему рівнянь

$$T'_n(t) = \zeta_1 T_n(t) - \lambda_1 \left(\frac{\pi n}{l} \right)^2 T_n(t - \tau) + f_n(t), \quad (1.14) \\ f_n(t) = \frac{2}{l} \int_0^l F(s, t) \sin \frac{\pi n}{l} s ds = \frac{2}{\pi n} \left[\zeta_1 \left((-1)^{n+1} \bar{\mu}_2(t) + \right. \right. \\ \left. \left. + \bar{\mu}_1(t) \right) - \left((-1)^{n+1} \dot{\bar{\mu}}_2(t) + \dot{\bar{\mu}}_1(t) \right) \right], \quad n = 1, 2, \dots$$

Попередньо розглянемо рівняння

$$\dot{x}(t) = ax(t) + bx(t - \tau) + f(t). \quad (1.15)$$

Як слідує з [4] має місце наступне твердження.

Теорема 3. Розв'язок неоднорідного рівняння (1.15), що задовольняє нульовим

початковим умовам $x(t) \equiv 0$, $-\tau \leq t \leq 0$ має вигляд

$$x(t) = \int_0^t \exp_r \{ b e^{-a\tau}, t - \tau - s \} e^{a(t-s)} f(s) ds. \quad (1.16)$$

Звідси випливає, що розв'язок кожного з рівнянь (1.14) можна записати у вигляді

$$T_n(t) = \int_0^t \exp_r \{ r_1, t - \tau - s \} e^{\zeta_1(t-s)} f_n(s) ds, \quad (1.17)$$

$$r_1 = -\lambda_1 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_1 \tau}, \quad n = 1, 2, \dots$$

Отже, розв'язок неоднорідного рівняння (1.8) з нульовими крайовими $\xi(0, t) = 0$, $\xi(l, t) = 0$, $t \geq -\tau$ і нульовими початковими умовами $\xi(x, t) = 0$, $-\tau \leq t \leq 0$, $0 \leq x \leq l$ має вигляд

$$\xi_1(x, t) = \sum_{n=1}^{\infty} \left\{ \int_0^t \exp_r \left\{ -\lambda_1 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_1 \tau}, t - \tau - s \right\} \times \right. \\ \left. \times e^{\zeta_1(t-s)} f_n(s) ds \right\} \sin \frac{\pi n}{l} x.$$

Розв'язок першої крайової задачі

Використовуючи всі попередні результати, розв'язок першої крайової задачі для рівняння (1.5) запишемо у вигляді

$$\xi(x, t) = \sum_{n=1}^{\infty} \left\{ \exp_r \left\{ -\lambda_1 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_1 \tau}, t \right\} e^{\zeta_1(t+\tau)} \times \right. \\ \left. \times \Phi_n(-\tau) + \int_{-\tau}^0 \exp_r \left\{ -\lambda_1 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_1 \tau}, t - \tau - s \right\} \times \right. \\ \left. \times e^{\zeta_1(t-s)} [\Phi'_n(s) - \zeta_1 \Phi_n(s)] ds + \right. \\ \left. + \int_0^t \exp_r \left\{ -\lambda_1 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_1 \tau}, t - \tau - s \right\} \times \right. \\ \left. \times e^{\zeta_1(t-s)} f_n(s) ds \right\} \sin \frac{\pi n}{l} x + \bar{\mu}_1(t) + \frac{x}{l} [\bar{\mu}_2(t) - \bar{\mu}_1(t)], \quad (1.18)$$

Список використаних джерел

1. Гантмахер Ф.Р. Теория матриц. – М.: Наука, 1966. – 276 с.
2. Эльсгольц Л.Э., Норкин С.Б. Введение в теорию дифференциальных уравнений с отклоняющимся аргументом. – М., Наука, 1970. – 270 с.
3. Хусаинов Д.Я., Шуклин Г.В. Об относительной управляемости в системах с чистым запаз-

$$\Phi_n(t) = \frac{2}{l} \int_0^t \bar{\varphi}(s, t) \sin \frac{\pi n}{l} s ds + \frac{2}{\pi n} [(-1)^n \bar{\mu}_2(t) - \bar{\mu}_1(t)], \\ f_n(t) = \frac{2}{\pi n} \left[\zeta_1 [(-1)^{n+1} \bar{\mu}_2(t) + \bar{\mu}_1(t)] - \right. \\ \left. - \left[(-1)^{n+1} \dot{\bar{\mu}}_2(t) + \dot{\bar{\mu}}_1(t) \right] \right], \quad n = 1, 2, \dots$$

Аналогічно друге рівняння (1.4) має розв'язок виду

$$\eta(x, t) = \sum_{n=1}^{\infty} \left\{ \exp_r \left\{ -\lambda_2 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_2 \tau}, t \right\} e^{\zeta_2(t+\tau)} \times \right. \\ \left. \times \Psi_n(-\tau) + \int_{-\tau}^0 \exp_r \left\{ -\lambda_2 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_2 \tau}, t - \tau - s \right\} \times \right. \\ \left. \times e^{\zeta_2(t-s)} [\Psi'_n(s) - \zeta_2 \Psi_n(s)] ds + \right. \\ \left. + \int_0^t \exp_r \left\{ -\lambda_2 \left(\frac{\pi n}{l} \right)^2 e^{-\zeta_2 \tau}, t - \tau - s \right\} \times \right. \\ \left. \times e^{\zeta_2(t-s)} g_n(s) ds \right\} \sin \frac{\pi n}{l} x + \bar{\theta}_1(t) + \frac{x}{l} [\bar{\theta}_2(t) - \bar{\theta}_1(t)], \\ \Psi_n(t) = \frac{2}{l} \int_0^t \bar{\psi}(s, t) \sin \frac{\pi n}{l} s ds + \frac{2}{\pi n} [(-1)^n \bar{\theta}_2(t) - \bar{\theta}_1(t)], \\ g_n(t) = \frac{2}{\pi n} \left[\zeta_2 [(-1)^{n+1} \bar{\theta}_2(t) + \bar{\theta}_1(t)] - \right. \\ \left. - \left[(-1)^{n+1} \dot{\bar{\theta}}_2(t) + \dot{\bar{\theta}}_1(t) \right] \right], \quad n = 1, 2, \dots \quad (1.19)$$

Тоді розв'язок крайової задачі вихідної системи (0.1), що задовольняє умовам (1.1), (1.2) остаточно має вигляд

$$u(x, t) = s_{11} \xi(x, t) + s_{12} \eta(x, t), \\ v(x, t) = s_{21} \xi(x, t) + s_{22} \eta(x, t),$$

де розв'язки $\xi(x, t)$, $\eta(x, t)$ перетвореної системи (1.4), визначені в (1.18), (1.19).

дыванием // Прикладная механика, Т.41, в.2, 2005. – С.118-130.

4. Khusainov D.Ya., Shuklin G.V. Linear Autonomous Time-Delay System with Permutation Matrices Solving // Studies of the University of Zilina, Mathematical Series, Vol. 17, 2003. – pp. 101-108.

Надійшла до редколегії 11.12.2010

УДК 81:519.25

Тарануха В.Ю., асист., Порхун О.В., к.ф.-м.н.,
асист.

Автоматичне встановлення авторства текстів з використанням аналізу звукової організації мови

В статті розглядаються методи визначення авторства текстів за допомогою аналізу послідовностей літер та звуків. Проаналізовано якості ймовірнісної та косинусної міри близькості. Виділено нову ознаку авторства – збіги кількох приголосних чи голосних на стиках слів.

Ключові слова: визначення авторства, ознаки авторства, косинусна міра близькості.

V.Y. Taranukha, E.V. Porkhun, PhD.

Automatic definition of authorship using analysis of letter and phoneme sequences in language

Methods of definition of authorship using character and sound sequences analysis are investigated. Cosine and probabilistic proximity measures are evaluated. New authorship feature based on hiatus and consonants on word borders.

Key Words: definition of authorship, letter sequence, phoneme sequences, cosine proximity measure

*E-mail: elena_porkhun@mail.ru, taranukha@ukr.net
Статтю представив д.ф.-м.н., проф. Анісімов А.В.

При вирішенні задач атрибуції текстів застосовування методів побудови моделі тексту на основі використання статистичних характеристик вимагає попередньої обробки, що полягає у зведенні слів до канонічної форми. В свою чергу, це пов'язано з процедурами морфологічного аналізу. В попередніх дослідженнях [1], зокрема при вирішенні задачі атрибуції художніх творів російських письменників, було виявлено, що процес створення частотних словників із застосуванням морфологічного аналізу для великих корпусів текстів супроводжується значними часовими витратами та певним відсотком похибки розпізнавання змістовної тотожності та близькості слів, зумовленого багатою системою словозміни та словотворення, що властива слов'янським мовам.

У зв'язку з цим заслуговують уваги статистичні методи виділення інформативних елементів текстів, які використовують частотні характеристики різних фрагментів тексту. Одна із переваг такого підходу полягає в тому, що він може застосовуватися для будь-якої природної мови і не потребує участі спеціалістів у конкретній мові. Іншим важливим фактором є те, що при використанні фрагментів постійної довжини можна встановити максимальну кількість фрагментів для даної мови. Крім того, даний метод не потребує додаткової лінгвістичної обробки.

Даний підхід базується на представленні тексту у вигляді поліграмної моделі. У поліграмній моделі зі ступенем n та основою M текст представляється вектором $\{f_i\}, i=1...M^n$, де f_i – частота появи i -ої n -грами в тексті. N -грама є послідовністю n символів виду: $a_1, ..., a_{n-1}, a_n$, причому символи a_i належать алфавіту, розмірність якого збігається з M . Наприклад, для російської мови кількість n -грам при $n=3$ в алфавіті з літер та пробілу дорівнює 35937. Однак, на практиці в реальних текстах реалізується не більше 25-30% n -грам від їх допустимого числа. Так для російської мови їх не більше 10000.

Використання n -грам дозволяє проаналізувати звукову організацію мови – її фоніку. Звуковий матеріал, який властивий художній мові, задається самою мовою; він завжди обмежений (наприклад, в російській мові всього 41 фонема), тому випадкове повторення тих чи інших звукових явищ виникає в будь-якій мові. Художня мова автора впорядковує ці повтори, використовуючи їх для впливу на читача, що є характерною ознакою його стилю. При фонетичному аналізі виявляється звукова складова тексту у співвідношенні з особливостями мови у творах певної епохи, у творчості письменників певного літературного напрямку, в інших творах автора тексту, що досліджується.

Отже, частота появи *n*-грамів в тексті несе важливу інформацію про його властивості. Зокрема, розробником системи «Лінгвоаналізатор» Хмельовим Д. В. було показано, що задачу визначення авторства тексту можна вирішити за допомогою залучення інформації про частоти пар літер, що послідовно зустрічаються в тексті [2,3].

Трансформація тексту

Перед виділенням визначених вище *n*-грам та обчисленням їх частот, була проведена трансформація тексту, що полягає у записі звуків мови тексту з врахуванням їх позицій. Обрано зважений підхід, що взяв найбільш зручні елементи від Московської та Ленінградської (Петербурзької) фонологічних шкіл. Такий підхід зумовлений намаганням спростити аналіз та перетворення, особливо при обробці наголосів.

Зокрема, були зроблені заміни літер *я, е, ё, ю* на відповідні звуки, які вони позначають. А саме, після голосних літер, на початку слова і після роздільних *ъ* та *ь* ці літери позначають два звуки – приголосний [й] та відповідний голосний:

Я – [йа], *е* – [йэ], *ё* – [йо], *ю* – [йу].

Дані літери сигналізують про те, що приголосна літера, яка стоїть перед ними, позначає м'який приголосний звук:

Сяду – [с'аду], *сел* – [с'эл], *сёл* – [с'ол], *сюда* – [с'уда].

Крім даних літер для позначення м'якості приголосного використовуються літери *и* та *ь*. В російській мові 30 приголосних, парних за твердістю-м'якістю (15 літер) – *п, б, ф, в, м, т, д, с, з, н, л, р, к, г, х*, на які діють літери *и, я, ю, е, ё, ь*.

Літери *и, е, ё* після шиплячих літер *ж* та *ш* та літери *ц* не вказують на їх м'якість [4]. Приголосні звуки [ж], [ш] та [ц] в сучасній російській літературній мові завжди тверді.

Шил – [шы́л], *жест* – [жэ́с'т'],

шёл – [шо́л], *цена* – [це́на].

Приголосні звуки [ч] та [щ] в російській мові завжди м'які.

Чудо – [чу́до], *щука* – [щу́ка].

часть – [ча́с'т'].

При трансформації тексту м'якість приголосних вказується за допомогою літери *ь*: *ки, кьи, ни, ны, ги, гьи, ви, вьи* і т.д. Крім того для позначення сполучень літери *ь* з приголосним на кінці слова в якості пом'якшення або виконання граматичної функції (наприклад, у слові *мышь* *ь* означає, що даний іменник жіночого роду) були використанні літери латиниці. Всі "невитрачені"

літери *ь* вилучаються при трансформації. Так само вилучаються всі літери *ъ*.

Виконані наступні додаткові трансформації:

- враховуючи, що не всі тексти містили літеру *ё*, всі її входження були попередньо замінені на входження *е*;

- вилучені всі знаки лапки ('" ' та "' '), які не є сигналами для розстановки пауз;

- вилучені всі латинські літери;

- вилучені всі цифри.

Розстановка пауз

При автоматичному аналізі мови тексту однією з важливих задач є локалізація меж інтонаційних фраз та вибір фонетичних засобів їх реалізації. Під інтонаційною фразою розуміється зв'язний, граматично організований фрагмент тексту (речення або його частина), якому приписується одна інтонаційна модель [5]. Межі інтонаційних фраз визначаються паузами, які відповідають фізичній перерві у мовному сигналі. Пауза приймає участь у передачі певних синтаксичних та смислових відношень в тексті, вона є засобом поділу мови на інтонаційно-смислові одиниці та засобом виявлення характеру зв'язку між частинами висловлювання.

Практика створення автоматичних систем синтезу мови за довільним текстом показує, що найважливішими ключами для визначення інтонаційних меж та, відповідно, розстановки пауз в тексті є розділові знаки. Зокрема, в [5] описано експеримент, спрямований на визначення ступеня однорідності розстановки пауз при читанні одного і того ж тексту різними людьми. Для відібраних варіантів прочитаного тексту була побудована таблиця ймовірності появи пауз (Таблиця 1) у прочитаному тексті та аналізувалося питання надійності розділових знаків для розстановки фізичних пауз в тексті при його озвучуванні автоматичною системою синтезу мови. Згідно з отриманими результатами середня ймовірність появи паузи для внутрішніх розділових знаків склала 0,7, в той час як паузи, які були виділені зовні розділових знаків, мають середню ймовірність, яка складає лише 0,28. Більша частина пауз з'являється на кінцях речень та у місцях синтаксичних меж, яким відповідають внутрішні розділові знаки. Однак, слід зауважити, що внутрішнім розділовим знакам не завжди відповідає пауза. Особливо це стосується *коми*, яка є найбагатозначнішим із знаків пунктуації. Серед випадків, в яких вона вживається, є достатньо таких, в яких пауза не потрібна.

Таблиця 1

Кількість пауз

	Відповідних розділовим знакам				Що не відповідають розділовим знакам (всього 59)	Всього
	Зовнішнім	Внутрішнім				
	Крапки	Коми (всього 39)	Тире (всього 4)	Двокрапка (всього 5)		130
$P_{сер}$	0,99	0,64	0,84	0,93	0,28	

Таким чином, на етапі трансформації тексту до множини знаків, що утворюють розриви (паузи) та не вилучаються з тексту, включаються: *знаки оклику, знаки питання, крапки, крапки з комою, двокрапки та дужки*. Знак *тире* замінюється на *крапку* з метою уникнення випадків помилкового трактування знаку дефіса як тире (що часто виникає при невірному вживанні знаку '-' замість '—'). Всі *дефіси* вилучаються.

Додаткові розриви утворюються спеціальними знаками, наприклад #, №, &, тощо. Очевидно, що такий знак якщо і утворює якийсь зв'язок, то його аналіз є дуже ускладненим. Також до причин розриву віднесені цифри.

Відповідно, текст тоді представляється, як послідовність звуків. Для контролю було реалізовано трансформацію, описану в [2], перехідний варіант та чисто фонічний варіант, а саме:

- текст перетворюється на послідовність літер алфавіту та пробілів (*перша трансформація*);
- текст перетворюється на послідовність звуків та пробілів (*друга трансформація*);
- текст перетворюється на набір послідовностей звуків (*третья трансформація*).

Збір даних на трансформованих текстах

В існуючих підходах атрибуції текстів [6] з використанням літерної інформації елементами тексту є послідовності літер від початку до кінця тексту. При цьому, така ознака, як послідовність літер заданої довжини в межах слова є, як правило, характеристикою сильно залежною від словника тексту, який в свою чергу сильно залежить від предметної області або теми тексту.

Було висунуто гіпотезу, що послідовності літер (n -грами) на межах суміжних слів також можуть слугувати хорошою ознакою авторства,

будучи менш залежними від словника тексту, а більше – від того, яким чином автор тексту сприймає написаний ним текст і наскільки він приділяє увагу благозвучності. Тоді найбільш суттєвими ознаками є збіг голосних або збіг приголосних на межах суміжних слів; вони виражають тип порушення благозвучності мови. Наприклад, *«Всех чувств взор есть благороднее»*.

Також було висунуто гіпотезу, що сукупність n -грам, не перетворена до матриці ймовірностей переходів марківського процесу може давати кращу якість розпізнавання, за умови вживання деякої іншої міри близькості, відмінної від описаної в [2].

Формування вектора ознак тексту

На основі перетворень досліджено властивості наборів ознак за частотами наступних n -грам:

- 1) біграм (при $n=2$) для тексту першої трансформації, які перетворені відповідно до марківського процесу;
- 2) біграм сформованих для тексту, що зазнав другої трансформації;
- 3) біграм для тексту другої трансформації, які перетворені відповідно до марківського процесу;
- 4) біграм, сформованих для тексту, що зазнав третьої трансформації;
- 5) біграм для тексту третьої трансформації, що перетворені відповідно до марківського процесу;
- 6) біграм, що приходяться на межі між словами та містять лише приголосні або лише голосні, для тексту у третій трансформації;
- 7) біграм, що містяться лише і виключно в словах та не містять розривів, для тексту у третій трансформації;

8) триграм (при $n=3$), для тексту у третій трансформації;

9) триграм, що приходяться на межі між словами та містять лише приголосні або лише голосні, для тексту у третій трансформації;

10) триграм, що містяться лише і виключно в словах та не містять розривів, для тексту у третій трансформації;

11) об'єднаний комплект n -грам, а саме, біграм та триграм, що приходяться на межі між словами та містять лише приголосні або лише голосні, для тексту у третій трансформації;

12) об'єднаний комплект n -грам, а саме біграм та триграм, що містяться лише і виключно в словах і не містять розривів, для тексту у третій трансформації;

13) об'єднаний комплект n -грам, а саме біграм та триграм, для тексту у третій трансформації.

Кожен з 13-ти векторів ознак тексту містить частоти всіх вживань відповідного типу n -грам. При цьому, всі частоти приводяться до розміру з розрахунку на 30000 звуків тексту. Тобто, для k -го вектора ознак тексту T

$$v_k^T = (ch_{k1}^T, ch_{k2}^T, \dots, ch_{km}^T), \quad k = \overline{2, 4, 7, 13};$$

$$ch_{ki}^T = \frac{ch_{ki}^T}{n^T} \cdot 30000, \quad i = \overline{1, m}$$

де ch_{ki}^T – частота i -ї n -грами відповідного типу k для тексту T , n^T – кількість фонем у тексті T , m – може бути різним для кожного k та кожного тексту T .

Для приведення частот біграм до значень ймовірностей обчислювались числа: m_{ij} – кількість переходів із i -ї літери в j -ту в тексті, що аналізується та n_i – загальна кількість переходів із i -ї літери. Позначимо $p_{ij} = m_{ij} / n_i$ – ймовірність переходу із i -ї літери в j -ту. Тоді для $k = 1, 3, 5$:

$$v_k^T = (p_{11}, p_{12}, \dots, p_{14}, \dots, p_{41}, p_{42}, \dots, p_{44}).$$

Визначення подібності текстів з використанням міри косинусів

Схожість між текстами, які задані в рамках однієї фонічної структури, пропонується визначати за допомогою міри косинусів. Міра косинусів враховує напрямок вектора у просторі. Напрямок вектора у просторі ознак тексту інтерпретується як якісна подібність (близькість) текстів за цими ознаками.

Якщо $X = (x_1, x_2, \dots, x_n)$ та $Y = (y_1, y_2, \dots, y_n)$ – вектори в n -вимірному просторі ознак та θ – кут між векторами X та Y , то їх косинус кута дорівнює:

$$\cos \theta = \frac{X \cdot Y}{\|X\| \cdot \|Y\|},$$

де $X \cdot Y$ – скалярний добуток між векторами X та Y :

$$X \cdot Y = x_1 y_1 + x_2 y_2 + \dots + x_n y_n.$$

Атрибуція тексту, тобто віднесення його до відповідного класу – автора, здійснюється шляхом обчислення косинусів між вектором ознак даного тексту та вектором ознак, який є центром даного класу, тобто складається із середніх значень ознак даного класу. При обчисленні косинусів вектори приводяться до однієї розмірності шляхом дописування нульових значень для компонент вектора тексту, в якому відсутня дана n -грама. При визначенні центра класу, який в дійсності відповідає автору тексту, що аналізується, даний текст не враховується. Нехай k – номер класу текстів, що відповідає їх автору, тоді при визначенні класу тексту T , що аналізується, центр класу з номером k

визначається за формулою $x_{ck}^T = \frac{\sum_{j=1}^{n_k} x^j - x^T}{n_k - 1}$, для k , що відповідає реальному автору тексту T та

$$x_{c_k}^T = \frac{\sum_{j=1}^{n_k} x^j}{n_k}, \quad \text{для } k, \text{ що відповідає іншому автору.}$$

Тут c_k – номер центра k -го класу, n_k – кількість текстів k -го класу, $x_{c_k}^T$ – вектор-центр k -го класу відповідно до тексту T , що аналізується, x^j – вектор ознак j -го тексту, x^T – вектор ознак тексту T , що аналізується.

Обчислення косинусів проводиться між вектором ознак тексту, що розглядається, та кожним з центрів класів. Номер класу (автор) N , що відповідає даному тексту, визначається наступним чином:

$$N = \underset{k}{\text{Arg max}} \cos \theta_k,$$

де:

$$\cos \theta_k = \frac{x^T \cdot x_{ck}^T}{\|x^T\| \cdot \|x_{ck}^T\|}, \theta_k - \text{кут між векторами } x^T \text{ та } x_{ck}^T.$$

Визначення подібності текстів з використанням ланцюгів Маркова

Для обчислення близькості між векторами ознак використовується наступна формула:

$$A_k = - \sum_{i=1}^l \sum_{j=1}^l m_{ij} (\ln(\frac{m_{ij}}{n_{ji}}) - \ln(\frac{m_{2ij}}{n_{2i}}))$$

де m_{ij} - число переходів із i -ї літери в j -ту в тексті, що аналізується, n_{ji} - загальне число переходів із i -ї літери в тексті, що аналізується (для x^T - вектора ознак тексту T),

m_{2ij}, n_{2i} - аналогічні числа для вектор-центру класу x_{ck}^T .

Номер класу (автор) N , що відповідає даному тексту, визначається наступним чином:

$$N = \text{Arg min}_k A_k.$$

де A_k - значення відповідної міри близькості.

Необхідно зауважити, що ця формула містить знакозмінну компоненту, що повинно було б не найкращим чином відбиватися на точності розпізнавання. Проте результати отримані в роботі [6], показують, що при введенні в формулу модуля точність практично не змінюється.

Результати атрибуції текстів та їх інтерпретація

Для атрибуції були використані твори 20 авторів, дані представлені у таблиці 2.

Таблиця 2

Автор	Об'єм тексту в Кб	Кількість текстів	Середній об'єм в Кб
А.Столяров	1376.5	6	229.4
А. Толстой	267	4	66.8
М. Булгаков	1572.3	7	224.6
А.Чехов	422.9	5	84.6
Ф.Достоевский	879.2	6	146.5
Н.Гоголь	1365.1	8	170.6
М.Горький	204.7	4	51.2
Е. Хаецкая	546.6	5	109.3
С. Казменко	947.9	4	237
Е.Козловский	940	8	117.5
Л.Кудрявцев	568.6	4	142.2

Таблиця 2 (Продовження)

Автор	Об'єм тексту в Кб	Кількість текстів	Середній об'єм в Кб
А.Куприн	745.4	7	106.5
Л.Толстой	595.3	6	99.2
А.Лазаревич	1128.8	5	225.8
Ю.Латынина	3398.8	7	485.5
С.Логинов	1921.3	6	320.2
А.Пушкин	846.1	6	141
С.Щеглов	4397.5	6	732.9
М.Шолохов	898.1	5	179.6
И.Тургенев	1193.7	9	132.6
Всього	24215.8	118	

Для визначення якості критеріїв використовується набір оцінок, що базується на рангах. Рангом будемо називати позицію оцінки близькості у відсортованому списку таких оцінок. Нульовий ранг відповідає найкращій оцінці, останній - найгіршій.

Оцінки:

- 1) кількість творів, що отримали нульовий ранг (кількість точних визначень авторства) та обчислений на її основі відсоток точності;
- 2) середній ранг за всіма творами;
- 3) усереднений квадрат рангів, зібраний за всіма творами, як оцінка діапазону, в якому знаходиться ранг істинного автора.

Результати першої групи експериментів, з комплектами біграм за номерами 2, 4, 6 в загальній послідовності дали результати, що представлені у таблиці 3.

Таблиця 3

Набір ознак	(1)	(3)	(5)
Кількість точних визначень	77	79	82
Точність	65,25%	66,94%	69,49%
Сума рангів	245	203	199
Середній ранг	2.08	1.72	1.68
Середньо-квадратичний ранг	26.08	19.36	19.23

З чисельного експерименту видно, що перехід від літер до звуків підвищує точність та надійність. Перехід від біграм, що містять компоненти з пробілами, до біграм, де замість них представлені відомості про збіги на межах

слів, також підвищує точність та надійність розпізнавання.

Друга група експериментів стосувалася використання косинусної міри близькості.

Результати експерименту подані в таблиці 4.

Таблиця 4

Набір ознак	(2)	(4)	(8)
Кількість точних визначень	87	89	94
Точність	73,72%	75,42%	79,66%
Сума рангів	87	97	104
Середній ранг	0.737	0.822	0.881
Середньоквадратичний ранг	4.14	5.09	8.23

Як видно з даного чисельного експерименту, використання косинусної міри близькості підвищує точність та додатково підсилює висновок, що перехід до даних, які містять інформацію про збіги звуків на межах слів є хорошим перетворенням для підвищення точності.

В той же час механічне збільшення розмірності даних (перехід від біграм до триграм) є дещо сумнівним, оскільки крім зростання точності значно виросло розсіювання.

Третя група експериментів стосувалася спроби виділити специфічну інформацію щодо гіпотези про авторське сприйняття тексту. Результати експерименту подано в таблиці 5.

Таблиця 5

Набір ознак	(6)	(7)
Кількість точних визначень	82	94
Точність	69,49%	79,66%
Сума рангів	150	106
Середній ранг	1.271	0.898
Середньоквадратичний ранг	9.08	8.61

Як видно з чисельного експерименту, якість розпізнавання на 7-му наборі даних, а саме, біграмах, що формуються лише в межах слів і не визначаються на межах слів, є порівняльною з якістю 8-го набору даних – триграм. Якість розпізнавання на 6-му наборі даних, який мало залежить від словнику – суттєво нижча за бажану.

Четверта група експериментів (табл.6) сформована аналогічно третій, але для триграм.

Таблиця 6

Набір ознак	(9)	(10)
Кількість точних визначень	62	90
Точність	52,54%	76,27%
Сума рангів	302	95
Середній ранг	2.56	0.805
Середньоквадратичний ранг	25.19	5.071

Низька точність 9-го набору даних з нашої точки зору пояснюється низькою частотою збігів трьох голосних чи приголосних у тексті. Справді, кількість таких збігів не перевищує 530 на 30000 звуків. Більш низька точність, досягнута на 10-му наборі порівняно з 8-м пояснюється тим, що збіги по 3-м приголосним чи голосним майже не трапляються всередині слів. Таким чином 8-й набір даних містить 9-й та 10-й, як майже незалежні, тому і виграє у точності та надійності.

П'ята група експериментів (табл. 7) сформована для векторів даних, зібраних одночасно з біграм та триграм. З нашої точки зору, таке змішування є допустимим, оскільки обчислення і біграм, і триграм є різними способами згорнути частотний словник тексту та набір фонічних властивостей, притаманних автору. Як видно з чисельного експерименту, на основі лише інформації про збіги приголосних та голосних на межах слів можна отримати точність співмірну з точністю біграм, зібраних на всіх звуках.

Таблиця 7

Набір ознак	(11)	(12)	(13)
Кількість точних визначень	88	90	93
Точність	74,58%	76,27%	78,81%
Сума рангів	115	94	104
Середній ранг	0.975	0.797	0.881
Середньоквадратичний ранг	5.96	5.02	8.44

Також помітно, що додавання біграм до триграм (2-го набору даних до 8-го) не покращує результати визначення авторства.

Крім того, було проведено окремий чисельний експеримент, зумовлений тим, що в ході попередніх експериментів виявилось, що досить високі показники кожного критерію окремо та їх часткова незалежність не дають повної

значень точності, які могли б бути досягнуті. Справді, коли в одному векторі зібрані біграми та триграми, то компоненти вектора не перекриваються. Також, біграми та триграми, зібрані на межах слів, трапляються суттєво рідше всередині слів, або часто не трапляються взагалі. Тому було застосовано метод голосування Борда для 11-го та 12-го набору даних (табл.8).

Таблиця 8

Набір ознак	(11)	(12)	Голосування
Кількість точних визначень	88	90	85
Точність	74,58%	76,27%	72,03%
Сума рангів	115	94	82
Середній ранг	0.975	0.797	0.695
Середньоквадратичний ранг	5.96	5.02	2.83

При точності 72,03% (порівняно з потенційно можливою 79,66% при іншому способі виділення ознак) метод з використанням методу голосування Борда показав дуже високу стабільність.

Список використаних джерел

1. Анисимов А.В., Порхун Е.В., Тарануха В.Ю. Алгоритм формирования параметрического вектора для решения задач классификации нейронной сетью прямого распространения // Кибернетика и системный анализ. – 2007. – №2. – С. 3-14.
2. Хмелев Д.В. Распознавание автора текстов с использованием цепей А.А. Маркова // Вестник МГУ. – Сер.9: Филология. – 2000. – №2. – С. 115-126.
3. Кукушкина О.В., Поликарпов А.А., Хмелев Д.В. Определение авторства текста с использованием буквенной и грамматической информации // Проблемы передачи информации. – 2001. – т. 37. – вып. 2. – С. 96-108.

Це дозволяє стверджувати, що навіть якщо відповідь не точна, то справжній автор буде в близькому околі від отриманого результату окремого тесту.

Висновки

В роботі отримано наступні результати:

- 1) Косинусова міра близькості краще за якістю (як точністю, так і стабільністю) мір, що базуються на представленні тексту, як марківського ланцюга.
- 2) Використання фонем замість літер дещо підвищило точність та стабільність визначення авторства, проте вимагає подальшого дослідження.
- 3) За результатами голосування критеріїв в останньому чисельному експерименті масмо найнижчий середній ранг та найнижчий середньоквадратичний ранг. Він в 2,98 разів кращий за середньоквадратичний ранг у випадку комбінування критеріїв у 13-му наборі даних. З цього можна зробити висновок про необхідність виділення з тексту якомога більшої кількості незалежних критеріїв.
- 4) Ознака якості звучання тексту при читанні є прийнятною ознакою визначення авторства.
4. Балашова В.В. Дементьев Л.В. Фонетика, словообразование, морфология и орфография. Режим доступа - http://www.licey.net/russian/phonetics/1_5
5. Кривнова О.Ф., Чардин И.С. Паузирование при автоматическом синтезе речи // Теория и практика речевых исследований (АРСО-99). – Москва. – 1999.
6. Поддубный В.В., Шевелев О.Г., Фатыхов А.А. Сравнительный анализ эффективности алгоритмов распознавания авторства по частотам переходов // Збірка наукових праць у п'яти томах Міжнародної наукової конференції "Інтелектуальні системи прийняття рішень та прикладні аспекти інформаційних технологій" (ISDMIT '2005). – т.1. – Євпаторія. – 2005. – С. 125-129.

Надійшла до редколегії 14.03.2011

УДК 004.9:007

Бісікало О.В., к.т.н., доц.

Побудова нечітких відношення і простору сенсу образних конструкцій

У статті розглядається онтогенетичний метод побудови нечітких відношення і простору сенсу образних конструкцій. Мета дослідження полягає у визначенні нечіткого відношення сенсу та його основних властивостей, формалізації простору з нечіткою мірою та інтерпретації базових операцій з нечіткими відношеннями сенсу для задач пізнавальної діяльності людини.

Ключові слова: нечітке відношення, простір сенсу, нечітка міра, образні конструкції.

*E-mail: obisikalo@gmail.com

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ. Феноменологічний характер підходу до розуміння сенсу навчального контенту (НК) [1] дозволяє припустити, що в основу простору образних конструкцій (ОК) має бути покладена нечітка міра. Запропонована у [2] функція сенсу як характеристика сили асоціативного зв'язку між парою образів за формальними ознаками близька до функції належності нечіткого відношення, а розглянутий у [3] онтогенетичний спосіб побудови таких функцій моделює природний шлях накопичення знань людиною. Відсутнім поки лишається огляд особливостей ОК як бінарних нечітких відношень, для яких функцією належності є функція сенсу-параметру. Отже, задачами дослідження вважатимемо визначення нечіткого відношення сенсу та його основних властивостей, формалізацію простору з нечіткою мірою та інтерпретацію базових операцій з нечіткими відношеннями сенсу для задач пізнавальної діяльності людини.

Нечітке відношення сенсу. Припустимо, що система S , яку в подальшому називатимемо інфологічною (ІС), здатна розпізнавати образи з нескінченної множини $I = \{i_1, i_2, \dots, i_n, \dots\}$ та сприймати асоціативні зв'язки між парами образів як елементи множини $\omega \in \Omega$, де $\Omega \subseteq I \times I$ – довільна множина упорядкованих пар. ОК будемо вважати будь-яку підмножину $\gamma \subset \Omega$ як елемент $\gamma \in F$ – σ -алгебри підмножин Ω , в т.ч. $\Omega \in F$. ІС обмінюється інформацією з

O.V. Bisikalo, Ph.D.

Fuzzy relation and space of sense of image structures constructing

The ontogenetic method to construct the fuzzy relation and space of sense of image structures is examined in the article. The purpose of the paper is to define the fuzzy relation of sense and it's basic properties, to formalize the fuzzy measure space, to interpretative the base operations with fuzzy relations of sense for tasks of cognitive activity of the human.

Key Words: fuzzy relation, space of sense, fuzzy measure, image structures.

зовнішнім світом як «чорним ящиком» виключно у вигляді ОК, з яких розрізняють послідовність входних подій $X = \{x_1, x_2, \dots\}$ та множину образних реакцій системи $Y = \{y_1, y_2, \dots\}$, причому $x_i \in F$, $y_j \in F$. Введено онтогенетичний принцип побудови системи S – множина Ω , що визначає сенс її діяльності, отримується виключного шляхом послідовного накопичення оператором $\oplus$ параметрів чергових ω з зовнішнього «чорного ящику» та подальшого самовдосконалення Ω [2,3].

Розглянемо ключові ознаки онтогенетичного методу створення бінарного нечіткого відношення сенсу Q ІС, яке визначено на одній базисній множині (універсумі) образів I

$$Q = \{ \langle i_l, i_j \rangle, \mu_Q(\langle i_l, i_j \rangle) \}, \quad (1)$$

де $\mu_Q(\langle i_l, i_j \rangle)$ – функція належності бінарного нечіткого відношення, що задається як відображення $\mu_Q: I \times I \rightarrow [0,1]$. Через $\langle i_l, i_j \rangle$ позначимо кортеж з двох елементів (пару образів), причому $i_l \in I$, $i_j \in I$. Бінарне нечітке відношення (1) застосовано в якості онтогенетичної характеристики множини Ω , тоді функцію належності

$$\mu_Q(\langle i_l, i_j \rangle) = f(k_j, i_L), \quad (2)$$

де k_{ij} – кількість зафіксованих ІС зв'язків між i -м та j -м образами на момент часу t_L , запропоновано вважати природною чисельною мірою сенсу 1 *Sav* (синтагматичної асоціації вага) або *Saw* (Syntagmatic association weight) [2].

У відповідності до концептуального підходу деталізуємо функцію належності, що породжує бінарне нечітке відношення сенсу (1) на таких чотирьох послідовних рівнях, побудованих на базовому (2):

1. Рівень імовірнісного прогнозування – з метою нормування функції належності у проміжку $[0,1]$ передбачується розрахунок статистичної оцінки λ (математичного сподівання): якщо $k_{\Sigma} = \sum_{i=1}^n \sum_{j=1}^n k_{ij}$, а m – кількість ненульових елементів матриці A_Q , то $\lambda = k_{\Sigma} / m$ – в цьому випадку застосуємо відому сигмоїдальну функцію

$$\mu_Q(< i_l, i_j >) = f_1(k_{ij}, \lambda) = 1 / (1 + e^{-k_{ij} + \lambda}).$$

Внаслідок нормування з'являється характерна властивість функції належності, отриманої за онтогенетичним методом – середнє значення

$$\overline{\mu_Q} = \sum_{j=1}^m \mu_{Qj} / m = 0,5.$$

2. Рівень врахування емоційного стану – введемо можливість врахування емоційних станів ІС за рахунок показника $\mu = \{..., -2, -1, 1, 2, ...\}$, тоді

$$\mu_Q(< i_l, i_j >) = f_2(k_{ij}, \lambda, \mu) = 1 / (1 + e^{\frac{k_{ij} - \lambda}{|\mu|}}). \quad (3)$$

3. Рівень врахування мотиваційної компоненти на основі образів-центрів потреб – пропонується мотивом ІС на момент часу t_L вважати досягнення образу-центру потреби j^* , а також розрахувати дисперсію та середньоквадратичне відхилення результатів спостережень k_{ij} як $D = \frac{1}{m} \sum_{i=1}^n \sum_{j=1}^n (k_{ij} - \lambda)^2 \mid k_{ij} > 0$

і $\sigma = \sqrt{D}$. Тоді, в залежності від ступеня наближення пари образів $< i_l, i_j >$ до j^* , функцію (3) можна зміщувати вліво за віссю абсцис шляхом зменшення математичного сподівання для цієї пари $\lambda_{ij} = \lambda - r \cdot \sigma$, де $r = \{0, 1, 2, 3\}$, зрештою маємо

$$\mu_Q(< i_l, i_j >) = f_3(k_{ij}, \lambda_{ij}, \sigma, \mu, i^*) = 1 / (1 + e^{\frac{k_{ij} - \lambda_{ij}}{|\mu|}}). \quad (4)$$

Розглянутий підхід передбачає застосування окремого алгоритму для визначення ступеню наближеності r пари $< i_l, i_j >$ до образу-потреби j^* , наприклад:

- а) якщо $j = j^*$, то $r = 3$, інакше
- б) якщо $\exists e_k \in E \mid e_k = < v_l, v_{j^*} >$, то $r = 2$, інакше
- в) якщо $\exists e_k \in E \mid e_k = < v_l, v_j >$, то $r = 1$, інакше $r = 0$.

4. Рівень врахування рефлексів та результатів зовнішнього навчання – пропонується фіксувати в ІС статистику результатів образної реакції y' на вхідну ОК x' при мотиваційній меті i^* як зміну емоційного параметру $\Delta\mu$. Тоді, при наявності в попередньому досвіді схожих ситуацій $< i^*, x' >$ для певної підмножини пар $e_{ij} \in E'$, де $E' \subseteq E$, застосовується зміна параметру $\lambda_{ij} = \lambda \pm r \cdot \sigma$. Вибір $r = \{0, 1, 2, 3\}$ залежить виключно від знаку та значення $\Delta\mu$, а функція належності матиме вигляд (4) з такими змінами

$$\mu_Q(< i_l, i_j >) = f_4(k_{ij}, \lambda_{ij}, \sigma, \mu, i^*, x', \Delta\mu).$$

Простір з нечіткою мірою. Розглянемо можливості аксіоматизації простору з мірою на основі запропонованого нечіткого відношення сенсу Q . Відомо, що мірою називають функцію множини $m: P(X) \rightarrow R^+$, яка відповідає таким трьом аксіомам [4]:

- 1. $\forall A \subseteq X \Rightarrow m(A) \geq 0, m(\emptyset) = 0$;
- 2. $A \subseteq B \Rightarrow m(A) \leq m(B)$;
- 3. якщо $A, B \in P(X)$ та $A \cap B = \emptyset$, то $m(A \cup B) = m(A) + m(B)$,
де $P(X)$ – множина всіх підмножин X (σ -алгебра), $R^+ = [0, \infty]$ – множина додатних дійсних чисел.

При $R^+ = [0, 1]$ ці аксіоми визначають імовірнісну міру, в основу якої покладено базове поняття події з можливостями експериментальної перевірки. В теорії нечітких мір замість події застосовується поняття суб'єктивного ступеня впевненості людини в цій події, який також зручно обмежити відрізком $[0, 1]$. За рахунок узагальнення поняття міри виникає можливість уникнути обмежувальної вимоги адитивності, що

забезпечують відомі λ -нечіткі міри Сугено для визначення ступеня необхідності, довіри, правдоподібності, можливості тощо [5].

Відповідно до концепції розуміння сенсу елементів НК та визначення пізнавальної діяльності простір сенсу-параметру неможливо обмежити мірою на відрізку $[0,1]$. Не дивлячись на те, що вчитель-експерт ставить оцінку студенту за будь-якою системою оцінювання знань саме на основі особистого суб'єктивного ступеня впевненості, проте об'єктом порівняння для нього при визначенні оцінки служать власні знання. Отже, результати пізнавальної діяльності мають властивість необмеженого або обмеженого фізичною тілесністю зростання, а відповідний до них сенс-параметр будемо вимірювати на $R^+ = [0, \infty]$.

На основі базового поняття знань/вмін/навичок як результатів пізнавальної діяльності людини [3] і властивостей бінарного нечіткого відношення сенсу задамо такі аксіоми нечіткої міри сенсу $Se: F \rightarrow [0, \infty]$:

1. $\forall \gamma \subseteq \Omega \Rightarrow Se(\gamma) \geq 0, Se(\emptyset) = 0, Se(\Omega) \leq \infty$;
2. $\forall \omega \in \Omega \Rightarrow Se(\omega) \leq 1, Se(\omega) = \mu_Q(\omega)$;
3. якщо $\gamma_1, \gamma_2 \in F$ та $\gamma_1 \subseteq \gamma_2$, то $Se(\gamma_1) \leq Se(\gamma_2)$ (монотонність);
4. якщо $\gamma_i \in F$, де $\{\gamma_i, i = 1, 2, \dots\}$ є монотонною послідовністю $\gamma_1 \supseteq \gamma_2 \supseteq \dots \supseteq \gamma_i \supseteq \dots$, то $\lim_{i \rightarrow \infty} Se(\gamma_i) = Se(\lim_{i \rightarrow \infty} \gamma_i)$ (неперервність);
5. якщо $\{\gamma_i\}_{i=1}^{\infty} \in F$ – рахункове сімейство множин з F , що попарно не перетинаються, тобто $\gamma_i \cap \gamma_j = \emptyset, i \neq j$, то $Se(\bigcup_{i=1}^{\infty} \gamma_i) = \sum_{i=1}^{\infty} Se(\gamma_i)$ (σ -адитивність).

Під простором сенсу з нечіткою мірою будемо розуміти трійку (Ω, F, Se) , де Se – нечітка міра сенсу або сенс-параметр, що визначається на основі аксіом 1÷5. Запропонована міра Se вважається σ -кінцевою, оскільки існує рахункове сімейство множин, що вимірюється $\{\gamma_i\}_{i=1}^{\infty} \in F$, таке що

$$Se(\gamma_i) < \infty, i \in N^+ \text{ та } \Omega = \bigcup_{i=1}^{\infty} \gamma_i, \text{ де } N^+ \text{ – множина}$$

додатних натуральних чисел. Використання σ -кінцевої міри Se призводить до того, що весь простір сенсу може бути представленим у вигляді рахункового об'єднання множин, що вимірюється через кінцеву міру. На основі

аксіоми 5. задамо означення I : якщо у ОК x та y є перетин, то $Se(x + y) = Se(x) + Se(y) - Se(x \cap y)$.

Покажемо, що можливість подання сенсу у числовому вигляді також дозволяє розглядати простір упорядкованих пар образів Ω як топологічний і квазіметричний. Простір асоціативних пар Ω є топологічним, оскільки множина F її підмножин є топологією на Ω , для якої виконуються умови [6]:

1) $\Omega \in F, \emptyset \in F$ – множина Ω і пуста множина належать F ;

2) об'єднання та перетин довільного сімейства множин, що належать F , належить F (згідно з властивостями σ -алгебри).

Простір асоціативних пар Ω є квазіметричним простором [3], оскільки будь-яким двом елементам $\omega_i, \omega_j \in \Omega$ ставиться у відповідність невід'ємне число q таке, що $q(\omega_i, \omega_i) = 0$, а для будь-якої трійки $\omega_1, \omega_2, \omega_3 \in \Omega$ виконується нерівність трикутника $q(\omega_1, \omega_3) \leq q(\omega_1, \omega_2) + q(\omega_2, \omega_3)$ за умови, що квазівідстань $q(\omega_i, \omega_j) = |Se(\omega_i) - Se(\omega_j)|$.

Простір Ω не можна вважати метричним простором, оскільки не виконується умова про те, що $q(\omega_i, \omega_j) = 0$ тоді і тільки тоді, коли $\omega_i = \omega_j$.

Операції з відношеннями сенсу. Розглянемо пізнавальну інтерпретацію основних операцій над бінарними нечіткими відношеннями сенсу $Q = \{ \langle i_l, i_j \rangle, \mu_Q(\langle i_l, i_j \rangle) \}$ та $R = \{ \langle i_l, i_j \rangle, \mu_R(\langle i_l, i_j \rangle) \}$, що задані на декартовому добутку $I \times I$ того ж самого універсума образів, тобто $\langle i_l, i_j \rangle \in I \times I$. Будемо вважати, що відношення Q відповідає інфологічній системі S_Q , а відношення R відповідає іншій ІС S_R .

Результатом перетину нечітких відношень сенсу Q та R назвемо таке нечітке відношення сенсу $P = \{ \langle i_l, i_j \rangle, \mu_P(\langle i_l, i_j \rangle) \} = Q \cap R$, яке задане на тому ж самому декартовому добутку $I \times I$ та функція належності якого визначається за формулою

$$\mu_P(\langle i_l, i_j \rangle) = \min\{\mu_Q(\langle i_l, i_j \rangle), \mu_R(\langle i_l, i_j \rangle)\}.$$

Операція перетину (« $\cap$ ») визначає нове нечітке відношення сенсу P , що має спільні знання двох незалежних інфологічних систем S_Q та S_R .

УДК 004.4

Бондар Є.С.

Глибовець М.М.

Гороховський С.С.

Bondar Ye.,

Glybovets M.,

Gorokhovsky S.

Хмарні обчислення та їх застосування

У світі з'явилась нова мережна парадигма, що обіцяє значно змінити комп'ютерний світ, підходи до Інтернету і вплинути на подальший розвиток інформаційних технологій – хмарні обчислення (Cloud Computing). У статті проведено загальний огляд нової технології. Також проведено аналіз переваг, недоліків і ризиків хмарних обчислень і загально описані групи користувачів, для яких нова технологія могла б стати у нагоді. Проаналізовано місце хмарних обчислень посеред інших мережних парадигм і технологій.

Ключові слова: хмарні обчислення, розподілені обчислення, Грід, віртуалізація.

*E-mail: glib@ukma.kiev.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

У світі з'явилась нова мережна парадигма, що обіцяє значно змінити світ Інтернету і вплинути на розвиток інформаційних технологій – хмарні обчислення (Cloud Computing). Але у зв'язку з тим, що технологія нова, мало хто знає про її можливості, як і про те, в який спосіб найкраще її використовувати.

Сформулюємо основні задачі цієї оглядової статті:

1. Зробити загальний огляд нової технології, її структури, ієрархії та можливостей;
2. Провести аналіз переваг і недоліків хмарних обчислень;
3. Проаналізувати місце хмарних обчислень посеред інших парадигм і технологій.

Останнім часом здавалось, що Інтернет здатен зробити все, так само, як і знищити все, що було до нього. Після початку бурхливого розвитку онлайн-літератури всі боялись, що індустрія паперових книжок помре, аж поки не з'явився Amazon. Ніхто не знав і не вмів правильно шукати інформацію, аж поки не народився Google. "Нащо мені Інтернет, якщо я не можу робити на ньому комерцію?", – і народився Ebay. Потім всі подумали, що Інтернет

Cloud Computing and its Applications

A new network paradigm that promises to change the computer world, approaches to the Internet and to make great impact on further development of information technology - Cloud Computing. The paper is devoted to overview of new technology. We also analyzed the advantages, disadvantages, and risks of Cloud Computing and generally described a group of users for whom the new technology could be useful. The place of Cloud Computing among other network paradigms and technologies is investigated.

Key Words: Cloud computing, distributed Computing, Grid, virtualization.

це легко, просто і всі проблеми вже вирішено, аж поки не стався крах доткомів¹.

У світі, в якому поява нових технологічних трендів та сміливих ідей є нормою, на наших очах виник новий напрям індустрії, який обіцяє бути більшим ніж просто мода. Останні кілька років всі говорять про хмарні обчислення ([1]-[6]). Як і колись, кожен розуміє під цим щось своє, а в оцінці корисності думки експертів і простих користувачів часто прямо протилежні.

Тому спробуємо розібратись в цьому новому явищі інформаційного світу.

Хмарні обчислення (ХО) - це омріяне віртуальне середовище спеціалізованої обробки інформації.

Це та мрія, що робиться реальною прямо на наших очах. ХО стали можливим тільки зараз через нові технологічні можливості.

ХО як технологія, виникла на основі мрії будувати спеціалізовані середовища з мінімальними витратами для розв'язання будь-якої задачі користувача. Вона має хорошу нагоду докорінно змінити значну частину ІТ індустрії, зробити програмне забезпечення, як сервіс, привабливішим та деякою мірою змінити методи,

¹ Так називали інтернет-компанії, в назві вебсайту яких звичайно стояло .com

за якими розроблюється та продається устаткування.

І найважливіше. ХО обіцяють змінити сам підхід користувача до роботи з інформацією та програмами. Замість використання десктопного застосування на стаціонарному комп'ютері чи ноутбукі, все буде зберігатись у "хмарі" – туманне визначення серверів та комп'ютерів в Інтернеті. Хмарні системи дозволять мати доступ до інформації та сервісів з будь-якого місця в світі, звільнивши нас від необхідності мати стаціонарний комп'ютер та зробивши легшою кооперацію та спільну роботу багатьох людей, що знаходяться в різних місцях у світі.

Розробникам та користувачам спеціалізованих комп'ютерних систем більше не потрібно купляти дороге мережне обладнання для старту чи супроводу проекту, а потім витрачати значні кошти для його обслуговування. Компанії, що пропонують сервіси і послуги, чітко зорієнтовані на конкретно і обмежено групу користувачів, можуть збільшувати задіяні ресурси настільки швидко, наскільки зростають їхні програми, бо вартість оренди тисячі серверів на одну годину стала сумірною до оренди одного сервера на тисячу годин. Така еластичність ресурсу без пені за великі обсяги оренди є безпрецедентною в історії ІТ.

Певною мірою, появу ХО можна порівняти з революцією в електриці минулого сторіччя.

Десктопно-орієнтована робота може відійти у минуле, як тільки ми зможемо отримати постійний 24/7 універсальний доступ до своєї інформації та засобів роботи з нею, як і до засобів організації спільної роботи через хмару.

Проблеми термінології

Треба зазначити певну проблематику вживання термінів при перекладі з англійської на українську. Переклад *Cloud Computing* як *Хмарні обчислення* вже ввійшов у використання і змінити його неможливо, навіть недоцільно. Треба мати на увазі деякі втрати змісту під час перекладу.

Якщо до слова *cloud* нема суттєвих зауважень, і переклад *хмара* є доцільним і вичерпним, то ситуація зі словом *computing* є складнішою. Останнє має два значення:

- 1) обчислення;
- 2) обробка інформації, загальна робота з використанням комп'ютера.

При перекладі терміну логічно був вжитий коротший варіант, зручний для застосування і

зрозумілий, але варто наголосити, що в даному випадку під словом *computing* скоріше розуміється другий варіант, як загальна робота на комп'ютері, обробка інформації. Це іноді призводить до непорозумінь, коли під терміном *cloud computing* розуміють різновид або синонім розподілених або мережних обчислень. Це хибне уявлення, яке ми зараз спростуємо.

Помилки у сприйнятті концепції

При використанні традиційних методів роботи на комп'ютері запускається окремий елемент програмного забезпечення. Документи, що створюються, зберігаються там, де були створені, хоча вони можуть бути доступні з інших хостів мережі, за межами мережі вони недоступні. Вся інформація зосереджена в місці роботи обробника.

З використанням ХО більшість програмного забезпечення та даних зберігається на серверах, доступних через Інтернет. Навіть, якщо станеться збій певної програми на певному сервері, вони можуть у прозорий спосіб бути замінені. Дані та документи, залишаються доступними для розподіленої обробки верифікованим користувачем (обробником) в режимі реального часу. На відміну від традиційної роботи, ця модель є *документно-орієнтованою*. Який комп'ютер використовується для доступу, просто не має значення.

Давайте подивимось детальніше на проблему.

ХО не є мережними обчисленнями. В мережних системах документи та програмні застосування зберігаються на серверах компанії і можуть бути доступними лише вузькому колу користувачів в мережі компанії, або ззовні після процедури автентифікації.

ХО - поняття значно ширше. Вони включають різноманітні компанії, різноманітні та розподілені сервери і структуровані розподілені мережі. На відміну від мережних обчислень, хмарні сервіси та сховища доступні з будь-якого місця в світі з підключенням до Інтернету, тоді як мережний сервіс корпорації доступний лише з корпоративної мережі.

Базові концепції ХО

Сама концепція *хмари* на диво проста. Для початку можна сказати, що це велика група комп'ютерів, зв'язаних між собою. Ці комп'ютери можуть бути персональними або

мережними серверами, публічними або приватними, але їх взаємодія є організаційно та платформно незалежною та прозорою для користувача.

Наприклад, Google хостить² у себе хмару, що складається як з малих РС, так і з великих серверів. Хмара Google приватна (тобто компанія Google володіє всіма правами на неї), але публічно доступна (для користувачів Google). Всі сервіси, застосування, дані, що обслуговуються хмарою, доступні для широкого кола користувачів платформо та організаційно незалежно. Доступ - лише через Інтернет. Будь-який авторизований користувач може отримати доступ до сервісів з будь-якого комп'ютера через будь-яке підключення Інтернет. Також, що дуже важливо з точки зору користувача, технологія та інфраструктура, що лежить за хмарою, невидимі. Забезпечується повна прозорість протоколів і платформ.

З точки зору Google є шість ключових властивостей ХО.

- *Користувацько-орієнтованість.* Якщо ви користувач і підключені до хмари, все, що там міститься – документи, повідомлення, зображення, застосування – доступні вам. Можна легко зробити свою інформацію доступною іншим.
- *Задачно-орієнтованість.* Система сфокусована на оптимізацію того, що потрібно зробити і як програма може це зробити. Наприклад, нам байдуже, який тип текстового процесора надає хмара (і де він знаходиться). Важливіше якість і доступність створеного документа.
- *Потужність.* При потребі можна об'єднати потрібну кількість обчислювальних ресурсів.
- *Доступність.* Оскільки дані зберігаються у хмарі в розподілений спосіб, користувачі можуть миттєво отримати доступ до декількох репозиторіїв.
- *Інтелектуальність.* Доступ, обробка та аналіз даних у хмарі проходить на необхідному рівні інтелектуалізації, що визначається автоматично, чим і забезпечується значна ефективність роботи.
- *Програмованість.* Багато завдань, що покладаються на хмарні обчислення можуть бути оптимізовані, з використанням стандартних підходів, реалізованих

програмно. Наприклад, для захисту цілісності даних інформація не зберігається на єдиному комп'ютері у хмарі, а резервується ще декількох комп'ютерах у хмарі.

Можливо, найкращим і найпоширенішим прикладом сервісів хмарних обчислень є пакет застосувань від Google – Gmail, Google Docs & Spreadsheets, Google Calendar, Picasa тощо. Всі вони зберігаються на серверах Google, доступні через Інтернет і можуть використовуватись для колективної роботи будь-кому будь-коли будь-де.

ХО будують прихований міст від комп'ютера до користувача, від програми до завдання. Користувач тепер не переймається управлінням даними. Найважливіше, що дані зберігаються у хмарі і доступні, коли тільки потрібні для нього та інших авторизованих користувачів разом з програмними інструментами для їх обробки.

Відмінності відомих мережних парадигм і ХО

Природним видається необхідність порівняння ХО із використовуваним раніше поняттям *віртуальності* в ІТ.

Це абсолютно різні технології. У ХО масштабовані віртуальні ресурси даних забезпечують користувачів через Інтернет. База даних може бути фізично розташована у безпечному віддаленому місці розташування і дані можуть бути доступні через клієнтські комп'ютери за допомогою сервера бази даних. Натомість, віртуалізація створює віртуальні версії технологій, таких як сервери, операційні системи, мережеві ресурси і пристрої зберігання даних. Віртуалізація, по суті, дозволяє одному користувачеві досягти до декількох фізичних пристроїв. При віртуалізації або одна операційна система використовує багато комп'ютерів для роботи з базою даних, або один комп'ютер керує декількома машинами.

Не можна опустити з порівняння і *розподілені обчислення*.

Розподілені обчислення передбачають поділ великої задачі на дрібніші фрагменти, координацію обчислень. На кінець, розподілені обчислення мають певну мету, тоді як хмарні обчислення просто надають певний набір послуг через Інтернет.

Очевидним є і зіставлення з Грід. Тут можна послатися на відому статтю [8].

² Це ще один невдалий термін, що став загальноприйнятою калькою з англійського hosting

У Грід-обчисленнях один комп'ютер координує спільне розв'язання задач. Ця парадигма значно ближча до паралельних обчислень. Іншими словами, метою Грід є ілюзія створення єдиного потужного самокерованого віртуального комп'ютера, що сильно відрізняється від мети хмари.

Обидві технології обробки залучають мультиоренду і багатозадачність. Це означає, що багато клієнтів можуть виконувати різні завдання, маючи доступ до одного чи кількох екземплярів застосувань. Поділ ресурсів між великим об'єднанням користувачів допомагає в скороченні витрат на інфраструктуру та пікову потужність навантаження. Як Хмара так і Грід забезпечують угоду про рівень обслуговування (SLA) для гарантованої доступності. Якщо рівень сервісу опускається нижче гарантованого рівня обслуговування, споживач отримує кредит обслуговування для отримання даних пізніше. У Гріді для отримання максимальної віддачі обсяги розподілених даних мусять бути великими. Обчислювальний Грід фокусується на операціях з інтенсивними обчисленнями. В ХО пропонують два рівні підтримки надання послуг: стандартний та з високим навантаженням центральних процесорів.

Можна відзначити три спільні проблеми Грідів і Хмар [9].

Мається на увазі *поріг переходу* від експериментального запуску програми до виробничого застосування (наприклад, масивне збільшення запитів).

Важливим є й вирішення проблеми взаємозамінності (interoperability) та забезпечення сумісності застосувань від різних постачальників (API, формати баз даних, обмін повідомленнями тощо).

Насамкінець, якщо тестування ПЗ показує несподівані результати або вирішено ресурси, що не використовуються, необхідно вирішити цю проблему перед запуском програми в хмарі або в Гріді.

Ринок послуг хмарних обчислень

Для кращого розуміння переваг та можливостей ХО, коротко розглянемо кілька найвідоміших проектів світових проектів.

GoogleApps (пошта, календар, спільні таблиці й документи, групи, сайти, відео, переклад, блоги, мапи, малюнки тощо).

Базовий пакет – безкоштовний, розширений пакет (Premier Edition) – коштує 50 USD за рік на

одного користувача, і забезпечує систему безпеки, підтримку конференцій, доступ до програмного інтерфейсу, 24/7 підтримку тощо.

Освітній пакет (Education Edition) безкоштовний для неприбуткових освітніх організацій, шкіл, коледжів, університетів, з деякими обмеженнями, щодо дискового простору і контролем зареєстрованих користувачів.

Amazon Web Services (AWS). Проект започаткований ще в 2002 році. Це загальна назва для великого набору веб-сервісів і онлайн-застосувань. Більшість з них орієнтована на невеликі компанії, але не виключає роботи поодиноких користувачів, яких зараз вже кілька мільйонів.

Salesforce.com, компанія продає кілька корисних сервісів для бізнесу за методологією SaaS. Основним продуктом є онлайн система *Salesforce CRM (Customer Relationship Management)*.

OnLive. Новий цікавий проект в досить закритій від інших і самодостатній галузі ІТ і розваг – комп'ютерних іграх, що надає сервіси цифрової дистрибуції комп'ютерних ігор.

NetSuite – компанія, що надає послуги доступу до застосувань для середнього бізнесу на принципах SaaS. Послугами NetSuite на початку 2009 користувалося близько 6600 компаній. Компанія надає послуги доступу до бізнес-системи, що дозволяє управляти всією діяльністю підприємства, включаючи CRM (Управління взаємовідносинами з клієнтами), ERP (Управління ресурсами підприємства), "Електронна торгівля" (Електронна комерція) і PSA (Управління проектами).

Технологія ХО

Технологія ХО стосується як до застосування, що пропонується як сервіс Інтернет, так і до обладнання, на якому воно працює, та системного програмного забезпечення в датацентрах, які надають послуги в хмарі. Подібні сервіси раніше створювались на основі парадигми Програма-як-Сервіс (Software as a Service – SaaS). Спеціалізовану організацію програмного забезпечення дата центру для реалізації тако типу сервісів ми називемо Хмарою.

Коли вони пропонуються як передплачені сервіси вільного доступу типу плати-і-користуйся, їх називають публічною хмарою (Public Cloud). Можна продавати й застосування як публічні послуги (Utility Computing) на кшталт

житлово-комунальних послуг. Приватна хмара (Private Cloud) – внутрішні датацентри, що пропонують сервіси для бізнесу і недоступні широкій публіці. Таким чином, хмарні обчислення це сума SaaS та Utility Computing без приватних хмар.

Переваги SaaS як для користувачів, так і для провайдерів добре зрозумілі. Сервіс-провайдери користуються простою процедурою інсталяції програмних систем, їх обслуговуванням, підтримкою та версійним контролем. Хмарні обчислення дають можливість провайдерам пропонувати свої послуги як SaaS, не вкладаючи гроші у розвиток датацентрів. Хмарні обчислення дають можливість розробляти SaaS застосування і збільшувати їх потужність без створення або оренди власних датацентрів. Аналогічно до того, як SaaS дав можливість користувачам (end user) перекласти значну частину проблем на провайдера, так само ХО дають можливість SaaS-провайдерам (SaaS Provider/CloudUser) перекласти велику кількість проблем і витрат на провайдерів хмарних обчислень.

З апаратної точки зору, два аспекти є новими в ХО:

1. Створюється ілюзія нескінченних обчислювальних ресурсів, доступних за запитом, що усуває необхідність планувати спеціалізовану апаратну підготовку майбутньої роботи. Ресурси надаються за потребою (на вимогу).

2. Здатність платити за використання обчислювальних ресурсів на короткотерміновій основі в міру необхідності (наприклад за процесорну потужність за кожну годину роботи і дисковий простір в день) і звільняти їх в міру необхідності, дозволяючи не платити за ресурси, коли вони непотрібні.

У хмарних обчисленнях апаратні ресурси надаються як сервіс (IaaS) – Infrastructure as a Service. Цією інфраструктурою, що хмарний провайдер надає вам як сервіс, може бути що завгодно – від звичайного сервера, до операційної системи або API від третіх осіб.

Нові можливості

Економічні причини ведуть до того, що дані будуть розміщуватися поряд з програмами, що їх оброблюють, оскільки витрати на створення розподілених мереж знижуються повільніше (і залишаються відповідно дорожчими), ніж інші витрати на ІТ обладнання.

Накреслимо типи застосувань, які можуть становити економічну доцільність і перспективи.

1. **Інтерактивні застосування.** Зрозуміло, що майбутнє належить сервісам, які обробляють у реальному часі інформацію, що надходить від користувачів або сенсорів. Такі сервіси можуть бути привабливими для хмари насамперед тому, що загалом добре працюють з великими масивами інформації, які зберігаються у великих ДЦ.

2. **Паралельна пакетна обробка даних.** Хмара створює унікальну можливість швидкої паралельної пакетної обробки даних. Якщо у даних досить паралелізму у внутрішній структурі інформації і система хмари здатна забезпечити таку обробку, то користувачі можуть використати сотні серверів на короткий термін за ціною кількох серверів за довгий термін.

3. **Розвиток і підйом аналітики.** Окремим випадком паралельної пакетної обробки даних є бізнес-аналітика. Якщо раніше більша частина індустрії баз даних була зосереджена на розвитку і підтримці транзакцій, то зараз ця тенденція змінюється до бізнес-аналітики.

4. **Розширення ресурсомістких десктопних застосувань.** Хоча могло скластись враження, що класичні десктопні та хмарні програмні комплекси жорстко конкурують між собою і одночасне їх використання в межах однієї системи не є можливим, насправді це не зовсім відповідає дійсності. Останні версії потужних математичних програмних комплексів Matlab та Mathematica можуть використовувати хмарні обчислення для збільшення швидкості обчислень. Інші десктопні застосування так само можуть з часом стати нерозривним цілим з хмарою не лишаючи можливості працювати у режимі офлайн. Цікавою альтернативною моделлю може бути зберігання даних у хмарі та створення потужного товстого клієнту на стороні клієнта, що буде відповідати за візуалізацію інформації на відповідний графічний інтерфейс. Також офлайновий рендерінг або 3D анімація може бути простим прикладом. Після створення опису об'єктів сцени та характеристик світла у звичайні десктопні студії типу 3DMax чи Maya, побудова зображення розпаралелюється ресурсами хмари.

Будь-який серйозний аналіз хмарних обчислень, як і будь-якої технології, що швидко розвивається повинен враховувати переваги і недоліки. Так що є добре і що погано у хмарних обчисленнях?

Переваги

- Непотрібен сучасний, потужний (і, відповідно, дорогий) комп'ютер для виконання складної обробки інформації.
- Комп'ютери в обчислювальній системі хмари будуть завантажуватися швидше, і працювати швидше, оскільки вони матимуть менше програм і процесів, завантажених у пам'ять.
- Менші витрати на IT інфраструктуру за рахунок використання обчислювальних ресурсів хмари для доповнення або заміни внутрішніх комп'ютерних ресурсів.
- Хмарні обчислення значно скорочують апаратне та програмне обслуговування для організацій усіх розмірів.
- Менші витрати на програмне забезпечення.
- При кожному зверненні до веб-застосування, можна отримати останню версію без необхідності сплати за неї і завантаження оновлення.
- Збільшення обчислювальних потужностей за потребою (на вимогу).
- Хмара пропонує віртуально необмежений простір для зберігання інформації.
- Надійність збереження даних, комп'ютерний збій у хмарі не призведе до втрати важливих даних.
- У хмарі немає жодного значення, якою операційною системою ви користуєтесь.
- Покращена сумісність форматів документів. Не існує жодних форматних несумісностей, коли всі користувачі і всі документи знаходяться в хмарі.
- Можливість багатьох користувачів легко організувати спільну роботу над документами і проектами.
- Хмара завжди має останню версію документа.
- І нарешті, головна і незаперечна перевага хмарних обчислень – відсутність належності до одного комп'ютера чи мережі. Документи і програми залишаються однаковими незалежно від того, за яким комп'ютером ви працюєте.

Недоліки та ризики ХО

Розглянемо ризики хмарних обчислень.

- Потребує постійного підключення до Internet. Хоча треба відмітити, що деякі останні веб-розробки

проектуються таким чином, щоб можна було з працювати і режимі офлайну, а потім завантажувати результат у фінальній частині роботи. Таким чином наприклад працює Google Gears, що перетворює веб-застосування Google у локальні застосування.

- Малоефективна робота при низькій швидкості каналу зв'язку. Навіть використовуючи швидкий канал, веб-застосування може іноді бути повільнішим за звичайну програму на вашому PC, тому, що все у програмі, від інтерфейса до документа, над яким ви працюєте має комунікувати з комп'ютерами у хмарі.

- Можливості можуть бути обмежені. Даний конкретний недолік поступово змінюється, але сьогодні багато хмарних систем просто не в змозі надати такі ж можливості як їх десктопні брати. Порівняйте, наприклад, можливості Google Presentations з Microsoft PowerPoint. Тим не менш, з часом веб-застосування розширюватимуть свою функціональність. Це, безсумнівно, було у випадку з Google Docs і Spreadsheets. Обидві почали з кількох простих функцій, але згодом впровадили багато з тих можливостей, що є у Microsoft Word&Excel. Тим не менше, треба дивитись на функції, перш ніж зробити хід. Переконайтеся, що хмарні застосування можуть зробити все, що вам потрібно, перш ніж ви змініте традиційні програми на хмару.

- Проблема безпеки даних. З хмарними обчисленнями всі ваші дані зберігаються у хмарі, але наскільки хмара безпечна? Чи може неавторизований користувач отримати доступ до ваших конфіденційних даних?

- Створення хмарної системи це серйозна технічна проблема, як з апаратної, так і з програмної точки зору. Спочатку потрібно вкласти великі кошти у створення хмарних систем і розробку хмарних веб-застосувань. А як потім заробляти гроші на хмарних сервісах?

- Хмарні сервіси доступні тільки при наявності постійного швидкісного доступу в Інтернет.

- Наскільки безпечною є хмара? Якщо ви зберігаєте важливий документ у веб, чи ви гарантуєте що він буде вам доступний наступного разу? Компанії, що пропонують сервіси хмарних обчислень говорять, що їх

сервіси абсолютно безпечні. Але зрозуміло, що це зовсім не так.

• І, нарешті, є велике питання чи готове комп'ютерне суспільство довіряти документам і програмам якими вони не зовсім "володіють". Існує ефект "ковдри над головою" коли ви знаєте, що ваші сервіси і документи фізично розташовані на іншому комп'ютері, ніж коли вони тут і ви можете "доторкнутись до них руками". Відчуття того, що ваша інформація зберігається десь у хмарі, але точно невідомо де, може викликати досить дивне відчуття і різні реакції від тих, хто раніше з цим не стикався. Зміна парадигми думки і сприйняття нової концепції власності на електронну інформацію і питання її розташування та зберігання вимагає від світового комп'ютерного товариства досить значних зусиль. Можна стверджувати, що, хмарні обчислення – руйнівна технологія. Вона зруйнує багато софтверних компаній і систем, що будуть не здатні швидко змінитись, і багато існуючих компаній будуть ставитись антагоністично до концепції.

Майбутнє ХО

Омріяна і довгоочікувана ідея доступних комп'ютерних послуг стає реальністю. Еластичність хмари дозволяє розв'язувати завдання бізнесу і надавати користувачам сервіси значно простіше і в коротші терміни. ДЦ отримують можливість надавати свої послуги значно більшій кількості юридичних і фізичних осіб.

Розробники зможуть успішно думати про наступні генерації своїх продуктів у співпраці з технологіями хмарних обчислень. Загалом під особливою увагою має бути горизонтальна масштабованість систем для переходу від сотень до тисяч віртуальних машин. Але є деякі спеціальні моменти.

Програмні застосування майбутнього матимуть частину, що працює на клієнтській машині та частину, що працює у хмарі, причому хмарна частина має швидко розширюватись вгору для роботи з тисячами серверів в разі потреби так само легко, як і зменшуватись вниз для роботи на одній віртуальній машині. Вимогою до клієнтів буде можливість ефективно працювати і виконувати корисну роботу, хоч і з

обмеженою функціональністю навіть після від'єднання від хмари, що зараз не на часі для багатьох застосувань Web 2.0. Такі програми звичайно муситимуть мати відповідну ліцензію для їх використання у межах хмарної системи.

Інфраструктурне програмне забезпечення майбутнього має бути пристосовано до того, що відтепер воно працює на віртуальних машинах. Більш того, воно потребує вбудованої платіжної системи.

Системи обладнання майбутнього мають розроблятись радше як контейнери багатьох сутностей, ніж як місце для одного сервера чи стійки, оскільки це мінімум, при якому така система буде ефективною. Мають бути розроблені системи гнучкого керування енергетичним забезпеченням, щоб зробити можливим переведення у режим енергозбереження одразу низку серверів разом з усією пам'яттю, мережею і системами зберігання аналогічно, як ми це можемо зробити зараз з монітором і процесором. Зокрема, такі системи мають розроблюватись з розрахунку на те, що найменшим об'єктом керування скоріше буде віртуальна машина а не операційна система. А це вимагає іншого підходу до організації ієрархії пам'яті між DRAM та диском. І, нарешті, ми потребуємо покращення у ширині смуги пропускання та вартості як мережного обладнання ДЦ, так і маршрутизаторів розподілених мереж.

Хоча ми зараз дуже оптимістично ставимось до майбутнього хмарних обчислень, треба подивитись у майбутнє і спрогнозувати якими популярними вони будуть через 5 років.

Висновки

У роботі було проведено загальний і огляд нової технології, її ієрархічної архітектури та внутрішньої структури. Наведено перелік можливих застосувань ХО з метою уніфікації різноманітних думок, уявлень і суперечок про ХО.

Був проведений аналіз переваг і недоліків хмарних обчислень. Виділено і групи користувачів для яких нова технологія могла б стати у пригоді. Проведено порівняння хмарних обчислень з пограничними технологіями.

Список використаних джерел

1. Above the Clouds: A Berkeley View of Cloud Computing – Technical Report, Electrical Engineering and Computer Sciences University of California at Berkeley
<http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html> (verified at 03.06.2010)
2. Fingar, Peter. Dot Cloud: The 21st Century Business Platform Built on Cloud Computing
3. Michael Miller. Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online. 2009 by Que Publishing
4. Anthony T. Velte, Toby J. Velte, Ph.D., Robert Elsenpeter. Cloud Computing: A Practical Approach. 2010 by The McGraw-Hill Companies
5. Jon Pyke, Chief Strategy Officer, Cordys. White paper. Now is the Time to Take the Cloud Seriously.
6. Introduction to Cloud Computing architecture. White Paper. Sun Microsystems, Inc. 1st Edition, June 2009
7. George Reese. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud. Published by O'Reilly Media, 1 edition (April 3, 2009)
8. Ian Foster, Yong Zhao, Ioan Raicu, Shiyong Lu. Cloud Computing and Grid Computing 360-Degree Compared. Grid Computing Environments Workshop, 2008. GCE '08. 12-16 Nov. 2008. Austin, TX. Print ISBN: 978-1-4244-2860-1. Date of Current Version: 06 January 2009. On page(s): 1 – 10
9. Judith M. Myerson. Cloud computing versus grid computing.
www.ibm.com/developerworks/web/library/wa-cloudgrid/

Надійшла до редколегії 15.12.10

УДК 519.72

Борисенко О.А., д.т.н., проф., Петров В.В.,
аспірант

Матричне біноміальне кодування

В статті розглянуті матричні біноміальні числа, які отримані на основі лінійних біноміальних чисел. Дані числа дають можливість створювати алгоритми обробки і перетворення інформації, які можуть бути використані при побудові швидкодіючих завадостійких цифрових пристроїв і систем. Виявлені основні властивості матричних біноміальних чисел, наданий метод їх отримання та доведені їх похідні властивості.

Ключові слова: матричні біноміальні числа, завадостійкість, швидкодія, системи числення.

E-mail: electron@sumdu.edu.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

1. Вступ

Матриці - це одне з найпоширеніших зображень комбінаторних об'єктів, серед яких виділяються двійкові матриці. За допомогою таких матриць зображуються блок-схеми, графи, кодові комбінації та інші комбінаторні конфігурації [1]. Однак вони, поряд з перерахованими об'єктами, можуть зображувати й числа [2]. Таке зображення дозволяє одержувати замість лінійних біноміальних чисел, більш складні матричні біноміальні числа в даному випадку в вигляді площинних двійкових біноміальних матриць, а в перспективі й об'ємних. Безумовно, такий запис утрудняє роботу із числами через складність їхнього подання, але в той же час дозволяє підвищити швидкість і завадостійкість обробки цифрової інформації за рахунок наявності в їхніх структурах природної надмірності. Ці властивості матричних біноміальних чисел мають досить важливе значення при побудові на їх основі деяких програмних продуктів, пов'язаних з числовою обробкою інформації, наприклад, довгих чисел, а також при синтезі цифрових пристроїв і систем. Матричні біноміальні числа мають такі ж корисні властивості, що й лінійні біноміальні числа - здатні ефективно стискати й захищати інформацію від несанкціонованого доступу, а

A.A. Borysenko, the professor, V.V. Petrov,
postgraduate

Matrix binomial coding

The article deals with the matrix binomial numbers obtained from linear binomial numbers. These numbers gives capabilities to create algorithms for processing and transformation of information that can be used for building high-speed digital noise immunity devices and systems. The basic properties of binomial matrix numbers was provided, a method of obtaining them and their derivatives properties was given.

Key words: matrix binomial numbers, noise immunity, speed, number systems.

також породжувати комбінаторні об'єкти й, отже, використовуватися, наприклад, в задачах діагностики і комбінаторної оптимізації [5]. Однак все це вони виконують із більшим рівнем завадостійкості й швидкодії, чим звичайні біноміальні числа, щоправда, програючи при цьому проти них в складності.

В основу розглянутих нижче матричних біноміальних чисел покладена лінійна біноміальна система числення [3], у якій числа записуються у вигляді послідовностей двійкових цифр кінцевої довжини $A_i = (a_{j-1}, a_{j-2}, \dots, a_0)$, $i = 0, 1, \dots, P-1$. Кількісний еквівалент числа n -розрядної k -біноміальної системи числення має вигляд:

$$A_i = a_{j-1}C_n^{k-q_j} + \dots + a_l C_n^{k-q_{l+1}} + \dots + a_0 C_n^{k-q_1},$$

де q_0 - число одиниць у біноміальному числі;
 j - кількість розрядів біноміального числа (довжина);

$l = 0, 1, \dots, j-1$ - порядкові номери розряду;

$q_l = \sum_{i=l}^j a_i$, - сума одиничних значень цифр від $(j-1)$ -го розряду до l -го включно;

$q_j = a_j = 0$.

При цьому повинна дотримуватися наступна система обмежень:

Біноміальні нерівномірні числа $n = 5, k = 4$

Біноміальне число	Кількісний еквівалент	Біноміальне число	Кількісний еквівалент
00	$0C_5^4 + 0C_4^4$	10111	$1C_5^4 + 0C_4^3 + 1C_3^3 + 1C_2^2 + 1C_1^1$
010	$0C_5^4 + 1C_4^4 + 0C_3^3$	1100	$1C_5^4 + 1C_4^3 + 0C_3^2 + 0C_2^2$
0110	$0C_5^4 + 1C_4^4 + 1C_3^3 + 0C_2^2$	11010	$1C_5^4 + 0C_4^3 + 0C_3^2 + 1C_2^2 + 0C_1^1$
01110	$0C_5^4 + 1C_4^4 + 1C_3^3 + 1C_2^2 + 0C_1^1$	11011	$1C_5^4 + 1C_4^3 + 0C_3^2 + 1C_2^2 + 1C_1^1$
01111	$0C_5^4 + 1C_4^4 + 1C_3^3 + 1C_2^2 + 1C_1^1$	11100	$1C_5^4 + 1C_4^3 + 1C_3^2 + 0C_2^1 + 0C_1^1$
100	$1C_5^4 + 0C_4^3 + 0C_3^3$	11101	$1C_5^4 + 1C_4^3 + 1C_3^2 + 0C_2^1 + 1C_1^1$
1010	$1C_5^4 + 0C_4^3 + 1C_3^3 + 0C_2^2$	1111	$1C_5^4 + 1C_4^3 + 1C_3^2 + 1C_2^1$
	10110	$1C_5^4 + 0C_4^3 + 1C_3^3 + 1C_2^2 + 0C_1^1$	

$$\begin{cases} q_0 = k, \\ k \leq j \leq n, \\ a_0 = 1. \end{cases}$$

$$\begin{cases} n - k = j - q_0 - 1, \\ 0 \leq q_0 \leq k - 1, \\ a_0 = 0. \end{cases}$$

Діапазон чисел лінійної біноміальної системи числення $P = C_{n+1}^k$.

Особливістю лінійних біноміальних чисел є те, що вони є нерівномірними і, як наслідок, префіксними, тобто жодне з них не є початком іншого (див. табл. 1). Крім того, ці числа задовольняють загальним вимогам до будь-яких систем числення - скінченності, ефективності, однозначності [4]. Виконання цих вимог стосовно лінійних біноміальних систем числення доведено раніше в роботах [2, 3]. Для систем числення характерною вимогою є також можливість виконання основних арифметичних операцій. На даний час стосовно лінійних біноміальних систем числення розроблені лише операції додавання й віднімання одиниці, а загальні алгоритми операцій додавання й віднімання ще перебувають в стадії розробки.

Всі вищевказані властивості систем числення повинні бути присутніми у позиційних системах числення, до яких відносяться й біноміальні системи, як лінійні, так і матричні. Позиційні системи числення мають алфавіт, який повинен починатися з 0 і далі в ньому в зростаючій послідовності повинні йти числа натурального ряду. При наявності хоча б одного пропущеного числа цього ряду в алфавіті, наприклад, нуля система числення, в тому числі і біноміальна, втрачає властивості позиційних систем числення

й перетворюється в більше складне утворення, що зображає вже не систему числення, а систему кодування. Відповідно цифри такої системи числення перетворюються в букви, а числа в слова.

У розглянутій двійковій біноміальній системі числення алфавіт цифр складається із двох цифр - 0 і 1, але є й більш складні багатозначні біноміальні системи числення, що містять у своєму алфавіті більше двох цифр [2, 3]. На цей час вони менш детально розроблені, ніж двійкові біноміальні системи, і ще перебувають в стадії дослідження, хоча вже зараз можна сказати, що за своїми можливостями вони перевершують двійкові біноміальні системи числення.

2. Постановка задачі

До матричного біноміального числа або біноміальної числової матриці відноситься будь-яка матриця, що має вигляд

$$\begin{bmatrix} x_{01} & x_{02} & \dots & x_{0j} & \dots & x_{0k} \\ x_{11} & x_{12} & \dots & x_{1j} & \dots & x_{1k} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{i1} & x_{i2} & \dots & x_{ij} & \dots & x_{ik} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ x_{(n-k)1} & x_{(n-k)2} & \dots & x_{(n-k)j} & \dots & x_{(n-k)k} \end{bmatrix}$$

і яка складається з $(n - k + 1)$ рядків і k стовпців, $n \geq k$, в яких стоять або нулі, або одиниці, порядок розміщення яких задається наданими нижче правилами.

Діапазон матричних біноміальних чисел співпадає з діапазоном відповідних лінійних

біноміальних чисел, з яких вони були отримані - $P = C_{n+1}^k$. Лінійні біноміальні числа перетворюються в матричні біноміальні числа та зворотно за допомогою алгоритмів, які розроблені в [2, 3]. В цих же роботах були виявлені основні властивості біноміальних числових матриць, однак без їхнього поділу на основні, що задані як аксіоми, й похідні, що виведені з основних. Такий поділ дозволив би більш детально і формалізовано дослідити властивості біноміальних числових матриць, а правила, чи метод їх побудови сформулювати більш строго, що важливо при використанні цих матриць на практиці. Саме ці завдання й поставлені в даній роботі, в якості основної задачі, що в ній вирішується.

3. Рішення задачі

В біноміальних числових матрицях повинні виконуватися наступні основні властивості, які можна сприймати як аксіоми:

1. Біноміальна матриця, яка кодує 0, складається тільки з нулів.

2. У кожному стовпці матриці не може бути більше однієї одиниці.

3. Одиниці в матриці розташовуються по рядках так, що перша одиниця перебуває в першому стовпці, а наступні розташовуються по різних стовпцях, як в одній, так і в різних рядках одна за другою без проміжних нулів.

4. Серед елементів діагоналей матриці, спрямованих зліва направо, може перебувати тільки одна одиниця.

Користуючись даними властивостями біноміальних матриць і їх визначенням, доведемо деякі їх похідні властивості.

Твердження 1. Число елементів біноміальної матриці $I = (n - k + 1)k$.

Доведення. Відповідно до визначення, біноміальна матриця складається з $(n - k + 1)$ рядків і k стовпців. Тоді кількість елементів матриці дорівнює $I = (n - k + 1)k$. Твердження доведене.

Твердження 2. Число нулів в біноміальній матриці дорівнює I .

Доведення. Відповідно до першої властивості матриці, матриця яка кодує нуль складається тільки з нулів. Тому максимальне число нулів в матриці, дорівнює I .

Твердження 3. Число одиниць q_M у матриці не перевищує значення k .

Доведення. Оскільки матриця має в своєму складі k стовпців і в кожному стовпці відповідно до властивості 2 не може стояти більше однієї одиниці, то загальна кількість одиниць у матриці не перевершує k . Твердження доведене.

Твердження 4. Число одиниць у рядку біноміальної матриці не перевищує значення k .

Доведення. Рядок матриці за її визначенням не може вмістити більше k двійкових цифр. Тому він не може мати більше k одиниць. Твердження доведене.

Твердження 5. Одиниці в матриці розташовуються так, що одиниця кожного наступного стовпця перебуває або в тім же рядку що й у попередньому стовпці, або в одному з верхніх рядків.

Доведення. Якби це було не так, як зазначено у твердженні, то в силу того, що одиниці йдуть у стовпцях матриці послідовно і без проміжків, то будь-яка b одиниця, що виявилася b нижче попередніх її одиниць утворювала b з ними діагональ із двома і більше одиницями, що за умовою властивості 4 заборонено. Твердження доведене.

Твердження 6. Якщо в першому стовпці відсутня одиниця, то й в інших стовпцях не може бути одиниць.

Доведення. У першому стовпці одиниця може бути відсутньою тільки в матриці, що кодує 0, а вона складається тільки з нулів. Твердження доведене.

Отримані твердження разом з чотирма незалежними властивостями дозволяють контролювати помилки в матриці, а в разі випадків і виправляти їх, тому що вони є наслідком порушень обмежень, які притаманні матриці. Ці властивості матричних біноміальних чисел можуть бути використані при конструюванні цифрових пристроїв та при передачі інформації. Наприклад, поява в стовпці або діагоналі направленої зліва направо двох одиниць є ознакою помилки. Крім того, кожне наступне матричне біноміальне число по відношенню до попереднього з незаповненими до кінця одиницями верхнім рядком відрізняється наявністю в ньому додаткової одиниці. Це дозволяє за рахунок виключення переносів значно збільшити швидкість відповідних алгоритмів перебору матричних біноміальних чисел.

4. Методи формування матричних біноміальних чисел

Наведені вище твердження дозволяють розробити методи формування матричних біноміальних чисел в порядку їх зростання. Він має наступний вигляд.

Числу 0 буде відповідати матриця, яка складається з k стовпців та $(n-k+1)$ рядків, заповнених нулями. Наступна матриця має в першому рядку і першому стовпчику 1. Далі за нею будуть йти матриці, в яких після першої 1 в цьому рядку з'являється друга, третя і так далі доти, поки не буде отримана матриця, в якій весь рядок не буде заповнений k одиницями. Після цього в наступній матриці у перший рядок заносяться нулі, а в другому рядку першого стовпчика з'являється одиниця. Далі будуть йти одна за одною матриці, в яких в першому рядку, починаючи з другого стовпчика, будуть послідовно з'являтися одиниці доти, поки в матриці не з'явиться k одиниць. Після цього, в другому рядку з'являється наступна друга одиниця і далі процедура формування матриць повторюється до того часу, поки не буде сформована матриця, в якій останній рядок буде заповнений одиницями k .

5. Приклад формування біноміальних матричних чисел

Розглянемо біноміальні матричні числа з параметрами $n=5$, $k=4$, які мають наступний вигляд:

Список використаних джерел

1. Тараканов В. Е. Комбинаторные задачи и $(0,1)$ – матрицы / Тараканов В. Е. – М.: Наука, 1985. – 192 с.
2. Борисенко А.А. Введение в теорию биномиального счета / Борисенко А.А.- Сумы: ИТД «Университетская книга», 2004. – 88 с.
3. Борисенко А.А. Биномиальное кодирование / Борисенко А.А., Кулик И.А. – Сумы: Изд-во СумГУ, 2010. – 206 с.

0	1	2	3
$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$
0 0 0 0 0 0	1 0 0 0 0 0	0 1 1 0 0 0	0 1 1 1 0 0
4	5	6	7
$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$
0 1 1 1 1 1	1 0 0 0 0 0	0 1 0 0 0 0	0 1 1 0 0 0
8	9	...	14
$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$	...	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$
1 0 1 1 1 1	1 1 0 0 0 0		1 1 1 1 1 1

В даному випадку матричні біноміальні числа подані у вигляді двійкових матриць, унизу під якими розташовані відповідні їм двійкові біноміальні числа, а вгорі - десяткові числа. Таким чином, одне і теж десяткове число можна записати у вигляді двох різних його біноміальних форм, серед яких найбільш важлива для даної роботи буде матрична форма.

6. Висновки

Таким чином біноміальні матриці зображають собою математичні об'єкти, що кодують числа і мають можливість створювати швидкодіючі і завадостійкі алгоритми кодування та перетворення інформації.

4. Поспелов Д.А. Арифметические основы вычислительных машин дискретного действия. / Поспелов Д.А.- М., «Высшая школа», 1970. – 308 с.

5. Borisenko A.A. Binomials Calculus: Advantages and Prospects. / Borisenko A.A., Kalashnikov V.V. // ISIC Express Letters An International Journal of Research and Surveya.- Japan, Tokai University, page 123-130.

Надійшла до редколегії 09.12.10

УДК 004.655

Буй Д.Б., д.ф.-м. н., с.н.с.,
Глушко* І.М., аспірантка

**Узагальнена таблицна алгебра,
узагальнене числення рядків,
узагальнене числення на домені та їх
еквівалентність**

Проведено узагальнення таблицної алгебри, побудованої на основі реляційних алгебр Кодда; узагальнення полягає в тому, що замість таблиць розглядаються пари, які складаються з таблиць та схем таблиць. Доведено, що узагальнена таблицна алгебра, узагальнене числення рядків та узагальнене числення на домені однаково виразні.

Ключові слова: реляційні (табличні) бази даних, числення рядків, числення на домені, таблицна алгебра.

*E-mail: buy@unicyb.kiev.ua, glushkoim@gmail.com
Статтю представив д.ф.-м.н., проф. Анісімов А.В.

Вступ

Уточнення реляції в термінах іменних множин було здійснене Редьком В.Н., Броною Ю.Й., Буєм Д.Б., Поляковим С.А. Результати роботи опубліковані у монографії [1]. Однією із особливостей такого підходу є те, що в реляції не вноситься схема. У даній роботі отримані результати узагальнено на таблиці, яким приписується певна схема. Оскільки в основі більшості реляційних мов запитів лежить реляційне числення, важливо дослідити зв'язок таблицної алгебри та двох форм реляційного числення: числення рядків (кортежів) та числення на домені. Основний результат роботи: узагальнена таблицна алгебра, узагальнене числення рядків, узагальнене числення на домені еквівалентні.

Узагальнення таблицної алгебри

Всі невизначені тут поняття та позначення розуміємо в смислі монографії [1]. Розглядаємо дві множини: A – множину атрибутів і D – універсальний домен. Під узагальненою таблицною алгеброю розуміємо алгебру $\langle T, \Omega_{P, \Xi} \rangle$, де T – множина усіх таблиць,

D.B. Buy DPhil, senior staff scientist,
I.M. Glushko*, postgraduate student

Generalized table algebra, generalized tuple calculus, generalized domain calculus and theirs equivalence

Generalization of table algebra built on the basis of Codd's relational algebras is conducted. Generalization is that instead of tables we consider pairs consisting of tables and scheme of tables. Proved that table algebra, tuple calculus, and domain calculus are equally expressive.

Key words: relation databases, tuple calculus, domain calculus, table algebra.

$\Omega_{P, \Xi} = \{ \cup_R, \cap_R, \setminus_R, \sigma_{p, R}, \pi_{X, R}, \otimes_{R_1, R_2}^{R_1}, \div_{R_2}^{R_1}, \rho_{\xi, R}, \sim_R \}$ – сигнатура, $p \in P, \xi \in \Xi, X, R, R_1, R_2 \subseteq A, P, \Xi$ – множини параметрів. Довільну (скінченну) множину атрибутів $R \subseteq A$ назовемо схемою. Рядком схеми R називається іменна множина на парі R, D , проекція якої за першою компонентою рівна R (тобто розглядається функція вигляду $s: R \rightarrow D$). Під (узагальненою) таблицею розуміємо пару $\langle t, R \rangle$, де $t \in T(R)$ – таблиця фіксованої схеми R . Тоді $T(R) = \{ \langle t, R \rangle \mid t \in T(R) \}$ – множина усіх (узагальнених) таблиць схеми R , а $T = \bigcup_R T(R)$ – множина всіх таблиць. Відмінність від визначення таблиці, яке подане у [1], полягає у тому, що кожній таблиці приписується певна схема. Це впливає тільки на випадок порожньої таблиці $t_\emptyset$, оскільки за непорожньою таблицею схема відновлюється однозначно. Запис $\langle t_\emptyset, R \rangle$ позначає порожню таблицю схеми R .

Виразом таблицної алгебри називається вираз, побудований з таблиць множини T при використанні операцій з сигнатури $\Omega_{P, \Xi}$.

Задамо операції. Визначимо спочатку теоретико-множинні операції. Під об'єднанням $\cup_R$ (перетином $\cap_R$, різницею $\setminus_R$) таблиць схеми R розуміється (параметрична) бінарна операція, отримана (по суті) обмеженням теоретико-множинного об'єднання (відповідно перетину, різниці) на множину всіх таблиць схеми R . Таким чином, $\cup_R: T(R) \times T(R) \rightarrow T(R)$,
 $\langle t_1, R \rangle \cup_R \langle t_2, R \rangle = \langle t_1 \cup t_2, R \rangle$;
 $\cap_R: T(R) \times T(R) \rightarrow T(R)$,
 $\langle t_1, R \rangle \cap_R \langle t_2, R \rangle = \langle t_1 \cap t_2, R \rangle$;
 $\setminus_R: T(R) \times T(R) \rightarrow T(R)$,
 $\langle t_1, R \rangle \setminus_R \langle t_2, R \rangle = \langle t_1 \setminus t_2, R \rangle$; де $t_1, t_2 \in T(R)$.

Нехай $p: S \rightarrow \{true, false\}$ – частковий предикат на множині рядків. Під селекцією за предикатом p таблиць схеми R розуміється унарна часткова параметрична операція $\sigma_{p,R}$, яка таблиці зіставляє її підтаблицю, що містить рядки, на яких предикат p істинний. Отже,
 $\sigma_{p,R}: T \rightarrow T$, $dom \sigma_{p,R} = \{\langle t, R \rangle \mid t \subseteq dom p\}$,
 $\sigma_{p,R}(\langle t, R \rangle) = \langle \{s \mid s \in t \wedge p(s) = true\}, R \rangle$, де
 $t \in T(R)$, $=$ – узагальнена рівність (тобто обидві частини або одночасно невизначені або одночасно визначені і рівні). Областю означеності операції селекції є таблиці

$$(\langle t_1, R_1 \rangle \otimes_{R_1, R_2} \langle t_2, R_2 \rangle) \otimes_{R_1 \cup R_2, R_3} \langle t_3, R_3 \rangle = \langle t_1, R_1 \rangle \otimes_{R_1, R_2 \cup R_3} (\langle t_2, R_2 \rangle \otimes_{R_2, R_3} \langle t_3, R_3 \rangle).$$

Проте для простоти читання виразів, застосовуючи дану операцію до трьох і більше таблиць, вважаємо, що дужки розставлені зліва направо, і тому ми їх опускаємо. Таким чином, запис $\langle t_1, R_1 \rangle \otimes_{R_1, R_2} \langle t_2, R_2 \rangle \otimes_{R_1 \cup R_2, R_3} \langle t_3, R_3 \rangle$ означає $(\langle t_1, R_1 \rangle \otimes_{R_1, R_2} \langle t_2, R_2 \rangle) \otimes_{R_1 \cup R_2, R_3} \langle t_3, R_3 \rangle$.

Під діленням таблиць схеми R_1 на таблиці схеми R_2 , де $R_2 \subseteq R_1$, розуміється бінарна параметрична операція $+_{R_2}^{R_1}$, значеннями якої є таблиці схеми $R_1 \setminus R_2$, що складаються з певних рядків, отриманих проекцією за множиною атрибутів $R_1 \setminus R_2$ таблиць схеми R_1 .

Отже, $+_{R_2}^{R_1}: T(R_1) \times T(R_2) \rightarrow T(R_1 \setminus R_2)$,

$$\langle t_1, R_1 \rangle +_{R_2}^{R_1} \langle t_2, R_2 \rangle = \left\langle \{s \in \pi_{R_1 \setminus R_2}(t_1) \mid \{s\} \otimes_{R_1 \setminus R_2, R_2} t_2 \subseteq t_1\}, R_1 \setminus R_2 \right\rangle, \text{ де } t_1 \in T(R_1), t_2 \in T(R_2).$$

$\langle t, R \rangle \in T(R)$, які містять рядки $s \in S(R)$, на яких предикат-параметр селекції p визначений. Надалі розглядаємо тільки предикати-параметри вигляду $p(s) = true \Leftrightarrow p(s(A_1), \dots, s(A_m)) = true$, де p – предикат на універсальному домені.

Нехай $X \subseteq A$ – (скінченна) множина атрибутів. Під проекцією за множиною атрибутів X таблиць схеми R розуміється унарна параметрична операція $\pi_{X,R}$, значеннями якої є таблиці схеми $R \cap X$, що складаються з обмежень за X рядків вихідних таблиць. Отже,
 $\pi_{X,R}: T(R) \rightarrow T(R \cap X)$,
 $\pi_{X,R}(\langle t, R \rangle) = \langle \{s \mid X \mid s \in t\}, R \cap X \rangle$, де $t \in T(R)$.

Під з'єднанням таблиць схем R_1 та R_2 розуміється бінарна операція $\otimes_{R_1, R_2}$, значеннями якої є таблиці схеми $R_1 \cup R_2$, що складаються з усяких об'єднань сумісних рядків вихідних таблиць. Отже, $\otimes_{R_1, R_2}: T(R_1) \times T(R_2) \rightarrow T(R_1 \cup R_2)$,
 $\langle t_1, R_1 \rangle \otimes_{R_1, R_2} \langle t_2, R_2 \rangle = \langle \{s_1 \cup s_2 \mid s_1 \in t_1 \wedge s_2 \in t_2 \wedge s_1 \approx s_2\}, R_1 \cup R_2 \rangle$, де $t_1 \in T(R_1)$, $t_2 \in T(R_2)$. Операція з'єднання асоціативна (див., наприклад, [1], [2]):

Під активним доповненням таблиць схеми R розуміється унарна операція $\sim_R$, яка таблиці зіставляє доповнення в її насиченні. Отже, $\sim_R: T(R) \rightarrow T(R)$, $\sim_R \langle t, R \rangle = \langle C(t) \setminus t, R \rangle$, де $t \in T(R)$.

Під перейменуванням таблиць схеми R , що відповідає ін'єктивній частковій функції перейменування атрибутів $\xi: A \rightarrow A$, розуміється унарна параметрична операція $Rt_{\xi,R}$, $dom Rt_{\xi,R} = \{\langle t, R \rangle \mid t \in T_{\xi}(R)\}$, де $T_{\xi}(R)$ – множина таблиць схеми R , а схема R ξ -допустима, значення якої задаються таким чином:
 $Rt_{\xi,R}(\langle t, R \rangle) = \langle Rs_{\eta,R}[t], \eta[R] \rangle$, $t \in T_{\xi}(R)$, де
 $\eta = \xi \cup id_{A \setminus dom \xi}$, $Rs_{\eta,R}(s) = \{\langle \eta(A), s(A) \rangle \mid A \in R\}$,
 $s \in S(R)$.

Дамо означення виразу табличної алгебри індукцією за довжиною виразів, приписуючи кожному виразу його схему.

1. Кожна стала таблиця схеми R , є виразом табличної алгебри схеми R .

2. Кожна змінна таблиця схеми R , є виразом табличної алгебри схеми R .

3. Якщо F_1 та F_2 – вирази табличної алгебри схеми R , то $F_1 \cup_R F_2$, $F_1 \cap_R F_2$, $F_1 \setminus_R F_2$ – вирази табличної алгебри схеми R .

4. Якщо F – вираз табличної алгебри схеми R , то $\sigma_{p,R}(F)$ – вираз табличної алгебри схеми R .

5. Якщо F – вираз табличної алгебри схеми R , то $\pi_{X,R}(F)$ – вираз табличної алгебри схеми $R \cap X$.

6. Якщо F_1 та F_2 – вирази табличної алгебри схем R_1 та R_2 відповідно, то $F_1 \otimes_{R_1, R_2} F_2$ – вираз табличної алгебри схеми $R_1 \cup R_2$.

7. Якщо F_1 та F_2 – вирази табличної алгебри схем R_1 та R_2 відповідно, то $F_1 \div_{R_2}^{R_1} F_2$ – вираз табличної алгебри схеми $R_1 \setminus R_2$.

$$\bigcup_{i=1}^n \left(\left\langle \left\langle A_1, d_{i1} \right\rangle \right\rangle_{\{A_1\}, \{A_2\}} \otimes \left\langle \left\langle A_2, d_{i2} \right\rangle \right\rangle_{\{A_1, A_2\}, \{A_3\}} \otimes \left\langle \left\langle A_3, d_{i3} \right\rangle \right\rangle_{\{A_1, A_2, A_3\}, \{A_4\}} \dots \otimes \left\langle \left\langle A_m, d_{im} \right\rangle \right\rangle_{\{A_1, \dots, A_{m-1}\}, \{A_m\}} \right)$$

Покажемо, що операції перетину, ділення, активного доповнення можна виразити через операції, зазначені у формулюванні леми. Дійсно,

$$\langle t_1, R_1 \rangle \div_{R_2}^{R_1} \langle t_2, R_2 \rangle = \pi_{R', R_1}(\langle t_1, R_1 \rangle) \setminus_{R'} \pi_{R', R_1}(\pi_{R', R_1}(\langle t_1, R_1 \rangle) \otimes_{R', R_2} \langle t_2, R_2 \rangle \setminus_{R_1} \langle t_1, R_1 \rangle),$$

де $R_2 \subseteq R_1$, $R' = R_1 \setminus R_2$; $\sim_R \langle t, R \rangle = \langle C(t) \setminus I, R \rangle$,

$$C(\langle t, R \rangle) = \pi_{A_1, R}(\langle t, R \rangle) \otimes_{\{A_1\}, \{A_2\}} \dots \otimes_{\{A_1, \dots, A_{n-1}\}, \{A_n\}} \pi_{A_n, R}(\langle t, R \rangle),$$

де $R = \{A_1, \dots, A_n\}$ – схема таблиці t (див., наприклад, [1], [2]). $\square$

Узагальнене числення рядків

В основі більшості реляційних мов запитів лежить реляційне числення. На відміну від реляційної алгебри числення виражає лише, яким повинен бути результат і не передбачає визначення того, як його отримати. Реляційне числення ґрунтується на численні предикатів першого порядку. Є дві форми реляційного числення: числення зі змінними-рядками та числення зі змінними на доменах. Ці форми запропоновані Е. Коддом (E. Codd) [3] та М. Лакруа (M. Lacroix) і А. Піротте (A. Pirotte) [4] відповідно. Дане питання також розглядали Д. Мейєр (D. Maier) [2], Дж.Д. Ульман (J.D. Ullman)

8. Якщо F – вираз табличної алгебри схеми R , то $\sim_R(F)$ – вираз табличної алгебри схеми R .

9. Якщо F – вираз табличної алгебри схеми R , то $R_{\xi, R}(F)$ – вираз табличної алгебри схеми $\eta[R]$.

10. Інших виразів, крім зазначених у пунктах 1-9, немає.

Лема 1. Будь-який вираз табличної алгебри можна замінити еквівалентним йому виразом, який використовує сталі таблиці з єдиним атрибутом і єдиним рядком, операції селекції, з'єднання, проекції, об'єднання, різниці й перейменування.

Доведення.

Справді, нехай $\langle t, R \rangle$ – стала таблиця, де $t = \{s_1, \dots, s_n\}$, $R = \{A_1, \dots, A_m\}$, причому $s_i = \left\langle \left\langle A_1, d_{i1} \right\rangle, \dots, \left\langle A_m, d_{im} \right\rangle \right\rangle$, $i = 1, \dots, n$. Замінімо дану таблицю виразом

маємо рівності:
 $\langle t_1, R \rangle \cap_R \langle t_2, R \rangle = \langle t_1, R \rangle \setminus_R (\langle t_1, R \rangle \setminus_R \langle t_2, R \rangle)$,

[5], К.Дж. Дейт (C.J. Date) [6], Т.М. Коннолі (T.M. Connolly) та К. Бегг (C.E. Begg) [7].

Вирази числення рядків мають вигляд $\{x(R) | P(x)\}$, де P – деякий предикат над змінним рядком x , а R – схема. Цей вираз позначає таблицю $\langle t, R \rangle$, $t \in T(R)$, що містить рядки, на яких предикат P істинний.

Введемо множину (так званих) дозволених формул числення рядків при використанні:

- множини атрибутів A і універсального домену D ;
- множини предметних змінних (змінних рядків) $x_1, x_2, \dots$;
- множини предметних констант $d_1, d_2, \dots$;
- множини функціональних символів $f_1^{n_1}, f_2^{n_2}, \dots$;

• множини предикатних символів
 $p_1^{m_1}, p_2^{m_2}, \dots$

Областю інтерпретації предметних констант є універсальний домен D , предметних змінних – множина всіх рядків. Введемо синтаксичні змінні як у [8]. Використовуємо x як синтаксичну змінну, областю зміни якої є предметні змінні; $f(p)$ як синтаксичну змінну, областю зміни якої є функціональні (відповідно предикатні) символи; d як синтаксичну змінну, областю зміни якої є предметні константи; A як синтаксичну змінну, областю зміни якої є атрибути.

Наступні вирази є термами (індукція за довжиною термів):

- d – терм;
- $x(A)$ – терм;
- якщо $u_1, \dots, u_n$ – терми, f – n -арний функціональний символ, то $f(u_1, \dots, u_n)$ – терм;
- вираз є термом, тільки тоді, коли це слідує з правил а), б), в).

Будемо використовувати u як синтаксичну змінну, областю зміни якої є терми.

Сформулюємо правила побудови формул. Атомарні формули (атоми) можуть бути двох типів.

а1. Нехай $\langle t, R \rangle$ – таблиця, а x – змінний рядок. Тоді $t(x)$ – атом, який (при інтерпретації) означає, що $x \in t$.

а2. Нехай $u_1, \dots, u_m$ – терми, а p – m -арний предикат на універсальному домені D . Тоді $p(u_1, \dots, u_m)$ – атом.

Використаємо логічні зв'язки $\neg, \wedge, \vee$, квантори $\exists, \forall$ та дужки $(,)$ для побудови формул із атомів.

Будемо використовувати P, Q та G як синтаксичні змінні, областю зміни яких є формули, визначені нижче.

- Кожний атом – це формула.
- Якщо P – формула, то $\neg P$ – теж формула.
- Якщо P і Q – формули, то $(P \wedge Q), (P \vee Q)$ – формули.
- Нехай x – змінний рядок, P – формула, $R \subseteq A$ – схема, тоді $\exists x(R)P$ – формула.
- Нехай x – змінний рядок, P – формула, $R \subseteq A$ – схема, тоді $\forall x(R)P$ – формула.
- Якщо P – формула, то (P) – теж формула.
- Інших формул немає.

В загальному випадку змінні рядки у формулах можуть бути вільні або зв'язані. Смысл цих понять такий самий як і у численні предикатів: входження змінної x у дану формулу називається зв'язаним, якщо x є змінною квантора ($\exists$ або $\forall$), який входить у дану формулу, або знаходиться області дії даного квантора; в протилежному випадку входження змінної x у дану формулу називається вільним. Змінна називається вільною (зв'язаною) змінною в даній формулі, якщо існують вільні (відповідно зв'язані) її входження в цю формулу (див., наприклад, [9]).

Для кожного змінного рядка x визначимо схему (скінченну множину атрибутів) $scheme(x, P)$ та множину атрибутів $attr(x, P)$, з якими рядок x зустрічається у формулах. Вирази $scheme(x, P)$ та $attr(x, P)$ визначені, якщо рядок x має вільне входження в формулу P , причому має місце включення $attr(x, P) \subseteq scheme(x, P)$ (що впливає з подальших означень, за умови визначення виразів).

Виділимо клас дозволених формул. Визначимо значення $attr$ спочатку для термів:

- якщо $u = d$, то $attr(x, u) = \emptyset$;
- якщо $u = x(A)$, то $attr(x, u) = \{A\}$, а $attr(x, y(A)) = \emptyset$, якщо $x \neq y$;
- якщо $u = f(u_1, \dots, u_n)$, де u_i – терми, то $attr(x, u) = \bigcup_{i=1}^n attr(x, u_i)$.

Нехай формула P – атом, тоді

а1. якщо $P = t(x)$, то (єдине) входження змінного рядка x є вільним у формулі P і $scheme(x, P) = attr(x, P) = R$, де R – схема таблиці t ;

а2. якщо $P = p(u_1, \dots, u_m)$, де u_i – терми, причому $x_1, \dots, x_k$ – всі змінні цих термів, то входження цих змінних рядків є вільними у формулі P , схема $scheme(x, P)$ не визначена, а

$$attr(x, P) = \bigcup_{j=1}^m attr(x, u_j), \quad i = 1, \dots, k.$$

Атомарні формули завжди дозволені. Для побудови всіх дозволених формул проведемо індукцію за довжиною формул. Припустимо, що G і Q дозволені формули.

f1. Якщо $P = \neg G$, то P – дозволена формула, а входження змінних у P вільні або зв'язані, залежно від того, вільні або зв'язані входження цих самих змінних у G . Якщо x входить у формулу G вільно, то

$$scheme(x, P) \approx scheme(x, G)$$

$$attr(x, P) = attr(x, G).$$

f2. Якщо $P = G \wedge Q$ або $P = G \vee Q$, то входження змінних у P вільні або зв'язані, залежно від того, вільні або зв'язані входження цих змінних у G або Q . Нехай змінний рядок x входить у підформули G та/або Q вільно. Визначимо схему та множину атрибутів, з якими рядок x зустрічається у формулах, для формули P . Мають місце наступні випадки.

а. Схеми формул $scheme(x, G)$ та $scheme(x, Q)$ визначені. Для того, щоб формула P була дозволена, має виконуватися рівність $scheme(x, G) = scheme(x, Q)$. Покладаємо за означенням $scheme(x, P) = scheme(x, G)$.

б. Схема визначена лише для однієї з підформул. Нехай схема $scheme(x, G)$ визначена, а схема $scheme(x, Q)$ – невизначена. Для того, щоб формула P була дозволена, має виконуватися включення $attr(x, Q) \subseteq scheme(x, G)$. Покладаємо за означенням $scheme(x, P) = scheme(x, G)$.

с. Схема невизначена для обох підформул. У цьому випадку і схема $scheme(x, P)$ невизначена.

Для будь-якого з розглянутих випадків $attr(x, P) = attr(x, G) \cup attr(x, Q)$.

f3. Якщо $P = \exists x(R)G$ і змінна x входить у формулу G вільно, то P – дозволена формула. Крім того, якщо $scheme(x, G)$ визначена, то повинна виконуватися рівність $scheme(x, G) = R$ за умови виконання включення $attr(x, G) \subseteq R$. Оскільки змінна x не входить вільно у формулу P , то $scheme(x, P)$ і $attr(x, P)$ невизначені. Якщо $y \neq x$, то будь-яке входження змінної y в P вільне або зв'язане, залежно від того, вільне або зв'язане входження y в G . Якщо y входить в P вільно, то $scheme(y, P) \approx scheme(y, G)$ і $attr(y, P) = attr(y, G)$.

f4. Якщо $P = \forall x(R)G$, то всі визначення і обмеження такі ж, як і у попередньому випадку для квантора існування.

f5. Якщо $P = (G)$, то P дозволена формула, а вільні та зв'язані входження змінних, схема і множина атрибутів, з якими змінний рядок зустрічається у формулах, залишаються такими як і для G .

Після введення множини дозволених формул можемо дати остаточне визначення виразу

і числення рядків. Отже, вираз числення рядків має вигляд $\{x(R) | P(x)\}$, де

1. формула P – дозволена;
2. змінна x – єдина змінна, яка входить у формулу P вільно;
3. якщо $scheme(x, P)$ визначена, то $scheme(x, P) = R$, інакше, $attr(x, P) \subseteq R$.

Нехай $P(x)$ – дозволена формула, $R \subseteq A$. В результаті підстановки конкретного рядка s схеми R замість x у формулу P отримаємо формулу $P(s/x)$. Надамо спочатку значення істинності атомам:

a1. нехай рядок x у підформулі $t(x)$ вільний в P . Атом $t(x)$ набуває значення істина при підстановці конкретного рядка s замість x , якщо $s \in t$, інакше атом $t(x)$ набуває значення хиба;

a2. нехай рядок x в підформулі $p(u_1, \dots, u_m)$ вільний в P , тоді при підстановці конкретного рядка s замість x , замінимо $x(A_i)$ на $d_i \in D$, де $\langle A_i, d_i \rangle \in s$ (d_i значення атрибута A_i в рядку s). Атом $p(u_1, \dots, u_m)$, де u_i – терми (предметні константи), набуває значення істина, якщо предикат p істинний на відповідних значеннях, інакше атом набуває значення хиба.

Набір значень істинності всіх атомів формули називають її інтерпретацією [10]. Нехай P дозволена формула без вільних змінних. Інтерпретація формули P визначається наступним чином.

f1. Якщо $P = \neg G$, то в G немає вільних змінних. Формула P істинна, коли G хибна, і хибна, коли формула G істинна.

f2. Якщо $P = G \wedge Q$ або $P = G \vee Q$, то в G і Q немає вільних змінних. Якщо $P = G \wedge Q$, то формула P істинна тоді, коли G та Q водночас істинні, у всіх інших випадках формула P хибна. Якщо $P = G \vee Q$, то формула P хибна тоді, коли G та Q водночас хибні, у всіх інших випадках формула P істинна.

f3. Якщо $P = \exists x(R)G$, то x – єдина змінна, яка входить у формулу G вільно. Формула P істинна, якщо є принаймні один рядок $s \in S(R)$, такий, що формула $G(s/x)$ – істинна, інакше формула P хибна.

f4. Якщо $P = \forall x(R)G$, то x – єдина змінна, яка входить у формулу G вільно. Формула істинна, якщо для кожного рядка $s \in S(R)$

формула $G(s/x)$ істинна, інакше формула P хибна.

f5. Якщо $P=(G)$, то формула P істинна, якщо формула G істинна і хибна, якщо формула G хибна.

Нехай $E = \{x(R) | P(x)\}$ – вираз числення рядків. Значенням виразу E назвемо таблицю схеми R , яка містить всі рядки $s \in S(R)$ такі, що формула $P(s/x)$ істинна.

Покажемо, що для довільного виразу табличної алгебри можна побудувати еквівалентний вираз числення рядків. Вираз числення рядків E еквівалентний виразу табличної алгебри F , якщо значення виразу E співпадає із значенням виразу F , за умови, коли табличні константи і табличні змінні інтерпретуються однаково в цих двох виразах.

Теорема 1. Якщо F – вираз табличної алгебри, то можна ефективно (і рівномірно) побудувати еквівалентний йому вираз E числення рядків.

Доведення. Згідно леми 1 при доведенні теореми розглядаємо вирази табличної алгебри, які містять тільки операції об'єднання, різниці, селекції, проєкції, з'єднання та перейменування, причому допускаються тільки одноатрибутні сталі таблиці з одним рядком. Доведення проведемо індукцією за числом операцій в F .

База індукції (немає операцій). Можливі два випадки. По-перше, $F = t$, де t – таблиця схеми R , покладемо $E = \{x(R) | t(x)\}$. По-друге, F – стала таблиця $t = \{\langle A, d \rangle\}$. Покладемо $E = \{x(\{A\}) | x(A) = d\}$. □

Крок індукції. Припустимо, що твердження теореми виконується для будь-якого виразу табличної алгебри, який містить менше $k = 1, 2, 3, \dots$ операцій. Нехай вираз F містить k операцій.

Випадок 1 (об'єднання). $F = F_1 \cup_R F_2$. Тоді F_1 і F_2 мають менше k операцій, тому існують вирази числення рядків $\{x(R) | P(x)\}$ і $\{x(R) | Q(x)\}$, еквівалентні F_1 і F_2 відповідно. Покладемо E рівним $\{x(R) | P(x) \vee Q(x)\}$. □

Випадок 2 (різниця). $F = F_1 \setminus_R F_2$. Тоді як і у випадку 1 існують вирази числення рядків $\{x(R) | P(x)\}$ і $\{x(R) | Q(x)\}$, еквівалентні F_1 і F_2 відповідно. Покладемо E рівним $\{x(R) | P(x) \wedge \neg Q(x)\}$. □

Випадок 3 (селекція). $F = \sigma_{\tilde{p}, R}(F_1)$. Нехай $\{x(R) | P(x)\}$ – вираз числення рядків, еквівалентний F_1 . Покладемо E рівним $\{x(R) | P(x) \wedge p(x(A_1), \dots, x(A_m))\}$, де $R = \{A_1, \dots, A_m\}$ схема таблиці, яка є значенням виразу F_1 . Тут припускається, що предикат-параметр селекції заданий так $\tilde{p}(s) = \text{true} \Leftrightarrow p(s(A_1), \dots, s(A_m)) = \text{true}$, $s \in S(R)$, де p – m -арний сигнатурний предикатний символ. □

Випадок 4 (проєкція). $F = \pi_{X, R}(F_1)$. Нехай $\{x(R) | P(x)\}$ – вираз числення рядків, еквівалентний F_1 . Покладемо E рівним $\{y(X \cap R) | \exists x(R)(P(x) \wedge \bigwedge_{A \in X \cap R} y(A) = x(A))\}$. □

Випадок 5 (з'єднання). $F = F_1 \otimes_{R_1, R_2} F_2$. Нехай $\{x(R_1) | P(x)\}$ і $\{y(R_2) | Q(y)\}$ – вирази числення рядків, еквівалентні F_1 та F_2 відповідно. Покладемо E рівним $\{z(R_1 \cup R_2) | \exists x(R_1) \exists y(R_2)(P(x) \wedge Q(y) \wedge \bigwedge_{A \in R_1} z(A) = x(A) \wedge \bigwedge_{A \in R_2} z(A) = y(A))\}$. □

Випадок 6 (перейменування). $F = Rt_{\xi, R}(F_1)$, де $\xi: A \rightarrow A$ – ін'єктивна функція, що здійснює перейменування атрибутів. Тоді існує вираз числення рядків $\{x(R) | P(x)\}$, еквівалентний F_1 . Покладемо E рівним $\{y(R_2) | \exists x(R)(P(x) \wedge \bigwedge_{C \in R \setminus \text{dom} \xi} y(C) = x(C) \wedge \bigwedge_{A \in R \cap \text{dom} \xi} x(A) = y(\xi(A)))\}$, де $R_2 = R \setminus \text{dom} \xi \cup \xi[R]$. □ □

Реляційне числення зі змінними на домені

М. Лакруа (M. Lacroix) та А. Піротте (A. Pirotte) [4] запропонували альтернативу до числення кортежів версію числення – реляційне числення зі змінними на доменах. Відмінність полягає у тому, що замість змінних рядків розглядаються змінні значення із домену, які представляють компоненти рядка. Крім того, числення на домені підтримує умову приналежності (membership condition) [6, 11]. У загальному вигляді умову приналежності можна записати як $\langle t(\langle A_1, d_1 \rangle, \langle A_2, d_2 \rangle, \dots) \rangle R$, де R – схема, A_i – атрибут таблиці t , а d_i – змінна

Нехай P дозволена формула без вільних змінних. Інтерпретація формули P визначається наступним чином.

f1. Якщо $P = \neg G$, то в G немає вільних змінних. Формула P істинна, коли G хибна, і хибна, коли формула G істинна.

f2. Якщо $P = G \wedge Q$ або $P = G \vee Q$, то в G і Q немає вільних змінних. Якщо $P = G \wedge Q$, то формула P істинна тоді, коли G та Q водночас істинні, у всіх інших випадках формула P хибна. Якщо $P = G \vee Q$, то формула P хибна тоді, коли G та Q водночас хибні, у всіх інших випадках формула P істинна.

f3. Якщо $P = \exists x (A) G$, то x – єдина змінна, яка входить у формулу G вільно. Формула P істинна, якщо існує константа $d \in D$, така, що формула $G(d/x)$ – істинна, інакше формула P хибна.

f4. Якщо $P = \forall x (A) G$, то x – єдина змінна, яка входить у формулу G вільно. Формула P істинна, якщо для кожної константи $d \in D$ формула $G(d/x)$ істинна, інакше формула P хибна.

f5. Якщо $P = (G)$, то формула P істинна, якщо формула G істинна і хибна, якщо формула G хибна.

Нехай $E = \{x_1, \dots, x_n \mid P(x_1, \dots, x_n)\}$ – вираз числення на домені. Значенням виразу E назвемо таблицю схеми $R = \{A_1, \dots, A_n\}$, яка містить всі рядки $s \in S(R)$, $s = \{\langle A_1, d_1 \rangle, \dots, \langle A_n, d_n \rangle\}$ такі, що формула $P(d_1/x_1, \dots, d_n/x_n)$ істинна.

Зведення числення рядків до числення на домені

Задамо відображення $\varphi: F \rightarrow E$, яке кожному виразу числення рядків ставить у відповідність еквівалентний вираз числення на домені. Нехай $F = \{y(R) \mid P(y)\}$ – вираз числення рядків, змінна y – єдина змінна, яка входить у формулу P вільно, а $R = \{A_1, \dots, A_n\}$. Використовуємо y як синтаксичну змінну, областю зміни якої є змінні числення рядків.

Застосуємо спочатку відображення φ до термів:

1. Кожну предметну константу d числення рядків залишимо без змін.

2. Кожен терм $x(A_i)$ числення рядків замінимо змінною $x_i \in D$ числення на домені

3. Кожен терм $f(v_1, \dots, v_n)$, де v_i – терми числення рядків, $i = 1, \dots, n$, замінимо на $f(u_1, \dots, u_n)$, де u_j – терми числення на домені, $j = 1, \dots, n$, отримані в результаті попередніх замін.

Застосуємо відображення φ до атомів:

a1. Кожен атом $t(z)$ числення рядків замінимо на $t(z_1, \dots, z_m)$, де $R = \{A_1, \dots, A_m\}$ схема таблиці t .

a2. Кожен атом $p(v_1, \dots, v_m)$, де v_i – терми числення рядків, $i = 1, \dots, m$, замінимо на $p(u_1, \dots, u_m)$, де u_j – терми числення на домені, $j = 1, \dots, m$, отримані в результаті попередніх замін.

Кожну формулу P числення рядків перетворимо на формулу P' числення на домені, де формула P' – це формула P , у якій для кожного атома виконані заміни a1-a2, а кожна вільна змінна z схеми $R = \{A_1, \dots, A_m\}$ числення рядків замінена m змінними $z_1, \dots, z_m$ числення на домені.

• Формулу з квантором $\exists z(R_2)G$, $R_2 = \{A_1, \dots, A_m\}$ перетворимо на $\exists z_1(A_1) \dots \exists z_m(A_m)G'$.

• Формулу з квантором $\forall z(R_2)G$, $R_2 = \{A_1, \dots, A_m\}$ перетворимо на $\forall z_1(A_1) \dots \forall z_m(A_m)G'$.

Виконавши вказані заміни, отримаємо вираз числення зі змінними на домені $E = \{y_1, \dots, y_m \mid P(y_1, \dots, y_m)\}$. В результаті проведених замін, очевидно, що кожна змінна z , числення на домені може приймати точно ті самі значення, що і $z(A_i)$. Отже, значення виразу E співпадає із значенням виразу F . Тому має місце наступна теорема.

Теорема 2. Якщо F – вираз числення рядків, то можна ефективно (і рівномірно) побудувати еквівалентний йому вираз E числення на домені.

Зведення числення на домені до табличної алгебри

Теорема 3. Для кожного виразу числення на домені E можна ефективно (і рівномірно) побудувати еквівалентний йому вираз F табличної алгебри.

Доведення. Нехай $E = \{x_1, \dots, x_n \mid P(x_1, \dots, x_n)\}$ – вираз числення на домені, значенням якого є

таблиця схеми $R = \{A_1, \dots, A_n\}$. Побудуємо для кожної підформули G формули P еквівалентний алгебраїчний вираз F_G . Доведемо індукцією по числу операторів у підформулі G з P , що якщо G містить вільні змінні на універсальному домені $y_1, \dots, y_m$, то для виразу $\{y_1, \dots, y_m \mid G(y_1, \dots, y_m)\}$ існує еквівалентний вираз табличної алгебри F_G . Тоді, якщо G є сама P , отримаємо алгебраїчний вираз для $\{x_1, \dots, x_n \mid P(x_1, \dots, x_n)\}$. Припускається, що $y_1, \dots, y_m$ – всі вільні змінні в G , і значенням виразу $\{y_1, \dots, y_m \mid G(y_1, \dots, y_m)\}$ є таблиця схеми $R_G = \{A_1, \dots, A_m\}$.

Замінімо змінні у формулі P таким чином, щоб жодна змінна не входила одночасно і вільно, і зв'язано у P , та не була зв'язана квантором у двох різних місцях. Тепер кожна змінна асоційована з атрибутом, який або входить у квантор $\forall x(A)$ чи $\exists x(A)$, якщо змінна зв'язана у формулі P , або, якщо змінна вільна в P , співставлений змінній x_i , що знаходиться у виразі E зліва від вертикальної лінії.

Для будь-якого атрибута A можна виписати алгебраїчний вираз. Значенням даного алгебраїчного виразу є одноатрибутна таблиця $\langle t, \{A\} \rangle$, яка містить всі рядки $s = \langle A, d_i \rangle$, $d_i \in D$ – всі елементи множини D . Наприклад, $\langle A, d \rangle \cup \overline{\langle A, d \rangle}$, де d – довільний елемент з D . Позначимо даний алгебраїчний вираз через $[D]$.

База індукції (немає операторів, тобто логічних зв'язок і кванторів). Підформула G – атом вигляду $p(u_1, \dots, u_m)$ або $t(a_1, \dots, a_n)$.

1) Нехай G має вигляд $p(u_1, \dots, u_m)$, де u_i – терми числення на домені, причому $y_1, \dots, y_k$ – всі змінні цих термів. Покладемо F_G рівним $\sigma_{\tilde{p}}([D]_1 \otimes_{R_1, R_2} \dots \otimes_{R_1 \cup \dots \cup R_{k-1}, R_k} [D]_k)$, де R_i – одноатрибутні схеми таблиць, причому ці таблиці є значеннями алгебраїчних виразів $[D]_i$, $i = 1, \dots, k$. Предикат-параметр селекції заданий наступним чином $\tilde{p}(s) = \text{true} \Leftrightarrow p(s(A_1), \dots, s(A_m)) = \text{true}$, $s \in S(R)$, де p – m -арний сигнатурний предикатний символ. $\square$

2) Нехай G має вигляд $t(a_1, \dots, a_n)$, де a_i – константа або змінна на універсальному домені D . Нехай $R = \{C_1, \dots, C_k\}$ – схема таблиці $\langle t, R \rangle$.

Покладемо алгебраїчний вираз F_G рівним $\pi_X(\sigma_{\tilde{p}}(t))$, де $\tilde{p}$ – предикат-параметр селекції, який є кон'юнкцією порівнянь $C_i = a_i$ по всіх i , для яких a_i – константа; X – множина атрибутів $\{C_j \mid a_j \text{ – змінна}\}$. $\square$

Крок індукції. Припустимо, що підформула G містить принаймні один оператор і що гіпотеза індукції справедлива для всіх підформул формули P , що мають менше операторів, ніж G .

Випадок 1. $G = \neg Q$. Нехай F_Q – алгебраїчний вираз для Q , а значенням виразу F_Q є таблиця схеми R_Q . Покладемо $F_G = \sim_{R_Q} F_Q$. $\square$

Випадок 2. $G = Q \vee Q'$. Нехай вільні змінні з Q – це $z_1, \dots, z_k, v_1, \dots, v_p$, а з Q' – це $z_1, \dots, z_k, w_1, \dots, w_q$, де змінні $v_1, \dots, v_p$ та $w_1, \dots, w_q$ попарно не співпадають. Нехай F_Q та $F_{Q'}$ – алгебраїчні вирази для Q і Q' відповідно, причому C_i , $i = 1, \dots, p$ – атрибути, асоційовані зі змінними $v_1, \dots, v_p$, а K_j , $j = 1, \dots, q$ – атрибути асоційовані із змінними $w_1, \dots, w_q$. Покладемо

$$F_1 = F_Q \otimes_{R_Q, \{K_1\}} [D]_1 \otimes_{R_Q \cup \{K_1\}, \{K_2\}} \dots \otimes_{R_Q \cup \{K_1, \dots, K_{q-1}\}, \{K_q\}} [D]_q,$$

$$F_2 = F_{Q'} \otimes_{R_{Q'}, \{C_1\}} [D]_1' \otimes_{R_{Q'} \cup \{C_1\}, \{C_2\}} \dots \otimes_{R_{Q'} \cup \{C_1, \dots, C_{p-1}\}, \{C_p\}} [D]_p',$$

де R_Q та $R_{Q'}$ – схеми таблиць, які є значеннями алгебраїчних виразів F_Q та $F_{Q'}$ відповідно, а C_i та K_j – одноатрибутні схеми таблиць, які є значеннями алгебраїчних виразів $[D]_i$ та $[D]_j'$ відповідно, $i = 1, \dots, p$, $j = 1, \dots, q$. Позначимо через R_{F_1} та R_{F_2} – схеми таблиць, які є значеннями алгебраїчних виразів F_1 та F_2 відповідно. Для відповідним чином побудованих виразів F_1 та F_2 , їхні схеми R_{F_1} та R_{F_2} рівні. Тому покладемо $F_G = F_1 \cup_{R_{F_1}} F_2$. $\square$

Випадок 3. $G = Q \wedge Q'$. Даний вираз числення на домені можна виразити через операції заперечення та диз'юнкції. За законом де Моргана $G = \neg(\neg Q \vee \neg Q')$. $\square$

Випадок 4. $G = \exists x(A)Q$. Нехай F_Q – алгебраїчний вираз для Q . Покладемо F_G

рівним $\pi_{X \setminus A, X}(F_Q)$, де X – схема таблиці, яка є значенням алгебраїчного виразу F_Q . $\square$

Випадок 5. $G = \forall x(A)Q$. Даний вираз числення на домені можна виразити через операцію заперечення та квантор існування $\forall x(A)Q = \neg(\exists x(\neg A))(\neg Q)$. $\square$

Отже, доведено, що при узагальненні таблична алгебра залишається не менш виразною, ніж числення на домені. Враховуючи теорему 1 та теорему 2, можна зробити висновок, що узагальнена таблична алгебра, (узагальнене) числення рядків і (узагальнене) числення на домені однаково виразні. Таким чином, встановлено основний результат

Теорема 4. Узагальнена таблична алгебра, (узагальнене) числення рядків і (узагальнене) числення на домені еквівалентні.

Висновки

1. Проведено узагальнення табличної алгебри на таблиці, яким приписується певна схема. Визначено основні операції над узагальненими таблицями.

2. Класичне числення рядків та числення на домені поповнено довільними предикатними та

функціональними сигнатурами на універсальному домені D (у той час як, зазвичай, розглядають лише бінарні предикати, а функціональна сигнатура взагалі порожня).

3. Визначено синтаксис термів, атомів та формул числення рядків; виділено клас дозволених формул, використовуючи поняття вільних та зв'язаних змінних рядків, введено поняття схеми $scheme(x, P)$ та множини атрибутів $attr(x, P)$, з якими змінний рядок зустрічається у формулах.

4. Доведено, що при такому узагальненні числення рядків не менш виразне, ніж таблична алгебра.

5. Визначено синтаксис термів, атомів і формул числення на домені; виділено клас дозволених формул.

6. Проведено редукцію числення рядків до числення на домені.

7. Доведено, що узагальнена таблична алгебра не менш виразна, ніж числення на домені.

8. Показано, що узагальнена таблична алгебра, узагальнене числення рядків та узагальнене числення на домені однаково виразні.

Список використаних джерел

1. Реляційні бази даних: табличні алгебри та SQL-подібні мови / В.Н. Редько, Ю.Й. Борона, Д.Б. Буй, С.А. Поляков. – Київ: Видавничий дім «Академпериодика», 2001. – 198 с.

2. Мейер Д. Теория реляционных баз данных: [пер. с англ.] / Д. Мейер. – Москва: Мир, 1987. – 608 с.

3. Codd E.F. Relational Completeness of Data Base Sublanguages / Codd E.F. // Data Base Systems. – New York: Prentice-Hall. – 1972. – P.65-93.

4. Lacroix M. Domain-oriented relational languages / M. Lacroix, A. Pirotte // Proc. 3rd Int. Conf on Very Large Data Bases. – Tokyo, October, 1977. – P. 370-378.

5. Ульман Дж. Основы систем баз данных: [пер. с англ.] / Дж. Ульман. – Москва: Финансы и статистика, 1983. – 334 с.

6. Дейт К. Дж. Введение в системы баз данных: [8-е изд.: пер. с англ.] / К. Дж. Дейт. –

Москва: Издательский дом «Вильямс», 2005. – 1328 с.

7. Коннолли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика: [3-изд.: пер. с англ.] / Т. Коннолли, К. Бегг. – Москва: Издательский дом «Вильямс», 2003. – 1440 с.

8. Шенфилд Дж. Математическая логика / Дж. Шенфилд. – Москва: Наука, 1975. – 528 с.

9. Мендельсон Э. Введение в математическую логику: [пер. с англ.] / Э. Мендельсон. – Москва: Наука, 1971. – 320 с.

10. Нікольський Ю.В. Дискретна математика: [3-тє вид.] / Ю.В. Нікольський, В.В. Пасічник, Ю.М. Щербина. – Львів: Магнолія, 2008. – 608 с.

11. Хомоненко А.Д. Базы данных / А.Д. Хомоненко, В.М. Цыганков, М.Г. Мальцев. – Санкт-Петербург: КОРОНА принт, 2004. – 736 с.

Надійшла до редколегії 24.12.2010

УДК 533.95

Ви́жол Ю.О., к.т.н., доц., Жорова А.М.,
аспірант, м.н.с., Муленко І.О., д.ф.-м.н., проф.,
Хомкін О.Л., д.ф.-м.н., професор

**Розробка системи аналітичних
обчислень на ЕОМ та її застосування в
кінетиці плазми**

З використанням стандартного
програмного забезпечення (MathCad) виконаний
аналітичний розрахунок транспортних перерізів
електрон-електронних зіткнень в плазмі для
короткодіючого кулонівського потенціалу
(ККП). На мові програмування Visual Basic
розроблена система аналітичних обчислень на
ЕОМ спеціального призначення, яка
використовується для розрахунку скобкових
інтегралів вищих порядків в інтегралі зіткнень
Больцмана.

Ключові слова: транспортний переріз,
скобковий інтеграл.

*E-mail: AZhorova@mail.ru, Ivan_Mulenko@mail.ru
Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

1. Вступ. Відомим є факт [1,2], що при
розв'язанні широкого класу фізичних задач,
особливо задач статистичної фізики, фізичної
кінетики, квантової хімії, астрофізики та ін.
методами теорії збурень виникає необхідність
виконання великих обсягів одноманітних
громіздких аналітичних обчислень. Формально,
більшість таких завдань може бути розв'язано з
будь-якою наперед заданою точністю. Фактично
ж у переважній більшості випадків врахування
третього і більш вищих порядків теорії збурень
призводить до необхідності виконання такого
значного обсягу аналітичних розрахунків, що
проробити їх «вручну» за невеликий час не
представляється можливим. Крім того,
дослідник, якою б високою кваліфікацією він не
володів, при виконанні таких обчислень не
застрахований від елементарних помилок. Тому
досить давно почали розроблятися методи і

U.O. Vyzhol, A.M.Zhorova*, I.O.Mulenko*,
O.L.Homkin

**Development of analytical calculations
of Eumenes and its application in the
kinetics of plasma**

Using standard software (MathCad) the
analytical calculation of transport cross sections of
electron-electron collisions in plasma for short-
range Coulomb potential (CCP). In the Visual Basic
programming language developed a system of
analytic calculations on computers of special
purpose, which is used to calculate skobkovykh
integrals of higher orders in the Boltzmann
collision integral.

Keywords: transport section, skobkovyy
integral.

системи аналітичного розв'язання таких завдань
за допомогою ЕОМ [1] (методи комп'ютерної
алгебри).

Метою нашої роботи є аналітичний
розрахунок транспортних перерізів розсіювання
заряджених частинок з використанням
стандартного програмного забезпечення
(MathCad), а також розробка спеціального
програмного забезпечення (системи аналітичних
обчислень спеціального призначення) для
розрахунку скобкових інтегралів в інтегралі
зіткнень Больцмана.

Найважливішим завданням кінетики
неідеальної плазми є розрахунок її електронних
кінетичних коефіцієнтів [3]. Наближене
аналітичне розв'язання даного завдання
реалізовано за допомогою методу Чепмена-
Енскога [4-7]. Нелінійне інтегро-диференціальне
рівняння

$$\bar{v}_k \frac{df_k}{dr} + \frac{e_k \bar{E}}{m_k} \frac{df_k}{d\bar{v}_k} = \sum_j \iiint [f_k(\bar{v}'_k) f_j(\bar{v}'_j) - f_k(\bar{v}_k) f_j(\bar{v}_j)] g_{kj} b db d\varphi d\bar{v}_j \quad (1)$$

за допомогою процедури Чепмена-Енскога, що
полягає в розкладі асиметричної по полю
поправки до функції розподілу електронів в ряд
по повній системі поліномів Соніна і подальшої

процедури інтегрування по швидкостям,
зводиться до нескінченної системи лінійних
алгебраїчних рівнянь [4]:

$$3\delta_{n0} \left[\frac{\partial \ln(n_e T)}{\partial \bar{r}} - \frac{e\bar{E}}{T} \right] - \frac{15}{2} \delta_{n1} \frac{\partial \ln T}{\partial \bar{r}} = - \sum_{r=0}^{\infty} L_{rn}^e \bar{P}_r^{(e)}, \quad (2)$$

В (1, 2) введені позначення: b_0 - прицільна відстань, що відповідає розсіюванню на кут 90° . Знак мінус у виразі для ξ відповідає розсіюванню різноіменних зарядів, знак плюс - однойменних. Диференціюючи співвідношення (4) по куту розсіювання, і підставляючи отриманий вираз в (3), отримаємо

$$d\sigma = \pi n_0^2 \xi^2 \frac{tg^2(\chi/2) d\chi}{\cos^2(\chi/2) [1 + \xi^2 tg^2(\chi/2)]^2} \quad (5)$$

Коли $\xi \gg 1$, основний внесок в переріз зіткнень дають зіткнення з розсіюванням на малі кути і (5) переходить в формулу Резерфорда [5].

Згідно з [9] транспортний переріз розсіювання n -ого порядку з передачею імпульсу визначається інтегралом:

$$\sigma_t^n = 2\pi \int_0^\pi b(\chi) \left| \frac{db}{d\chi} \right| (1 - \cos^n \chi) d\chi \quad (6)$$

Порядок транспортного перерізу обумовлений відношенням мас частинок, що зіштовхуються. Передача імпульсу при зіткненні легкої і важкої частинок визначається транспортним перерізом першого порядку, оскільки відношення мас частинок, наприклад електрона і іона (атома) мале. Інтегруючи (6) при $n=1$, для $e-i$ і $e-e$ зіткнень отримуємо:

$$\sigma_{t,e}^1 = 2\pi n_0^2 \left\{ \frac{\xi^2}{(\xi^2 - 1)^2} \ln \xi^2 - \frac{1}{\xi^2 - 1} \right\} \quad (7)$$

Труднощі виникають при розгляді транспортних перерізів частинок однакової маси, що зіштовхуються (електрон-електронних $e-e$). Члени $\sigma_{t,ee}^1$ - транспортні перерізи першого порядку перетворюються в нуль, оскільки в даному випадку передача імпульсу в середньому дорівнює нулю. Ненульовий внесок дають транспортні перерізи парних порядків $n=2,4,\dots$ Їх аналітичне обчислення, з математичної точки зору, не представляє особливих труднощів, але займає багато часу і сил. Значно спрощує це завдання використання математичної лабораторії MathCad. Перед тим як розв'язати задачу, перетворимо вираз (6). Підставляючи (5) в (6) і раціоналізуючи підінтегральний вираз, отримаємо:

$$d\sigma = 2\pi b db = 2\pi b(\chi) \left| \frac{db}{d\chi} \right| d\chi, \quad (3)$$

де b - прицільний параметр, який для ККП має вигляд:

$$b = \frac{r_{A0}}{[1 + \xi^2 tg^2(\chi/2)]^{1/2}}, \quad (4)$$

$\xi = \frac{r_{A0}}{b_0} \pm 1$, $b_0 = e^2 / \mu v^2$; χ - кут розсіювання, μ - приведена маса, v - відносна швидкість частинок,

$$\sigma_{ee}^{2n} = \frac{2\pi r_{sr}^2 \xi^2}{(\xi^2 - 1)} \cdot \left(\frac{\xi^2 - 1}{2\xi^2} - \frac{1}{(\xi^2 - 1)^{2n}} \int_2^{2\xi^2} \frac{((\xi^2 + 1) - z)^{2n}}{z^2} dz \right) \quad (8)$$

Застосовуючи до (8) вбудовані функції вирази для транспортних перерізів парних MathCad: *expand* і *factor* отримуємо наступні порядків. Наприклад 4-го і 6-го:

$$n=3 \quad \sigma_{ee}^6 = \frac{8}{5} \pi r_{sr}^2 \xi^2 \left\{ \frac{15(\xi^2 + 1)^5}{(\xi^2 - 1)^7} \ln \xi - \frac{(23\xi^8 + 48\xi^6 + 98\xi^4 + 48\xi^2 + 23)}{(\xi^2 - 1)^6} \right\}$$

$$n=4 \quad \sigma_{ee}^8 = \frac{32}{105} \pi r_{sr}^2 \xi^2 \left\{ \frac{105(\xi^2 + 1)^7}{(\xi^2 - 1)^9} \ln \xi - \frac{8(22\xi^{12} + 71\xi^{10} + 218\xi^8 + 218\xi^6 + 218\xi^4 + 71\xi^2 + 22)}{(\xi^2 - 1)^8} \right\}$$

3. Аналітичний розрахунок скобкових інтегралів на ЕОМ. Для розрахунку скобкових інтегралів частинок однакової маси, що зіштовхуються, необхідно виконати обчислення виду:

$$\begin{aligned} L_m^{ee} = & -\frac{16}{\pi} n_e \sqrt{\frac{T}{m_e}} \int_0^\infty x^2 \exp(-x^2) dx \int_0^\infty y^3 \exp(-y^2) dy \int_0^\pi \sin \theta d\theta \int_0^\pi b \left| \frac{db}{d\chi} \right| d\chi \int_0^{2\pi} d\varphi \times S_n^{3/2} \left(\frac{x^2}{2} - xy \cos \theta + \frac{y^2}{2} \right) \\ & \times \left\{ S_r^{3/2} \left(\frac{x^2}{2} - xy \cos \eta + \frac{y^2}{2} \right) \left[x^2 - xy(\cos \eta + \cos \theta) + y^2 \cos \chi \right] - S_r^{3/2} \left(\frac{x^2}{2} - xy \cos \theta + \frac{y^2}{2} \right) \times \right. \\ & \times \left[x^2 - xy \cos \theta + y^2 \right] + S_r^{3/2} \left(\frac{x^2}{2} + xy \cos \eta + \frac{y^2}{2} \right) \times \\ & \times \left[x^2 + xy(\cos \eta - \cos \theta) + y^2 \cos \chi \right] - S_r^{3/2} \left(\frac{x^2}{2} + xy \cos \theta + \frac{y^2}{2} \right) \left[x^2 + y^2 \right] \Big\}. \end{aligned} \quad (9)$$

З (9) видно, що початкові, проміжні і кінцеві результати можна записати за допомогою поліномів:

$$A = \sum_i A_i = \sum_i K_i^a \sqrt{\pi}^{\alpha_{i0}^a} x^{\alpha_{i1}^a} y^{\alpha_{i2}^a} \sin^{\alpha_{i3}^a} \chi \cos^{\alpha_{i4}^a} \chi \sin^{\alpha_{i5}^a} \theta \cos^{\alpha_{i6}^a} \theta \cos^{\alpha_{i7}^a} \varphi \left(1 - \cos^{\alpha_{i8}^a} \chi \right), \quad (10)$$

де K_i^a – постійний коефіцієнт, α_{ij}^a – показник ступеня певного співмножника i – члена полінома з ім'ям A , a – степінь даного полінома.

Програма реалізована на мові Visual Basic 6.0. Для зберігання в пам'яті ЕОМ кожного члена необхідно було виділити 8 байт для коефіцієнта K і 9 байт для показників степенів. Крім того, для прискорення пошуку подібних членів для кожного члена обчислювалася і зберігалася у пам'яті контрольна сума (4 байти): $\alpha_i^a = \sum_{j=0}^8 \alpha_{ij}^a d^j$,

де $d \geq \alpha_{ij}^a$ для всіх i, j .

Розв'язання даної задачі з використанням цілочисельного подання в пам'яті ЕОМ для чисельників і знаменників коефіцієнтів у полінома (10) виконано нами в [6, 7]. Проведення розрахунків за допомогою раціональних дробів дозволяє отримати абсолютно точні результати

для елементів до L_{77}^{ee} включно. У такому алгоритмі основна частина машинного часу витрачається на пошук найбільшого загального дільника при скороченні дробів. Обчислення скобкових інтегралів більш високого порядку, ніж L_{77}^{ee} вимагається внаслідок нерівномірного збіжності процесу Чепмена-Енскога для різних типів потенціалів (транспортних перерізів) взаємодії частинок, що зіштовхуються.

Перехід до подання коефіцієнтів у вигляді дійсного числа істотно (на декілька порядків) скорочує час розрахунків, але призводить до накопичення помилки округлення, яка знижує точність коефіцієнта на 4 - 5 порядків. Для оцінки точності обчислень в пам'яті зберігалася абсолютна помилка коефіцієнта члена ΔK (4 байти), яка обчислювалася за загальновідомими формулами:

$$Z = A + B; Z = A - B; (\Delta Z)^2 = (\Delta A)^2 + (\Delta B)^2,$$

$$Z = AB; Z = \frac{A}{B}; \left(\frac{\Delta Z}{Z}\right)^2 = \left(\frac{\Delta A}{A}\right)^2 + \left(\frac{\Delta B}{B}\right)^2.$$

Поліноми зберігалися в динамічних масивах, кожен елемент масиву займав у пам'яті 25 байт.

Реалізація елементарних операцій у програмі. Для виконання необхідних перетворень були визначені такі елементарні операції над поліномами: додавання, множення і інтегрування по змінним φ, θ, x . Для зменшення довжини поліномів і, відповідно, необхідних обсягів пам'яті після кожної елементарної операції слідувала операція зведення подібних доданків.

Нижче наведені використані алгоритми, записані за допомогою псевдокоду, який нагадує популярні мови програмування (C, Паскаль, Алгол). Величина «карнизу» визначає рівень вкладеності оператора. Кожен рядок містить після знака $\triangleright$ необхідні пояснення.

Додавання поліномів ($C = A + B$) виконувалося по алгоритму, який включає операції зведення подібних членів:

$C \leftarrow A$ $\triangleright$ поліном A повністю перекидається в поліном C

for $i \leftarrow 1$ to L_b $\triangleright L_b$ – довжина полінома B
find $\leftarrow$ false $\triangleright$ find – прапор наявності подібного члена в результаті

for $j \leftarrow 1$ to L_c $\triangleright L_c$ – довжина полінома C
if $\alpha^b[i] = \alpha^c[j]$ $\triangleright$ якщо контрольні суми членів рівні

then find $\leftarrow$ true $\triangleright$ то, подібний член в результаті існує

$K^c[j] = K^c[j] + K^b[i]$ $\triangleright$ накопичування коефіцієнту полінома

Обчислення абсолютної похибки $\Delta C[j]$
break $\triangleright$ завершення циклу по поліному C

if not find $\triangleright$ якщо подібний член не знайдено

Додавання нового члену в поліном C
 $L_c \leftarrow L_c + 1$ $\triangleright$ накопичування лічильнику довжини полінома C

$C[L_c] \leftarrow B[i]$ $\triangleright$ перекидання $B[i]$ члена в останній член полінома C

Множення поліномів ($C = AB$) виконувалося по алгоритму, який включає операції зведення подібних членів:

for $i \leftarrow 1$ to L_a $\triangleright L_a$ – довжина полінома A
for $j \leftarrow 1$ to L_b $\triangleright L_b$ – довжина полінома B

$M \leftarrow A[i] B[j]$ $\triangleright M$ – проміжний член

Обчислення контрольної суми α^m .

Обчислення абсолютної похибки ΔM

Пошук подібного члена і додавання M в поліном C

Обчислення коефіцієнтів полінома Соніна виконувалося по формулі:

$$S_n^{3/2}(x) = \sum_{j=0}^n \alpha_{nj} = \sum_{j=0}^n \frac{(-1)^j (2n+3)!!}{2^{n-j} j! (n-j)! (2j+3)!!} x^j,$$

де $m!!!$ – добуток всіх натуральних непарних чисел менших за m .

Обчислення поліномів Соніна ($C = S_n^{3/2}(A)$) від полінома A і формування полінома C виконувалося по наступному алгоритму:

$C \leftarrow \alpha[n, n] E$ $\triangleright E$ – одиничний поліном

for $j \leftarrow n - 1$ downto 0

$B \leftarrow C A$ $\triangleright B$ – проміжний поліном

$C \leftarrow B + \alpha[n, j] E$

Інтегрування по φ виконувалося по формулі:

$$\int_0^{2\pi} \cos^m \varphi d\varphi = \begin{cases} \frac{2\pi(m-1)!!}{m!!}, & \text{якщо } m - \text{іаdіа}, \\ 0, & \text{якщо } m - \text{іаіаdіа}, \end{cases}$$

де $m!!$ – добуток всіх натуральних парних чисел менших, ніж m . Для виконання інтегрування полінома A і формування полінома C використовувався наступний алгоритм:

for $i \leftarrow 1$ to L_a $\triangleright$ для кожного члену полінома A

if $\alpha^m[i, 7] \bmod 2 \neq 0$ $\triangleright$ якщо степінь $\cos \varphi$ – непарна

then continue $\triangleright$ то завершити ітерацію

else $\triangleright$ якщо степінь $\cos \varphi$ – парна

$M \leftarrow A[i]$ $\triangleright M$ – проміжний член

$$K^m \leftarrow \frac{2K^m(\alpha^m[7]-1)!!}{(\alpha^m[7])!}$$

$\alpha^m[i, 7] \leftarrow 0$ $\triangleright$ обнулення степені $\cos \varphi$

$\alpha^m[i, 0] \leftarrow \alpha^m[i, 0] + 2$ $\triangleright$ множення на π

Обчислення контрольної суми α^m

Обчислення абсолютної похибки ΔM

Пошук подібного члену і додавання члену M в поліном C

Інтегрування по x виконувалося для парних степенів (непарні степені x в розрахунках не зустрічалися) по формулі:

$$\int_0^\infty x^m \exp(-x^2) dx = \frac{\sqrt{\pi}(m-1)!!}{2^{\frac{m}{2}-1}}.$$

Інтегрування по θ виконувалося по формулі:

$$\int \cos^m \theta \sin^n \theta d\theta = \begin{cases} 0, & \text{якщо } m - \text{істотне парне}, \\ \frac{(n-1)!!(m-1)!!}{(m+n)!!}, & \text{якщо } n - \text{істотне парне}, \\ \frac{2(n-1)!!(m-1)!!}{(m+n)!!}, & \text{якщо } n - \text{істотне непарне}. \end{cases}$$

Алгоритм інтегрування по x і θ аналогічний алгоритму інтегрування по φ .

Вияток $\sin^m \chi$ із поліномів виконувалося по формулі (непарні степені $\sin \chi$ в розрахунках не зустрічалися):

$\sin^m \chi \cos^n \chi = \sin^{m-2} \chi \cos^n \chi - \sin^{m-2} \chi \cos^{n+2} \chi$,
і для формування полінома C із полінома A використовувався наступний алгоритм:

```
Repeat
  FlagSin ← False ▷ FlagSin – прапор
  наявності членів із співмножником sin χ
  for i ← 1 to La ▷ La – довжина полінома A
    if αn[i,3] > 0 then ▷ якщо степінь
    sin χ більше 0
      FlagSin ← True ▷ член із співмножником
      sin χ в поліномі існує
      M ← A[i] ▷ M – проміжний член
      αm[3] ← αm[3] - 2 ▷ обчислення члена до
      складу якого входить sinm-2 χ cosn χ
      обчислення контрольної суми αm
      пошук подібного і додавання члена M в
      поліном C
      Km ← -Km
      αm[3] ← αm[4] + 2 обчислення члена,
      до складу якого входить -sinm-2 χ cosn+2 χ
      обчислення контрольної суми αm
      пошук подібного і додавання члена M в
      поліном C
  A ← C ▷ поліном C повністю
  перекидається в поліном A
```

Until FlagSin ▷ закінчення процедури, якщо до складу полінома не входять члени із множителем $\sin \chi$

Виділення $1 - \cos^m \chi$ із поліномів виконувалося по формулі:
 $\cos^m \chi = -(1 - \cos^m \chi) + 1$, при цьому використовувався алгоритм, аналогічний алгоритму виключення $\sin^m \chi$.

Видалення нульових членів. У ході розрахунків багато доданків взаємно знищувалися, і результат мав при малих значеннях індексів ($r, n < 3$) члени з нульовими коефіцієнтами. Такі члени із результату видалялися. При великих значеннях індексів r і n до складу результату входили члени з коефіцієнтами порядку 10-15, які не утримували співмножників виду: $1 - \cos^m \chi$. Ці признаки говорили про те, що такі члени з'являються в результаті похибок округлення.

Але мала величина коефіцієнта члена не може служити критерієм видалення члену з результату, оскільки коефіцієнти поліномів Соніна високих порядків можуть мати низьке значення. Тому в ході розрахунків при кожній елементарній операції зі значенням коефіцієнтів членів поліномів проводилася оцінка абсолютної похибки, і член з результату видалявся тільки тоді, коли відносна похибка була ≈ 1 .

Крім того, оцінка похибки показала, що під час розрахунків точність обчислення коефіцієнтів може значно знизитися (до 5 порядків). Тому в даній роботі наведені тільки значущі цифри в результатах.

Алгоритм програми. Алгоритм обчислення виразу можна записати в наступному вигляді:

$$U = S_r^{\frac{3}{2}} \left(\frac{x^2}{2} - xy \cos \eta + \frac{y^2}{2} \right) [x^2 - xy(\cos \eta + \cos \theta) + y^2 \cos \chi] -$$

$$- S_r^{\frac{3}{2}} \left(\frac{x^2}{2} - xy \cos \theta + \frac{y^2}{2} \right) [x^2 - xy \cos \theta + y^2] + S_r^{\frac{3}{2}} \left(\frac{x^2}{2} + xy \cos \eta + \frac{y^2}{2} \right) \times$$

$$\times [x^2 + xy(\cos \eta - \cos \theta) + y^2 \cos \chi] - S_r^{\frac{3}{2}} \left(\frac{x^2}{2} + xy \cos \theta + \frac{y^2}{2} \right) [x^2 + y^2]$$

$$V = -S_n^{\frac{3}{2}} \left(\frac{x^2}{2} - xy \cos \theta + \frac{y^2}{2} \right) x^2 \sin \theta, \quad W = UV,$$

$$F_{rn}(y, \chi) = \int_0^\infty \exp(-x^2) dx \int_0^\pi d\theta \int_0^{2\pi} W d\varphi, \quad L_{rn}^{ee} = \frac{16}{\pi} n_e \sqrt{\frac{T}{m_e}} \int_0^\infty y^3 \exp(-y^2) dy \int_0^\pi b \left| \frac{db}{d\chi} \right| F_{rn}(y, \chi) d\chi, \quad (11)$$

Обчислення полінома U
Обчислення полінома V
Обчислення полінома W
Інтегрування по φ
Інтегрування по θ
Інтегрування по χ
Виняток $\sin^m \chi$
Виділення $1 - \cos^m \chi$
Видалення нульових членів
Інтегрування по θ
Неважко помітити, що вираз (12) можна записати наступним чином:

$$F_{rn}(y, \chi) = \sum_i \sum_j \int_0^\infty \exp(-x^2) dx \int_0^\pi d\theta \int_0^{2\pi} V_i U_j d\varphi, \quad (13)$$

Проведення розрахунків по формулі (13) не потребує формування полінома W і значних затрат на пошук подібних членів в кінцевому поліномі, оскільки поліном $F_{rn}(y, \chi)$ є достатньо коротким. Але при цьому потрібно виконувати множення, інтегрування по трьом змінним і зведення подібних членів в одній підпрограмі, яка замінить блок підпрограм, виділених в алгоритмі (12) пунктиром. Алгоритм цієї підпрограми має вигляд:

for $i \leftarrow 1$ to L_v $\triangleright L_v$ – довжина полінома V

$$L_{11}^{ee} = 16 n_e \left(\frac{e^2}{T} \right)^2 \sqrt{\frac{T}{\pi m_e}} \int_0^\infty y^7 \exp(-y^2) \left[\sum_k \sum_i K_k^i y^{2i} \sigma^{(2k)}(y^2) \right] dy, \quad (14)$$

де $\sigma^{(2k)}(y^2)$ – транспортний переріз електрон-електронного розсіювання. Отримано двохкратний інтеграл (14), який може бути використаний при чисельному інтегруванні по y для розрахунку електропровідності плазми. Після проведення описаних вище дій отримуємо

for $j \leftarrow 1$ to L_u $\triangleright L_u$ – довжина полінома U
 $M \leftarrow A[i] B[j]$ $\triangleright M$ – проміжний член
Інтегрування члена M по φ
Інтегрування члена M по θ
Інтегрування члена M по χ
Обчислення контрольної суми α^m
Обчислення абсолютної похибки ΔM
Пошук подібного члена і додавання M в поліном $F_{rn}(y, \delta)$.

Контроль правильності виконання операцій. Контроль правильності виконання проміжних операцій проводився шляхом порівняння результатів чисельного та аналітичного розрахунків. Результати інтегрування для коефіцієнтів до 4-го порядку збіралися з коефіцієнтами уже розрахованими. Порівняння здійснювалося за допомогою графічного відображення формул у загальноприйнятому вигляді.

Результати розрахунків. За допомогою даної програми вдалося розрахувати інтегральні коефіцієнти до 20 включно. Кінцевий результат аналітичних перетворень був отриманий у вигляді:

значення скобкового інтеграла, визначені за допомогою ЕОМ.

4. Висновки. Таким чином, в даній роботі створено систему аналітичних обчислень на ЕОМ транспортних перерізів і скобкових інтегралів для частинок однакових мас, що

зіштовхуються. Описано реалізацію програми аналітичних перетворень, за допомогою якої вдалося обчислити транспортні перерізи будь-яких порядків і матричні коефіцієнти вищих порядків (до 20 включно). Робота виконана за підтримки МОН України - держбюджетна НДР «Теоретичні дослідження і математичне моделювання термодинамічних і кінетичних процесів у неідеальній плазмі та твердому тілі».

Список використаних джерел

1. Гердт В.П. Аналитические вычисления на ЭВМ в приложении к физике и математике / В.П. Гердт, О.В. Тарасов, Д.В. Ширков // Успехи физ. наук. – 1980. – Т. 130. – Вып. 1. – С. 113-147.
2. Аналитические вычисления на ЭВМ и их применение в теоретической физике // Материалы международного совещания. – Дубна, 1985. – Дубна: ОИЯИ, 1985. – 420 с.
3. Фортов В.Е. Неидеальная плазма / В.Е. Фортов, И.Т. Якубов. – М.: Энергоатомиздат, 1994. – 368с.
4. Чепмен С. Математическая теория неоднородных газов / С. Чепмен, Т. Каулинг. – М.: ИЛ, 1960. – 329с.
5. Климонтович Ю.Л. Кинетическая теория неидеального газа и неидеальной плазмы / Ю. Л. Климонтович. – М.: Наука, 1975. – 289с.
6. Очерки физики и химии низкотемпературной плазмы. Под ред. Л.С.Полака. М.: Наука, 1971. – 436с.
7. Ferziger J.H. Mathematical theory of transport processes in gases / J.H. Ferziger, H.G. Kaper. – N-H PC, Amsterdam, N-H PC, LTD, London, 1972. – 582 p.
8. Ландау Л.Д., Лифшиц Е. М. Механика. – М.: Наука, 1988. – 216 с.

9. Движущаяся плазма./ Сб. переводов под ред. Е.В. Кудрявцева и В.П. Ионова, М.: Изд-во иностр. лит., 1961.

10. Котельников И.А. Лекции по физике плазмы: Учеб. Пособие для студентов физического факультета НГУ / И.А. Котельников, Г.В. Ступаков. – Новосиб. Ун-т. Новосибирск, 1996. – 136с.

11. Кочарян Ашот Эрнстович. Магнитная гидродинамика плазмы сложного химического состава : диссертация... канд. физ.-мат. наук : 01.04.02 Долгопрудный, 2007 140 с. РГБ ОД, 61:07-1/820.

12. Лагарьков А. Н. Метод молекулярной динамики в статистической физике / А.Н. Лагарьков, В.М. Сергеев // УФН – 1980. –Т. 125. – №8. – С. 409.

13. Хокни Р. Численное моделирование методом частиц / Р. Хокни, Дж. Иствуд. – М.: Мир. – 1987. – 640с.

14. Цветков И.В. Применение численных методов для моделирования процессов в плазме: учебное пособие / И.В. Цветков. – М.: МИФИ. – 2007. – 84 с.

15. Выжол Ю.А. Определение вклада электрон-электронных столкновений в проводимость полностью ионизированной плазмы для различных моделей расчёта транспортных сечений / Ю.А. Выжол, И.А. Муленко, Е.В. Федосеева, А.Л. Хомкин // Український фізичний журнал. – 1997. – Т.42. – №9. – С. 1083 – 1090.

16. Выжол Ю.А. Электронные кинетические коэффициенты неидеальной полностью и частично ионизированной плазмы / Ю.А. Выжол, А.Н. Жорова, И.А. Муленко, , А.Л. Хомкин.. // Фізична інженерія поверхні. – 7. – №4 – 2009, с.318-329.

Надійшла до редколегії 01.12.2010

УДК 519.6

Галкін О.В., к.ф.-м.н., Верес** М. М., к.ф.-м.н.

Функціональне програмування. Лямбда числення в мовах FP та F#

У статті розглядається лямбда числення на
прикладі мов F# й FP і використання
функціональних мов у паралельних обчисленнях.

Ключові слова: функціональне
програмування, лямбда числення, F#, FP.

*E-mail: galkin@unicyb.kiev.ua,

**E-mail: veres@unicyb.kiev.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Як відомо, теоретичні основи
імперативного програмування
були закладені ще в 30-х р. ХХ ст. Аланом
Тьюрінгом і Джоном фон Нейманом.

Теорія, покладена в основу функціонального
підходу, також народилася у 20 -

х - 30-х р. того ж століття. Розробниками
математичних основ

функціонального програмування можна назвати
Мозеса Шенфінкеля і Хаскелл

Каррі, що розробили комбінаторну логіку, а
також Алонзо Черча, який описав формальну
систему, названу лямбда-численням. Ця система
була насправді мовою програмування для уявної
машини [1].

Функціональне програмування - це
практична реалізація ідей Алонзо
Черча. Функціональне програмування, перш за
все, має справу з обчисленнями, і використовує
функції для виконання обчислень. Перевагами
функціонального програмування є:

1. Кожен символ в функціональному
програмуванні є константою, отже, ніяка функція
не може породжувати побічні
ефекти. Неможливо міняти об'єкти ні всередині
області видимості, ні зовні. Це означає, що
єдиним ефектом від обчислення функції є
результат, який повертає функція, і єдина річ, яка
впливає на результат - це значення аргументів.

O.V.Galkin, Ph.D, M.M.Veres, Ph.D

Functional programming. Lambda calculus in languages FP and F#

In this paper we consider the lambda calculus in
languages F# and FP and using functional
programming languages for the parallel
calculations.

Key Words: functional programming, lambda
calculus, F#, FP.

2. Функціональна програма вже готова для
виконання в паралельному режимі без
додаткових модифікацій. Немає необхідності
турбуватися про взаємоблокування і боротьбу за
ресурси, тому що просто не буде потрібно
використовувати примітиви синхронізації.

3. Програма, написана на функціональній
мові дозволяє робити гарячу заміну коду, так як у
функціональній програмі весь стан зберігається в
стеку в аргументах, переданих функціям.

4. Функціональні мови піддаються
математичного аналізу. Оскільки функціональна
мова - це просто реалізація формальної системи,
всі математичні операції, які могли б бути
зроблені на папері, також застосовуються до
програм, написаним на такій мові.

На сьогоднішній день існує цілий набір
функціональних мов (наприклад, KRC [2], SASL
[3] та ін), які роблять одне й те ж самим різними
способами. У даній статті будуть розглянуті
основи лямбда-числення, а також функціональні
мови програмування FP і F#, остання була
створена компанією Microsoft для реалізації
паралельних обчислень.

Лямбда-числення

Лямбда-числення - це числення анонімних
(безіменних) функцій. Воно дає, по-перше, метод
представлення функцій і, по-друге, множину

правил виводу для синтаксичного перетворення функцій. Наприклад, для представлена функції $\text{func } x = 2 * x$ у нотації лямбда-числення, необхідно опустити ім'я функції "func", роблячи її анонімною. Тоді така функція буде мати вигляд: $\lambda x . * 2 x$. Лямбда - вирази такої форми називаються лямбда-абстракціями. Символ x після λ називається зв'язаною змінною абстракції і відповідає поняттю формального параметра в традиційній процедури або функції. Вираз праворуч від точки називається тілом абстракції, і подібно коду традиційної процедури або функції, воно описує, що треба зробити за параметром, що надійшов на вхід функції.

До функцій в лямбда - нотації можна застосовувати каррінг, тому всі вони мають лише один аргумент. Строго кажучи, це означає, що застосування функцій повинні міститися в дужках, наприклад:

$(\dots (((\lambda x_1. \lambda x_2. \dots \lambda x_n. E) x_1) x_2) \dots x_n)$

Однак існує угода, за якою функція застосовується завжди до самого лівого аргументу, і тому дужки навколо кожного вкладеного застосування можна опустити без зміни сенсу виразу:

$(\lambda x_1. \lambda x_2. \dots \lambda x_n. E) x_1 x_2 \dots x_n$

Для обчислень лямбда виразів необхідні правила виведення. Найпростіший тип лямбда-вираза - це константа. Константи є самовизначенням, тобто їх неможливо перетворити в більш прості вирази.

δ - правило. Застосування константної функції, однак, може бути перетворено за наявності достатньої кількості аргументів з використанням вбудованих правил, що і становить суть даного правила. Розглянемо приклад. Нехай є константне вираз:

$+ 1 3$

Даний вираз може бути перетворено (тобто піддано δ редукції) з урахуванням вбудованого правила: $+ 1 3 \rightarrow 4$

β - правило. Дане правило описує застосування лямбда абстракції. Розглянемо наступне застосування: $(\lambda x. x x) 2$

Ситуація аналогічна виклику процедури або функції мови високого рівня з фактичним параметром, якій заміщує її формальний параметр. У лямбда-численні така заміна є чисто текстовою і, таким чином, фізично замінюється всі входження пов'язаної змінної x у тілі застосовуваної лямбда-абстракції на вираження аргументу 2, отримуючи в результаті змінену форму тіла. Отже:

$(\lambda x. x x) 2 \rightarrow 2$

Процес називається редукцією в тому сенсі, що в даному випадку спрощується вираз, з нього забирається символ λ і в результаті виходить змінена форма тіла лямбда-абстракції. Редукція абстракції, застосованої до деякого аргументу, може дати в результаті іншу абстракцію, і в цьому випадку процес може бути продовжений.

Редуціруємий вираз називається редексом, і тому можна зробити висновок, що процес редукції лямбда-виразу складається з редукцій, що застосовуються до редексів до тих пір, поки вираз містить хоча б один редекс.

α - правило. Заміна одного імені змінної на інше і становить суть даного правила. Розглянемо приклад: $(\lambda x. (\lambda x. x) (1 x)) 1$

Ясно, що x в тілі внутрішньої абстракції $(\lambda x. x)$ і x у виразі аргументу $(1 x)$ - це різні змінні, що мають однакове ім'я. Якщо застосувати цей вислів до аргументу, скажімо 1, то наступне виконання β -редукції буде помилкою:

$(\lambda x. (\lambda x. x) (1 x)) 1$

$\rightarrow (\lambda x. 1) (+ 1 1)$

$\rightarrow 1$

Таким чином, тут необхідно α - перетворення:

$(\lambda x. (\lambda x. x) (1 x)) 1$

$\rightarrow (\lambda x. (\lambda y. y) (1 x)) 1$

$\rightarrow (\lambda y. y) (+ 1 1)$

$\rightarrow (\lambda y. y) 2$

$\rightarrow 2$

η - правило. Дане правило говорить про те, що два вирази:

$(\lambda x. E x)$ та E еквівалентні за умови, що x не є вільною змінною в E . тому що

$(\lambda x. E x) A \rightarrow E A$ для будь-якого виразу A . Таким чином η - перетворення записується $(\lambda x. E x) \rightarrow E$

Кажуть, що лямбда-вираз перебуває в нормальній формі, якщо до нього не можна застосувати ніяке правило редукції. Іншими словами, лямбда-вираз - в нормальній формі, якщо він не містить редексів. Нормальна форма, таким чином, відповідає поняттю кінця обчислень у традиційному програмуванні.

У контексті функціональних мов і л-числення існують два важливих порядку редукцій. Апплікативний порядок редукцій (АПР), який пропонує спочатку перетворювати самий лівий із самих внутрішніх редексов. Нормальний порядок редукцій (НПР), який пропонує спочатку перетворювати самий лівий із самих зовнішніх редексов.

НПР ефективно відкладає обчислення будь-яких редексов всередині виразу аргументу до тих пір, поки це можливо, з розрахунку на те,

що таке обчислення може виявитися непотрібним. Стратегія вибору самого лівого із самих зовнішніх редексов наказує тому виконувати підстановку для x у $\lambda u. u$ до будь-якого перетворення виразу аргументу. В результаті за один крок виходить нормальна форма $\lambda u. u$. АПР, з іншого боку, обчислює вираз аргументу в першу чергу, що в даному випадку приводить до зациклення. Таким чином, НПР та АПР відповідають ледачому і енергійному обчисленню.

Для представлення рекурсії зручно використовувати погляд на рекурсію, при якому рекурсивну функцію можна представляти як функцію, що має саму себе як аргумент. У цьому випадку функція може виявитися пов'язаною з однією з її власних змінних і буде, таким чином, утримувати в своєму тілі посилання на саму себе. Розглянемо приклад. Нехай є рекурсивна функція в мові Haskell:

```
sum :: Int -> Int
```

```
sum n = if n == 0 then 0 else n + sum (n-1)
```

Це вираз може бути представлено у вигляді λ -абстракції, що має додатковий параметр, який при застосуванні цієї абстракції зв'язується з самою функцією:

```
SUM = \s. \n. COND (= n 0) 0 (n (s (- n 1)))
```

Тепер необхідно пов'язати змінну s зі значенням функції SUM

Це можна зробити, використавши спеціальну функцію, яка має назву Y-комбінатор та задовольняє наступному рівнянню: $Yf = f(Yf)$. Y також відомий як комбінатор фіксованої точки. Фіксована точка функції f - це вираз, який не змінюється при застосуванні до нього функції f (функція може мати декілька фіксованих точок, наприклад, функція $\lambda x. x$, має нескінченну множину таких точок). Вираз Yf дає найменшу фіксовану точку функції f .

У загальному випадку рекурсивна функція f з тілом, що задається виразом E , записується в λ -численні в вигляді $Y(\lambda f. E)$. Визначення Y комбінатора має вигляд:

```
\h. (\x. h (x x)) (\x. h (x x)).
```

Для того, щоб переконатися в коректності даного представлення розглянемо вираз:

```
Yf = (\h. (\x. h (x x)) (\x. h (x x))) f
  => (\x. f (x x)) (\x. f (x x))
  => f ((\x. f (x x)) (\x. f (x x)))
  = f (Yf)
```

Мова FP

Як вже було сказано у вступі однією з найбільш сильних сторін

функціональних мов є те, що вони піддаються формальним перетворенням. Однією з їхніх форм є перетворення програми, що використовує властивість прозорості посилань функціональних програм, щоб «замінити подібне подібним», поліпшивши при цьому характеристики програми під час виконання. Формальні перетворення часто легше висловити, якщо посилання на об'єкти можуть бути видалені з початкової програми, в результаті чого кожна функція редукується до форми, вільної від об'єктів (або вільною від змінних). У мові FP кожна функція виражається саме таким чином.

FP-система складається з трьох компонентів: об'єктів, примітивів і комбінуючих форм.

а) Об'єкти. Хоча об'єкти є частиною кожної FP-системи, не існує механізму, який дозволяв би функції користувача посилатися на об'єкт безпосередньо; об'єкти з'являються тільки на етапі виконання в якості аргументів і повертаються як результат виклику функції. У FP-системі існують три типи об'єктів, а саме $\perp$, атоми і послідовності. Об'єкт $\perp$ є невизначеним об'єктом і являє собою помилковий стан. Атомом в FP є будь-яка константа базового типу. Набір базових типів, підтримуваних FP-системою, довільний, але зазвичай він включає множину цілих чисел, символи, булеві константи T (true) і F (false), дійсні числа, рядки символів. Послідовність в FP - це симетричний список, що не має типу. Він симетричний в тому сенсі, що для кожної операції, що працює з початком (головою) послідовності, існує симетрична операція, яка працює з кінцем (хвостом) послідовності.

б) Примітиви. Примітивами є заздалегідь визначені функції, які підтримуються безпосередньо реалізацією, а не визначаються явно програмістом. Набір примітивів, підтримуваних FP-системою, довільний.

У FP застосування функції f до аргументу x позначається за допомогою символу "двокрапка" $f: x$. Якщо функція має більше одного аргументу, ці аргументи передаються у формі послідовності. Так, якщо f - це функція від декількох аргументів, то застосування f до аргументів $x_1, x_2, \dots, x_k$ записується у вигляді $f: \langle x_1 x_2, \dots, x_k \rangle$

в) Комбінуючі форми. Комбінуюча форма (яка має назву також функціональна форма або операція формування програми) - це просто програмна конструкція, тобто будівельний блок, за допомогою якого конструюються нові функції.

У FP існує шість комбінуючих форм: константа, умовний вираз, композиція, конструкція, «застосувати до всіх» і вставка. Варто зауважити, що функції користувача вводяться в FP за допомогою зарезервованого слова `def`. Наприклад:

`def f = ...`

визначає функцію `f`; тіло `f` виражається за допомогою комбінуючих форм.

1.Константа. Константа в FP - це просто функція, яка при застосуванні до будь-якого визначеного об'єкта завжди дає один і той самий результат. Наприклад,

`k: x = k`, якщо `k` не дорівнює `⊥`
`= ⊥`, якщо `k` дорівнює `⊥`.

2.Умовні вирази. Конструкція умовного виразу в FP має наступний синтаксис:

`p -> q`; `p` тут `p`, `q` і `r` є функціями. Розглянемо приклад,

`(p -> q; r): x = q: x`, якщо `p: x = T (true)`
`= r: x`, якщо `p: x = F (false)`
`= ⊥`, в іншому випадку.

3.Композиція. У FP композиція двох функцій `f` і `g` записується у вигляді `f ° g` що означає `(f ° g): x = f: (g: x)`.

4.Конструкція. За допомогою конструкцій у FP будуються послідовності. Конструкція з `n` функцій записується у вигляді `[f1, f2, ..., fn]`

Результатом застосування такої конструкції до об'єкта `x` є `n`-елементна послідовність

`[f1, f2, ..., fn]: x = <f1:x, f2:x, ..., fn:x>`

На завершення, наведемо приклад програми на FP: обчислення `n!`

`def fac = eq0 -> 1; * ° [id, fac ° (- ° [id, 1])]`,

тут `id` - тотожне перетворення, `*`-множення.

Варто відзначити, що на сьогоднішній день мова FP активно доповнюється.

Мова F #.

F # - це функціональна мова програмування загального призначення. Розроблено Доном Сайма в Microsoft Research в Кембриджі. Розповсюджується разом з .NET Framework і Visual Studio як частина Visual Studio 2010.

Синтаксис F # побудований на математичній нотації, а програмування чимось схоже на алгебру, що робить F # схожим на Haskell. Наприклад, коли визначається новий тип, то можете вказати, що змінними цього типу будуть «цілі або рядок». Це виглядає наступним чином:

`type myType = IntVal of int | StringVal of string`

Мова F # також підтримує конструкції функціонального програмування, наприклад обробку функцій як значень, використання у виразах неіменованим функцій, об'єднання функцій для утворення нових функцій, каррірованіє функції і неявне визначення функцій за допомогою часткового застосування аргументів функції, лямбда-вирази.

Функції визначаються за допомогою ключового слова `let` або, якщо функція рекурсивна, комбінації ключових слів `let rec`. Наступна конструкція `let f x = x + 1` еквівалентна `f (x) = x + 1` в алгебрі. У цьому прикладі ім'я функції = `f`, аргумент = `x` і належить до типу `int`, тіло функції = `x + 1`, значення, що повертається має тип `int`.

У мові F # всі функції вважаються значеннями (функціональні значення). Приклад функції, яка приймає як аргумент функціональне значення:

`let apply1 (transform: int -> int) y = transform y`

Тип функціонального значення можна задати за допомогою токена `->`. У лівій частині токена знаходиться тип аргументу, а в правій частині - значення, що повертається. У прикладі `apply1` є функцією, що приймає функцію `transform` як аргумент, де `transform` є функцією, яка приймає ціле число і повертає інше ціле число. Наступний код показує, як використовувати функцію `apply1`.

`let increment x = x + 1`

`let result1 = apply1 increment 100 // result1 дорівнює 101`

Кілька аргументів розділяються токеном `->`, як показано в наступному прикладі.

`let apply2 (f: int -> int -> int) x y = f x y`

`let mul x y = x * y`

`let result2 = apply2 mul 10 20 // result2 дорівнює 200.`

У попередніх прикладах замість визначення іменованих функцій `increment` і `mul` можна було б використовувати лямбда-вирази, як показано нижче.

`let result3 = apply1 (fun x -> x + 1) 100 // result3 дорівнює 101`

`let result4 = apply2 (fun xy -> x * y) 10 20 // result4 дорівнює 200`

Лямбда-вирази визначаються за допомогою ключового слова `fun`. Лямбда-вираз нагадує визначення функції, за винятком того, що замість токена `=` для відділення списку аргументів від тіла функції використовується токен `->`. Як і в звичайному визначенні функції,

типи аргументів можуть виводитися або бути задані явно.

Конвеєризація дозволяє об'єднати виклики функцій на ланцюжок послідовних операцій. Конвеєризація працює таким чином.

```
let result6 = 100 |> function1 |> function2 // result6 дорівнює 202
```

Крім того, бібліотека F# підтримує асинхронні робочі процеси для забезпечення паралельних обчислень і механізми для зв'язку між паралельними робочими процесами. Особливості мови F# забезпечують можливість виконання асинхронних обчислень, тобто без блокування інших операцій.

Функція `Parallel` створює асинхронне обчислення, яке виконує всі задані асинхронні обчислення, спочатку ставлячи кожне з них в чергу в якості робочого елементу і використовуючи шаблон розгалужень і з'єднань. Приклад паралельного обчислення сум.

```
let task1 = async (return 10 + 10)
let task2 = async (return 20 + 20)
Async.Run (Async.Parallel [task1; task2])
```

Існує декілька способів асинхронного програмування, і клас `Async` надає методи, що підтримують кілька сценаріїв. Загальний підхід полягає в створенні об'єктів `Async`, що представляють обчислення, які потрібно виконувати асинхронно, з наступним запуском цих обчислень за допомогою однієї з запускаючих функцій. Різні запускаючі функції забезпечують різні способи виконання асинхронних обчислень, і вибір залежить від

того, чи потрібно використовувати потік, фоновий потік або об'єкт завдання платформи .NET Framework, а також від того, чи є наступні функції, які повинні виконуватись після завершення поточного обчислення.

Висновки.

Перед сучасним програмуванням виникла низка складних проблем, насамперед це:

1. Складність навколишнього світу впливає на складність програмних систем.
 2. Паралелізм - потреба писати паралельний код, якій різко збільшує складність.
 3. Складна система - більша кількість помилок, різко зростає вартість якісного коду.
- Більш того, актуальність цих проблем швидко зростає з часом.

Функціональне програмування дозволяє ефективно розв'язувати ці проблеми. По-перше, як було зауважено у вступі функціональне програмування має дуже чітку математичну основу. По-друге, відсутність у програмі написаної на функціональній мові оператора присвоєння та циклів робить таку програму готовою для виконання в паралельному режимі без додаткових модифікацій. По-третє, навіть досить складний код, який написаний на функціональній мові виглядає досить просто і відлагодження програм досить легке, в силу відсутності побічних ефектів при обчисленні функцій, так як єдина річ, яка впливає на результат - значення аргументів.

Список використаних джерел

1. C. Hankin An Introduction to Lambda Calculi for Computer Scientists. College Publications, 2004
2. Turner, D. A., 'Recursion equations as a programming language'. In J. Darlington, P. Henderson & D. Turner (eds.), Functional programming and its applications ,1-28, Cambridge University Press, 1982
3. Turner, D.A., The SASL language manual. University of St Andrews, 1976
4. Chris Smith. Programming F#. O'Reilly Media, 2010

5. Функціональне програмування [Електронний ресурс] / Д.В. Сошников // Відеокурс — 2010. — Режим доступу до матеріалів: <http://www.intuit.ru/department/pl/funcprog/>.
6. Бібліотека MSDN: Visual F# [Електронний ресурс] — 2010. — Режим доступу до матеріалів: <http://msdn.microsoft.com/ru-%20ru/library/dd233154.aspx>.

Надійшла до редколегії 14.12.2010

УДК 519.7

Грунський І.С., к.ф.-м.н., с.н.с., Козловський*
В.А., к.ф.-м.н., с.н.с.

Дескрипція автоматів їхньою поведінкою

Розглянуто деякі питання дескрипції автоматів, у першу чергу, фрагментами поведінки. Наведено умови існування контрольних експериментів та ідентифікаторів станів автомата. Для систем визначальних співвідношень, якими може бути заданий автомат, вказано метричні характеристики, вказано зв'язок між визначальними співвідношеннями і контрольними експериментами. Для експериментів з груповими автоматами наведено точні оцінки складності таких експериментів

Ключові слова: скінченний автомат експеримент, визначальне співвідношення.

*E-mail: kozlovskii@iamm.ac.donetsk.ua, kozlovskii@list.ru

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Модель скінченного автомата є однією з базових для інформатики і, як і раніше, зберігає своє значення при розгляді багатьох задач, які виникають при проектуванні й експлуатації складних дискретних систем. У роботі обговорюються задачі аналізу й синтезу автоматів за їхньою поведінкою як прямі й зворотні задачі теорії автоматів. Класичними задачами теорії автоматів є задачі аналізу процесів перетворення інформації, здійснюваних автоматами, і властивостей автоматів (прямі задачі), і задачі синтезу автоматів із заданими властивостями і ідентифікації (відновлення, розпізнавання, верифікації, контролю й діагностики) автомата шляхом експерименту з ним (зворотні задачі). Як у багатьох інших теоріях, зворотні задачі звичайно складніше прямих, що було відзначено ще в роботі Э.Мура з теорії експериментів з автоматами. У ній вперше були розглянуті задачі відновлення автомата за вхід-вихідними послідовностями як результатами експериментів. Задачі, аналогічні задачам відновлення автоматів, виникають й у прикладних областях, таких як діагностика обчислювальної апаратури, а в останні роки в

I.S. Grunsky, dr., V.A. Kozlovskii, dr.

Automata characterisation through behaviour

Certain problems of automata description are considered, first and foremost with behaviour fragments. Existence conditions for checking experiments and automaton's state identifiers are given. Metric characterisation of the systems of automaton's defining relations is given, a link between checking experiments and defining relations is established. The exact complexity estimations for experiments with group automata are given.

Key Words: finite automaton, experiment, defining relation

області формальних методів синтезу й верифікації програмно-апаратних систем [3]. Задача синтезу полягає в побудові автомата за заданою його специфікацією - завданням на необхідну, можливу і заборонену поведінку, а задача верифікації - у визначенні властивостей автомата, аж до його повної побудови, проводячи експерименти із заданим "чорним ящиком" - реалізацією цього автомата. Априорі або в процесі експерименту задаються або виникають фрагменти дозволеної і/або (ко)фрагменти забороненої поведінки автомата. Фрагменти й кофрагменти утворюють єдину основу для рішення задач синтезу й ідентифікації. Зазначені задачі, у першу чергу, і розглядаються в роботі. При цьому фрагмент - кофрагмент задає автомат із заданим ступенем точності. Найчастіше як фрагмент розглядається спостережена поведінка - множина вхід-вихідних слів, яка називається контрольним експериментом й однозначно визначає автомат при накладенні певних обмежень на клас розглянутих автоматів. Зауважимо, що в загальному випадку така множина слів може бути нескінченною і визначатися так названими представленнями автоматів. У роботі наведено короткий огляд результатів, що відносяться до зазначеної поведінкової теорії з отримані з використанням

топологічних, теоретико-графових і алгебраїчних методів. Вони пов'язані з можливістю опису автомата фрагментами його поведінки і пов'язаними з цим питаннями відповідних об'єктів – експериментів і представлень, їхньою побудовою, характеристикацією і складністю (частина результатів наведена в [1, 2, 4, 5, 6, 10]).

Контрольні експерименти і їхнє існування

Розглядаються автомати Милі $A = (S, X, Y, \delta, \lambda)$, де S, X, Y – множини станів, входів і виходів відповідно, а δ, λ – функції переходів і виходів [1].

Під експериментом з автоматом розуміємо процес подачі на автомат послідовності входних сигналів, спостереження відповідної поведінки автомата й вивід висновків про функціонування й властивості автомата, заснованих на цих спостереженнях й апріорній інформації про автомат. Основна задача теорії експериментів полягає в розробці ефективних експериментів, що дозволяють одержати певну інформацію про будову автомата, його функції. При цьому виникає велике коло задач, пов'язаних із класифікацією експериментів, з питаннями можливості розв'язання задач розпізнавання тих або інших властивостей автомата певними видами експериментами, з оцінками складності мінімальних експериментів, достатніх для рішення тих або інших задач розпізнавання, а також з оцінками складності побудови цих експериментів.

Нехай задані автомат-еталон A і клас автоматів F . Контрольний експеримент (КЕ) автомата A відносно класу F – така множина W вход-вихідних слів, породжуваних автоматом A у деякому його стані, яка не породжується ніяким іншим (не еквівалентним еталону) автоматом із класу F (можливо, нескінченного). В [4] отримані умови існування контрольних експериментів. На множині ініціальних автоматів вводиться бэровська метрика β : $\beta(A, B) = 0$, якщо $A = B$, і $\beta(A, B) = \frac{1}{k}$, якщо $L_A^k \neq L_B^k$ й $L_A^{k-1} = L_B^{k-1}$. Тут L_A^k – множина вход-вихідних слів довжини k , породжуваних початковим станом автомата A . Для бэровської метрики β й довільних класів F приведених ініціальних автоматів справедливий наступний критерій існування контрольного експерименту.

Теорема 1[4]. Рівносильні твердження:

1. Існує контрольний експеримент автомата A відносно F ;
2. Множина L_A^k є контрольним експериментом відносно A й F для деякого k ;
3. $O_{\frac{1}{k}}(A) \cap F \subseteq \{A\}$ для деякого k ;
4. $O_{\frac{1}{k}}(A) \cap F$ – скінченна множина для деякого k ;
5. $A \notin \lim F$.

Тут $\lim F$ – множина граничних для класу F автоматів у заданій метриці. Цей критерій показує, що у бэровській метриці β процес виведення висновків у процесі експериментування зводиться до перевірки умови 3. Якщо клас F скінченний, то існує верхня оцінка t числа станів автоматів з $F \cup \{A\}$ і тому L_A^{2t} є контрольним експериментом. Для довільного нескінченного класу F критерій не конструктивний, однак існують нескінченні класи, для яких цей критерій конструктивний.

Фрагменти поведінки і представлення автоматів

Розширення поняття експерименту приводить до поняття фрагмента й представлення автомата, як більш загальних видів дескрипторів автоматів. Дескриптор автомата [4] у загальному випадку визначаємо як деякий об'єкт (елемент множини об'єктів, які і називаємо дескрипторами), пов'язаний з автоматом та визначаючий його властивості. Прикладом таких дескрипторів є експерименти автоматів. В залежності від класу автоматів такі дескриптори можуть задавати автомати з різним ступенем точності. Якщо такий дескриптор однозначно визначає автомат у заданому класі, то такий дескриптор названо представленням автомата відносно заданого класу. При дослідженні точності опису автомата фрагментами важливу роль відіграють так звані ідентифікатори властивостей автомата (станів, входів, виходів) [4, 5] – фрагменти, які дозволяють однозначно ідентифікувати ці властивості. Ідентифікатори дають ще один приклад дескриптора. Загальне визначення фрагмента враховує невизначеність, що може виникати при спостереженні в експерименті або завданні на синтез автомата. Розглянемо більше вузьке поняття фрагмента – безпосередні правильні фрагменти, і деякі ключові результати, що розширюються й на загальне поняття.

Частковий автомат R називається (безпосереднім) фрагментом автомата A , якщо існує гомоморфізм R в A . Гомоморфізм називається повним, якщо для кожної дуги автомата A існує її прообраз по цьому гомоморфізмові в автоматі R . Фрагмент R визначає деякий клас автоматів $K(R)$ як множину всіх тих автоматів, фрагментом яких є R . Ядром скінченного фрагмента R назвемо такий його підавтомат Q , для якого існує повний гомоморфізм φ із R в Q , причому $\varphi(R) = Q$, а всякий ендоморфізм фрагмента Q в себе є повним автоморфізмом.

Теорема 2[2]. Для кожного скінченного фрагмента існує єдине з точністю до ізоморфізму ядро.

Ця теорема показує наявність у кожному фрагменті R деякого ненадлишкового підфрагмента Q , що несе ту ж інформацію, що й весь фрагмент, тобто фрагмент і його ядро описують один і той же клас автоматів $K(R)$.

Нехай R – деякий фрагмент еталона A , і t – деякий довільний зафіксований стан фрагмента. Фрагмент із зафіксованим станом позначимо R_t . Фрагмент R_t назвемо ідентифікатором стану s еталона, якщо для будь-якого гомоморфізму φ фрагмента R в еталон A виконується рівність $\varphi(t) = s$.

Розглянемо окремих вид ідентифікаторів станів еталона, пов'язаний з експериментами. З кожним станом s еталона асоціюються дві множини вход-вихідних слів: λ_s – множина всіх вход-вихідних слів, що починаються в s , тобто породжених станом s , і двоїста множина ψ_s усіх вход-вихідних слів, що закінчуються в s , тобто таких $(p, q) \in (X \times Y)^*$, для яких знайдеться таке $\tilde{s} \in S$, що $\delta_A(\tilde{s}, p) = s$ й $\lambda_A(\tilde{s}, p) = q$. Пару (V_1, V_2) назвемо околom стану s еталона, якщо $V_1 \subseteq \psi_{\lambda_s}$, $V_2 \subseteq \lambda_{\lambda_s}$. З кожним околom (V_1, V_2) природно пов'язується фрагмент $R_s(V_1, V_2)$. Тому, використовуючи вільність мови, окіл (V_1, V_2) стану s будемо називати ідентифікатором стану s еталона, якщо $R_s(V_1, V_2)$ є таким ідентифікатором. Ідентифікатор (V_1, V_2) будемо називати початковим (кінцевим), якщо $V_1 = \emptyset$ ($V_2 = \emptyset$). Початковий (кінцевий) ідентифікатор еталона називається простим, якщо кратність $V_2(V_1)$ дорівнює 1. Прикладами

таких початкових ідентифікаторів є діагностичні експерименти, а кінцевих – установчі експерименти.

Наступна теорема дає критерій існування ідентифікаторів станів, що є їхніми околами.

Теорема 3[2]. Рівносильні твердження:

1. Існує окіл стану s , що є його ідентифікатором.
2. (ψ_s, λ_s) є ідентифікатором стану s .
3. Існує ідентифікатор (V_1, V_2) стану s , для якого $|V_1| + |V_2| \leq n - 1$ і висота $h(V_2) \leq 2^{2^n}$.

Характеризація представлень автоматів

Далі вважаємо, що клас $F \subseteq F_n$ – так званий n -щільний відносно A . Дещо спрощуючи, можна вважати клас F n -щільним, якщо йому належить всякий автомат, отриманий з A зміною хоча б одного значення його функції виходів або переходів. Нехай $A \in F$ і R – фрагмент A [2]. Фрагмент R називається представленням для (A, F) , якщо з існування гомоморфізму φ з R у $B \in F$ витікає еквівалентність A і B .

Відношення σ на множині станів фрагмента R називається верифікованим відношенням сумісності, якщо для будь-якої пари $(a, b) \in \sigma$ і гомоморфізму φ фрагмента R в $B \in F$ справедливою є рівність $\varphi(a) = \varphi(b)$. Це відношення породжує деяку мінімальну за включенням конгруенцію σ^* на множині станів і визначає, таким чином, фактор-автомат $[R]_{\sigma^*}$, який назвемо згортокою.

Відношення ρ на множині станів фрагмента R називаємо верифікованим відношенням несумісності, якщо для будь-якої пари $(a, b) \in \rho$ й гомоморфізму φ фрагмента R в $B \in F$ справедливою є нерівність $\varphi(a) \neq \varphi(b)$.

Пари $P = (\sigma, \rho)$ верифікованих відношень сумісності σ й несумісності ρ будемо називати верифікованою парою для автомата R (відносно класу F). З R і P зв'яжемо звичайний неорієнтований граф $G(R, P) = (V_R, U_R)$, визначений у такий спосіб: множина його вершин V_R є множина S/ε класів верифікованої конгруенції $\varepsilon = \sigma^*$; якщо $v = [s]$ й $w = [t]$ – класи цієї конгруенції, представниками якої є стани s й t відповідно, і $(s, t) \in \rho^*$, то $(v, w) \in U_R$.

На станах згортки $[R]_{\sigma^*}$ відношення ρ індукує верифіковане відношення несумісності ρ' : якщо

$[s]$ й $[t]$ є стани автомата $[R]_\sigma$, то $([s], [t]) \in \rho'$ тоді й тільки тоді, коли $(s, t) \in \rho_c^*$. Граф $G(R, P)$ будемо називати асоційованим з автоматом R за допомогою пари P , або більш коротко, асоційованим з парою (R, P) . Показується, що довільний звичайний неорграф є асоційованим при відповідному виборі R та F [2]. Наведемо приклад твердження, що показує можливість зводити вивчення представлень для доволі широких класів автоматів до вивчення задачі розфарбування асоційованих графів.

Нехай A визначено-діагностований автомат порядку 1 (ОД-1 автомат) з n станами, тобто такий, у якого кожне слово є діагностичним, F – n -щільний відносно A клас, а $\Phi^k(A)$ позначає множину всіх вхід-вихідних слів довжини k , породжуваних автоматом A .

Теорема 4[2]. R – представлення для (A, F) тоді й тільки тоді, коли існує така верифікована пара $P = (\sigma, \rho)$ для (R, F) , що виконуються умови:

1. Існує гомоморфізм φ автомата R в еталон A .
2. $\Phi^1(R) = \Phi^1(A)$.
3. Граф $G(R, P)$ однозначно n - розфарбований.

Зауважимо, що теорема 4 може бути узагальнена на випадок, коли замість єдиного еталона A розглядається множина автоматів $\tilde{A}$, тобто, коли фрагмент є представленням автомата, що визначається з точністю до деякої множини автоматів $\tilde{A}$. В цьому випадку розширюються поняття: представлення R множини автоматів $\tilde{A}$ відносно класу F (вимога мати гомоморфізм R у кожний автомат множини $\tilde{A}$); щільності множини $\tilde{A}$ (F – n -щільний відносно $\tilde{A}$, якщо він такий відносно кожного автомата з множини $\tilde{A}$). Тоді справедливою є узагальнена теорема

Теорема 4. R – представлення для $(\tilde{A}, F)$ тоді й тільки тоді, коли існує така верифікована пара $P = (\sigma, \rho)$ для (R, F) , що виконуються умови:

1. Існує гомоморфізм φ автомата R в кожний автомат $A \in \tilde{A}$.
2. $\Phi^1(R) = \Phi^1(A)$ для кожного $A \in \tilde{A}$.
3. Граф $G(R, P)$ має рівно $|\tilde{A}|$ неізоморфних розфарбувань.

Наведені достатні й необхідні умови для розпізнавання представлень постулюють тільки існування верифікованої пари. Для ОД-1 автоматів і його підкласів: групових, автоматів без втрати інформації (БВІ- автоматів), локально породжених автоматів, так званого базису локальної породжуваності, дається конструктивний спосіб побудови таких

верифікованих пар, і, відповідно, асоційованих графів. Одним з основних засобів побудови верифікованих пар є ідентифікатори станів. Їх легко верифікувати для ОД-1 автомата й класу F_n всіх автоматів з n станами і побудувати відповідний асоційований граф. У цьому випадку по фрагменту будується згортка послідовністю двох кроків, що чергуються: 1) ототожненням станів, з яких виходять дуги, позначені однаковими вхід-вихідними мітками (початкові ідентифікатори станів), і 2) детермінізацією отриманого на першому кроці недетермінованого у загальному випадку автомата. Кожному стану отриманої згортки відповідає множина вхідних символів, за якими визначений перехід з даного стану. Показується, що граф перетинань сукупності таких множин і є асоційований граф. Побудова таких графів, у свою чергу, дозволяє оцінити складність розпізнавання представлень (визначення того, чи буде заданий фрагмент представленням для обраних еталона й класу) у термінах теорії NP -повноти.

Розмічені експерименти й системи визначальних співвідношень для автоматів

У даному розділі розглянуто алгебраїчний підхід до задач характеристики представлень й оцінок їхніх параметрів. Системи визначальних співвідношень (СВС) можна вважати ще одним типом дескрипторів, які вперше для автоматів були розглянуті Ю. И. Соркіным. В [5] його ідеї узагальнено і розвинуто, а в [6] показаний й вивчений зв'язок СВС із контрольними експериментами й представленнями автоматів. Так, в [5] знайдена мінімальна канонічна система визначальних співвідношень κ_A , і розглянута її структура; запропонована процедура зведення (редукції) будь-якої скінченної системи визначальних співвідношень до канонічної; знайдений критерій, при якому скінченне бінарне відношення є системою визначальних співвідношень для заданого автомата A без побудови автомата; запропоновано рішення проблеми ізоморфізму для ініціальних кінцевих автоматів без явної побудови баз цих автоматів; показаний зв'язок класу систем визначальних співвідношень для даного автомата з іншими його властивостями (обходами) і запропоновані процедури переходу від обходу до системи визначальних співвідношень і навпаки; поняття й апарат систем визначальних співвідношень поширюються на часткові автомати.

Дослідження часткових автоматів відбувається за допомогою визначальної системи.

Наведемо один з основних результатів, що вказує оцінки параметрів мінімальних СВС. Нехай ρ - деяке бінарне відношення на X^* . Це відношення назвемо системою визначальних співвідношень (СВС) для A , якщо $[\rho] = \rho_A$, де $[\rho]$ - правоінгугруентне замикання ρ , тобто найменша права конгугруенція, що містить ρ .

Введено так звану канонічну систему визначальних співвідношень. Ця канонічна система складається з пар слів, що входять у найкоротший за певним лексикографічним порядком базис досяжності автомата, і доповнених символами вхідного алфавіту. Запропоновано процедуру редукції скінченного бінарного відношення ρ , що приводить до найбільш економного у певному змісті відношення. За допомогою процедури редукції знайдено рішення задачі характеристики бінарного відношення, а саме задачі визначення, чи є це скінченне бінарне відношення системою визначальних співвідношень для фіксованого автомата A . Рішенням задачі характеристики для скінченного ρ служить наступна

Теорема 5[5]. Бінарне відношення ρ є системою визначальних співвідношень для фіксованого автомата A тоді й тільки тоді, коли його редукція збігається з канонічною системою визначальних співвідношень для A .

На часткові автомати поширені поняття й апарат систем визначальних співвідношень. Частковий автомат характеризується парою об'єктів - бінарним відношенням ρ на вхідних словах і підмножиною множини вхідних слів M : ρ є аналогом системи визначальних співвідношень для всюди визначеного автомата, а M однозначно визначає область визначення функції переходів часткового автомата. Показано, що така визначальна система у вигляді пари (ρ, M) задає автомат A з точністю до ізоморфізму.

Нехай κ_A - довільна канонічна СВС для автомата $A = (A, X, \delta_A, a_0)$, у якого $|X| = m$ й $|A| = n$. $N(\kappa_A)$ - число канонічних систем. Справедливі наступні оцінки:

Теорема 6[5].

1) $1 \leq N(\kappa_A) \leq m!$, причому ці оцінки досяжні для всіх m ;

2) $|\kappa_A| = (m-1)n+1$;

$$3) S(\kappa_A) \geq \left(\frac{m^{l+1}-1}{m-1} - n \right) l + \left(m' + n - \frac{m^{l+1}-1}{m-1} \right) m(l+1),$$

де $l = \lfloor \log_m n(m-1) + 1 \rfloor - 1$;

$S(\kappa_A) \leq n(m-1)(n-1) + m(2n-1)$, причому ці оцінки досяжні для всіх $n, m > 1$.

Якщо вивчення СВС для автоматів мало загальноалгебраїчну мотивацію, і основні акценти дослідження також мали алгебраїчні витoki, зв'язок СВС з експериментами з автоматами має природне автоматне підґрунття, пов'язане, значною мірою, з цілою низкою алгоритмів побудови контрольних експериментів, які базуються на визначенні і застосуванні в таких експериментах спеціальних підслів - ідентифікаторів станів автомата у вищевказаному визначенні. Цей підхід, спочатку запропонований Ф.Хенні[7] для простих контрольних експериментів, потім був узагальнений і розвинений М.П. Василевським[8] і на випадок кратних експериментів (звернемо також увагу на огляд [9]) і був застосований також і для більш широких автоматних моделей у алгоритмах їх тестування. В [6] вивчено зв'язок контрольних експериментів із СВС. При цьому розглядалися контрольні експерименти з груповими автоматами відносно нескінченних класів автоматів, на відміну від випадку скінченних класів, який звичайно розглядається. Контрольні експерименти з автоматами щодо нескінченних класів існують тільки при додаткових припущеннях про можливості спостереження в експерименті деякої інформації про внутрішні стани досліджуваного автомата. Такі припущення формалізовано у понятті розміченого експеримента, яке розширює поняття звичайного експеримента введенням додаткових засобів спостереження. Такі засоби задаються спеціальними мітками, які можна спостерігати в експерименті, і які сигналізують про перебування автомата у деяких його станах. Зокрема, такими засобами можуть виступати ідентифікатори станів автомата. Такі експерименти були детально вивчені для випадку групових автоматів. Запропоновано дескриптор-представлення ініціального автомата без виходу у вигляді так званої визначальної пари $P = (R, L)$ двох множин вхідних слів, перша з яких складається із слів, циклічних у кожному стані, а друга - із слів, циклічних відносно начального

стану автомата. Доведено, що такий дескриптор пов'язаний з деякою групою та її підгрупою, які разом визначають груповий автомат. Знайдені характеристики контрольних експериментів групового автомата відносно нескінченного класу звідних групових автоматів, що зв'язують експерименти й визначальні пари; отримані точні довжини таких експериментів при мінімальних допущеннях про засоби спостереження. Збільшення потужності множини спостережуваних у розміченому експерименті міток розширює можливості експериментатора й може приводити до більш ефективних за довжиною експериментів.

Зафіксуємо деяку множину M міток потужності k . Нехай $w = (p, q) \in \lambda_i$, де i – стан деякого автомата B . Нехай також $p = p_0 p_1 p_2 \dots p_{k+1}$, $q = q_0 q_1 q_2 \dots q_{k+1}$, де $\delta(i, p_0) = s_1$, $\delta(s_1, p_1) = s_2$, ..., $\delta(s_k, p_k) = s_{k+1}$, $\lambda(i, p_0) = q_0$, $\lambda(s_1, p_1) = q_1$, ..., $\lambda(s_k, p_k) = q_{k+1}$, і p_i непусте слово при $i = 1, \dots, k$. Функцію φ , що співставляє кожному стану автомата деяку мітку або порожній символ ε , назовемо функцією розмітки автомата B . Стан $s \in S_B$, для якого $\varphi(s) \neq \varepsilon$, назовемо відміченим. Тоді розміченим словом (розміченим експериментом) називається слово $w_\varphi = (p_0, q_0) \varphi(s_1) (p_1, q_1) \varphi(s_2) \dots \varphi(s_k) (p_k, q_k) \varphi(s_{k+1}) (p_{k+1}, q_{k+1})$, якщо s_i – відзначений стан, $i = 1, \dots, k+1$. Якщо $\varphi(s_i) = \varepsilon$, то символ ε у слові w_φ не вказується. Множина всіх розмічених експериментів автомата B позначимо Φ_B .

Нехай F – деяка підмножина повністю визначених сильно зв'язних звідних автоматів й $A \in F$ – еталон. Розмічений експеримент $w_\varphi \in \Phi_A$ назовемо контрольним розміченим експериментом для A й F , якщо із приналежності $w_\varphi \in \Phi_B$ для $B \in F$ витікає ізоморфізм автоматів A й B . Контрольний експеримент з однією міткою названо циклічним (КЦЕ). Розглянуто експерименти для автоматів $A \in K_{n,m}$ і нескінченного класу $F(X, Y)$ звідних групових автоматів, де $K_{n,m} \subset F(X, Y)$ клас групових автоматів зі n станами й m вхідними символами. Нехай $d_{\min}^A$ – мінімальна довжина КЦЕ автомата A в алфавіті $\bar{X}$, що є поповненням вхідного алфавіту символами обернених елементів. В [6] доведена

Теорема 7[6]. Кратність звідного КЦЕ дорівнює $mn - n + 1$.

2. Для кожного $A \in K_{n,m} \subseteq F(X, Y)$ справедливі нерівності:

$$k((2m+1)(2m-1)^k - \frac{mn-n+1}{m}) + \\ + (k+1)(\frac{mn-n+1}{m} - (2m-1)^k)(2m-1) + \\ + mn - n + 1 \leq d_{\min}^A \leq n(mn - n + 1),$$

де $k = \lceil \log_{2m-1} \frac{mn-n+1}{m} \rceil$, причому обидві оцінки досяжні.

Оцінимо довжину мінімальних контрольних розмічених експериментів, у яких можлива наявність двох різних міток. Такі експерименти назовемо 2-розміченими, і розглядаємо їх для автоматів $A \in K_{n,m}$ і нескінченного класу $F(X, Y)$, де $K_{n,m} \subset F(X, Y)$ клас групових автоматів зі n станами й m вхідними символами. При цьому розглядаються звідні контрольні експерименти, тобто такі експерименти, видалення будь-якого слова з яких приводить до експерименту, що не є контрольним. Нехай $d_2^{n,m}$ – довжина мінімального контрольного 2-розміченого експерименту для $A \in K_{n,m}$ й $F(X, Y)$.

$$\text{Теорема 8[10]. } d_2^{n,m} \leq n(mn - n + 1) + \\ 2\lfloor l/2 \rfloor((m-1)\lfloor l/2 \rfloor + m - 2) + (m-1)l(l - 2n + 1) + \\ l - 2\lfloor l/2 \rfloor(\lfloor l/2 \rfloor 2(m-1) + 2m - 3),$$

де l – відстань між відзначеними станами в графі переходів автомата A без урахування орієнтації. Співставляючи оцінку з теореми 7 з оцінкою у випадку двох міток, одержуємо, що у варіанті двох міток є виграш по довжині експерименту, рівний $2\lfloor l/2 \rfloor((m-1)\lfloor l/2 \rfloor + m - 2) + (m-1)l(l - 2n + 1) + l - 2\lfloor l/2 \rfloor(\lfloor l/2 \rfloor 2(m-1) + 2m - 3)$, де l – відстань між відзначеними станами.

Висновки

Серед усього розмаїття задач теорії автоматів основна – дослідження відношення «автомат – поведінка». У роботі наведено ряд закінчених результатів у цьому напрямку, що відносяться до класичних типів автоматів. Значна кількість

результатів і методів цієї теорії може бути поширена на неklasичні автоматоподобні моделі, які виникли останнім часом у результаті розвитку формальних методів розробки, верифікації і

супроводу програмно-апаратних систем, теорії мобільних агентів, теорії дискретних подій.

Список використаних джерел

1. Кудрявцев В.Б. Анализ поведения автоматов. / В.Б. Кудрявцев, И.С. Грунский, В.А. Козловский // Дискретная математика. – 2009. – т. 21, вып.1. – С.3 – 35.
2. Грунский И.С. Синтез и идентификация автоматов. / И.С. Грунский, В.А. Козловский. – Киев: Наук. думка, 2004. – 245 с.
3. Крытый С.Л., Формальные методы анализа свойств систем. / С.Л. Крытый, Л.Е. Матвеева. // Кибернетика и системный анализ. – 2003. – № 2. – С. 15 – 36.
4. Максименко И.К. Эксперименты в финитно-определенных метрических пространствах автоматов: Автореферат дис. ... канд. физ.-мат. наук; 01.01.09 / СГУ – Саратов, 2000.
5. Сенченко А.С. Представление автоматов определяющими соотношениями их поведения: Автореферат дис. ... канд. физ.-мат. наук; 01.05.01 / ИК НАН Украины – Киев, 2005.
6. Мучникова Л.А. Контрольные эксперименты с групповыми автоматами: Автореф. дис. ... канд. физ.-мат. наук; 01.05.01 / ИК НАН Украины – Киев, 2006.
7. Hennie F.C. Fault detecting experiments for sequential circuits. / F.C. Hennie // Proc. 5 Annual Symp. "Switch. Circuits Th. and Logic Design". – 1964. – P.95 - 110.
8. Василевский М.П. О распознавании неисправностей автоматов. / М.П. Василевский // Кибернетика. – 1973. – № 4. – С. 93 - 108.
9. Bhattacharyya A. Checking experiments on sequential machines. / A Bhattacharyya. – New York: J. Wiley and Sons, 1989.
10. Козловский В.А. О 2-размеченных экспериментах групповых автоматов / В.А. Козловский, Л.А. Мучникова. // Материалы IX Международного семинара «Дискретная математика и ее приложения», посвященного 75-летию со дня рождения академика О.Б. Лупанова (Москва, МГУ, 18-23 июня 2007г.) / Под ред. О.М. Касим-Заде. – М.: Изд-во мех.-мат. ф-та МГУ. – 2007. – С.324-327.

Надійшла до редколегії 15.12.10

УДК 004.052.42

Губа А.А., студент

Методи синхронізації агентів при верифікації систем в термінах мультиагентних середовищ

У статті розглядаються методи синхронізації поведінок агентів при верифікації систем у термінах мультиагентних середовищ. Описані сценарний і подійний підходи при декомпозиції MSC-сценаріїв і діаграм послідовності подій UML opt-областей на базові протоколи. Запропоновано більш оптимальний алгоритм для генерації SDL.

Ключові слова: верифікація, декомпозиція MSC-сценаріїв, генерація SDL.

*E-mail: Anton.Huba@iss.org.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Для систем зі складною моделлю поведінки стає неможливим обійтися без автоматизації перевірки правильності. Під системою розуміється декілька компонент (об'єктів, агентів), які взаємодіють між собою і з навколишнім середовищем (середовище також може задаватися у вигляді окремого агента).

Часто вимоги до систем задаються у вигляді сценаріїв. Це можуть бути або MSC-сценарії, або діаграми послідовності подій UML (Sequence Diagrams). Для здійснення верифікації заданих вимог у системі VRS [2] необхідно здійснити їх формалізацію або побудувати систему базових протоколів, яка використовується як вхідні дані для трасового генератора (одна з компонент VRS). За допомогою трасового генератора ми отримуємо траси або послідовності базових протоколів, що відображають поведінку системи. У результаті ми отримуємо або траси з задовільним вердиктом про те, що покриті всі протоколи та вимоги, або траси, які призводять до тупикових станів системи і вердикт про непокриті вимоги. У будь-якому випадку траси повинні бути узгоджені зі сценарієм. Дане узгодження полягає в тому, що послідовності подій (ланцюжки подій) на діаграмі повинні збігатися з послідовностями подій на трасах. У статті розглядається одне із завдань, що виникають при декомпозиції сценаріїв на базові протоколи: декомпозиція вбудованих виразів (inline expressions). В якості таких ми використовуємо MSC оператори alt, opt. Якщо

A. A. Guba, student

Methods for synchronization behavior of the agents in verification of systems in terms of multi-agents environment

In the paper we consider the methods for synchronization behavior of the agents in verification of systems in terms of multi-agents environment. This paper also describes scenario and event approaches for decomposition MSC-scenarios and diagrams of the events sequences UML opt-regions on the basic protocols. More optimal algorithm to generate the SDL is proposed.

Key Words: verification, decomposition of MSC-scenarios, SDL generation.

MSC оператор opt з умовою, проблеми із синхронізацією не виникає, тому що сама умова вже є синхронізацією. Безумовні opt-оператори – приклад недетермінізму в динамічних системах. При відсутності синхронізації агентів, які беруть участь у цій області при декомпозиції сценарію, отримуємо абсолютно незалежні ділянки траси, що може суперечити вихідній діаграмі. Це веде до некоректності всієї траси, а також до помилкових тупикових станів.

В якості мови формального опису моделей розглядається мова базових протоколів (БП). Базовий протокол – вираз виду $\forall x(\alpha \rightarrow \langle u \rangle \beta)$ [1,6], де x – список (типізованих) параметрів, α і β – формули базової логічної мови, u – процес протоколу (кінцева поведінка композиції декількох агентів і середовищ MSC діаграм). Формула α називається передумовою, а формула β – постумовою базового протоколу.

В якості базової мови використовується мова багатосортного числення предикатів першого порядку. Формула базової мови може містити змінні і константи наступних простих типів: числові (цілий (int) і дійсний (real)) і символічні (тип Bool, перелічуваний, довільний символічний (symb)). У якості змінних, які змінюють своє значення у процесі функціонування системи, виступають атрибути та атрибутивні вирази. Атрибутними виразами є оператори доступу до елементу масиву (array) за індексами $arg(mas, i)$, функції доступу до списків (lists)

$get_from_head(l)$, $get_from_tail(l)$, $empty(l)$,
атрибутні вирази функціональних типів
 $a(x, y, z)$.

Перевага цього формалізму в простоті й ефективності використання методів верифікації – мова використовує базовий набір структур даних та операцій над ними і при цьому володіє достатньою виразною потужністю для опису систем, специфікованих в таких інженерних мовах, як MSC [3], SDL [5], UML [7].

У цій статті розглянемо також клас базових протоколів, який є частинним випадком конкретних базових протоколів з [2]. Цей клас можна визначити як клас асинхронних імперативних базових протоколів з одним ключовим агентом. Базові протоколи зазначеного класу будемо називати К-протоколами [8]. Асинхронність розуміється в тому сенсі, що обмін інформацією здійснюється по буферизованих каналах. Імперативність означає, що кожен протокол еквівалентний деякій імперативній програмі, а один ключовий агент повністю визначає функціонування базового протоколу.

Далі ми конвертуємо або сценарії, або опис локальних дій (прихід сигналу) в базові та К-протоколи і розглядаємо складності конвертації вбудованих виразів.

Основне призначення діаграм взаємодії – опис допустимих послідовностей подій в системі. Основним поняттям діаграми взаємодії є траса. Для кожного агента на діаграмі є окрема вертикальна вісь. На цій осі відзначаються події, які мають відношення до даного агента. Траса – це послідовність всіх подій від першої взаємодії в системі між агентами і до останнього. Вважається, що всі агенти існують одночасно, і послідовності подій агентів розвиваються паралельно.

Взаємодія між агентами (а також між

$$get_from_head(P.list_of_scenarios) = D \ \& \ get_from_head(P.list_of_labels) = number \quad (1)$$

Відправлення повідомлення $mes1$ агентом A і прийом його агентом B моделюється двома kbp (key basic one agent protocol або К-протоколи [8]). К-протоколи мають семантику SDL [5]: перед- та постумова обмежені одним агентом, заборонена

агентом і оточенням системи) здійснюється тільки за допомогою обміну повідомленнями. Повідомлення моделює взаємодію (обмін інформацією) між двома агентами системи або між агентом і оточенням системи. З точки зору системи взаємодія між двома агентами розбивається на дві пов'язаних події: посилка повідомлення одним агентом і прийом повідомлення іншим агентом. Повідомлення, які приходять від середовища, моделюються однією подією прийому повідомлення, а події, які відсилаються у середовище, моделюються однією подією посилки повідомлення. MSC-діаграми дозволяють створювати складні описи поведінки системи за допомогою спеціальних операторів. У MSC використовуються наступні типи операторів: *alt* – альтернативний оператор, *loop* – цикл, *opt* – опціональна область, *reference* – посилання, *exc* – виконання, *par* – паралельний оператор.

Конвертація сценаріїв – це представлення всіх подій (вхідних, вихідних, локальних) діаграми у формі базових протоколів. Розглянемо детальніше, які складнощі виникають при конвертації діаграми в множину базових протоколів, коли *opt* – спільна область для кількох агентів системи. Ми будемо розглядати роботу системи в термінах текстового формату MSC-діаграм [3].

Обидва агента перебувають у діаграмі D . Для визначення точної позиції агента в сценарії введемо два атрибути типу *list*. Перший *list_of_scenarios* – список діаграм, в які занурений агент; в голові списку – поточна діаграма і другий *list_of_labels* – список позицій агента на діаграмі. Control Flow для агента P – кон'юнкція рівності голови списку *list_of_scenarios* з ідентифікатором діаграми, і рівності голови списку *list_of_labels* з номером позиції в цій діаграмі:

спільна пам'ять, повідомлення інтерпретуються, дозволений тільки один вхідний сигнал для ключового агента.

Перший протокол (тут ключовий агент A):

$$\left(\begin{array}{l} (get_from_head(A.list_of_scenarios) = D_k) \\ \& (get_from_head(A.list_of_labels) = i) \end{array} \right) \rightarrow \left(\begin{array}{l} remove_from_head(A.list_of_labels); \\ add_to_head(A.list_of_labels, i+1); \\ add_to_tail(B.queue, mes1) \end{array} \right) \quad (2)$$

Другий протокол (тут ключовий агент B):

$$\left(\begin{array}{l} (get_from_head(B.list_of_scenarios) = D_K) \\ \& (get_from_head(B.list_of_labels) = j) \\ \& (get_from_head(B.queue) = mes1) \end{array} \right) \rightarrow \left(\begin{array}{l} remove_from_head(B.list_of_labels); \\ add_to_head(B.list_of_labels, i+1); \\ remove_from_head(B.queue) \end{array} \right) \quad (3)$$

Порожній MSC-сценарій в термінах БП (тут ключовий агент будь-який):

$$\left(\begin{array}{l} (get_from_head(A.list_of_scenarios) = D_K) \\ \& (get_from_head(A.list_of_labels) = i) \\ \& (get_from_head(A.list_of_scenarios) = D_K) \\ \& (get_from_head(B.list_of_labels) = j) \end{array} \right) \rightarrow \left(\begin{array}{l} remove_from_head(B.list_of_labels); \\ add_to_head(B.list_of_labels, j+1); \\ remove_from_head(A.list_of_labels); \\ add_to_head(A.list_of_labels, i+1); \end{array} \right) \quad (4)$$

Даний підхід забезпечує коректність вихідних трас і незалежність агентів один від одного при комунікації повідомленнями. Але протоколи перестають бути kbp, втрачається семантика SDL і відсутній інтерлівінг у випадку

порожньої MSC-діаграми, що є недоліками даного підходу. Розділимо протокол (4) на два, як це було зроблено при передачі та прийомі повідомлення:

Перший протокол (ключовий агент A):

$$\left(\begin{array}{l} (get_from_head(A.list_of_scenarios) = D_K) \\ \& (get_from_head(A.list_of_labels) = i) \end{array} \right) \rightarrow \left(\begin{array}{l} remove_from_head(A.list_of_labels); \\ add_to_head(A.list_of_labels, i+1) \end{array} \right) \quad (5)$$

Другий протокол (ключовий агент B):

$$\left(\begin{array}{l} (get_from_head(B.list_of_scenarios) = D_K) \\ \& (get_from_head(B.list_of_labels) = j) \end{array} \right) \rightarrow \left(\begin{array}{l} remove_from_head(B.list_of_labels); \\ add_to_head(B.list_of_labels, j+1) \end{array} \right) \quad (6)$$

Після протоколу (2) може виконатися протокол (6), і протокол (3) вже прийняти і обробити повідомлення не зможе. Таким чином, виникає проблема синхронізації в opt-областях MSC-діаграм і необхідно «повідомити» іншому агенту, коли перший заходить або проходить опціональну область.

В системі є N агентів. OptController – додатково занурений агент в середовище, необхідний для здійснення синхронізації.

Крок 1. Для кожної opt-області завести атрибут opt_id в кожному з агентів, які

приймають участь в цій області и глобальний атрибут. Це ідентифікатори opt-області.

Крок 2. Початкове значення кожного з них Unknown.

$$\forall i = \overline{1..N} : P_i.opt_id := Unknown; \\ opt_id := Unknown$$

Крок 3. Для кожного з агентів P_i , які входять в opt-область, повинні бути присутніми протоколи:

$$\left(\begin{array}{l} (opt_id = Unknown) \\ \& get_from_head(P_i.list_of_labels) = k_i \end{array} \right) \rightarrow \left(\begin{array}{l} opt_id := true; \\ P_i.opt_id := true; \\ remove_from_head(P_i.list_of_labels); \\ add_to_head(P_i.list_of_labels, k_i + 1) \end{array} \right)$$

$$\left(\begin{array}{l} (opt_id = Unknown) \& \\ get_from_head(P_i.list_of_labels) = k_i \end{array} \right) \rightarrow \left(\begin{array}{l} opt_id := false; \\ P_i.opt_id := false; \\ remove_from_head(P_i.list_of_labels); \\ add_to_head(P_i.list_of_labels, q_i) \end{array} \right)$$

де q_i – позиція агента на діаграмі після opt-області.

Крок 4. Для кожного з агентів P_i після останньої дії в opt-області повинен бути

присутнім протокол:

$$(get_from_head(P_i.list_of_labels) = q_i - 1) \rightarrow \left(\begin{array}{l} opt_id := false; \\ P_i.opt_id := false; \\ remove_from_head(P_i.list_of_labels); \\ add_to_head(P_i.list_of_labels, q_i) \end{array} \right)$$

Крок 5. Для агента OptController повинен бути присутнім протокол:

$$(\forall i(P_i.opt_id = exitopt)) \rightarrow \left(\begin{array}{l} opt_id := Unknown; \\ P_i.opt_id := Unknown \end{array} \right)$$

та

$$\forall i(P_i.opt_id = false) \rightarrow \left(\begin{array}{l} opt_id := Unknown; \\ P_i.opt_id := Unknown \end{array} \right)$$

Крок 6. Для кожного з агентів перед першою дією після opt-області повинен бути присутнім протокол, передумова якого має вигляд:

$$(get_from_head(P_i.list_of_labels) = q_i) \& (opt_id = Unknown)$$

У даній роботі розглянуті проблеми відповідності трас і діаграм, сценарний і подійний підходи при розкладанні MSC-сценаріїв і діаграм послідовності подій UML opt-областей на базові протоколи. Представлений алгоритм забезпечує коректність трас, зменшує складність генерації SDL-коду, ефективно забезпечує інтерлівінг агентів. Алгоритм вводить

синхронізацію для агентів, які беруть участь в безумовних opt-областях. Таким чином, при декомпозиції сценарію отримуємо абсолютно коректні ділянки траси, що не суперить вихідній діаграмі і не веде до помилкових тупиковим станам. Надалі планується забезпечити оптимізацію алгоритму для opt-областей високої вкладеності.

Список використаних джерел

1. Letichevsky A. Basic Protocols, Message Sequence Charts, and the Verification of requirements Specifications / A. Letichevsky, J.Kapitonova, A.Letichevsky Jr., V.Volkov, S.Baranov, V.Kotlyarov, T.Weigert // Computer Networks. – 2005. – P. 662-675.
2. Letichevsky A. System Specification with Basic Protocols / A. Letichevsky, J.Kapitonova, A.Letichevsky Jr., V.Volkov, S.Baranov, V.Kotlyarov, T.Weigert // Cybernetics and System Analyses, 4. – 2005. – P. 1-14.
3. International Telecommunications Union. Recommendation Z.120 Formal description techniques (FDT) – Message Sequence Chart (MSC), 2004. – P. 26-28
4. Letichevsky A.A. Insertion modeling in distributed system design / A. Letichevsky, J.Kapitonova, A.Letichevsky Jr., V.Volkov, S.Baranov, V.Kotlyarov, T.Weigert // Проблемы програмування. – 2008. №4. – P. 13-35.
5. ITU-T Recommendation Z.100 Specification and Description Language (SDL), ITU – 2002. – 301 p.
6. Потієнко С.В. Методи прямого и обратного моделирования систем, заданных базовыми протоколами / С.В. Потієнко // Проблемы програмування. – 2008. №4. – P. 41-45.
7. ITU-T Recommendation Z.109 – SDL combined with UML. – ITU, 1999. – P. 627-642.
8. Летичевский А.А. Об одном классе базовых протоколов / А.А. Летичевский // Проблемы програмування. – 2005. №4. – P. 3-14.

Надійшла до редколегії 15.12.10

УДК 004.622

Гученко І.В., аспірантка

Аналіз експертної бази на узгодженість методами Data Mining

Статтю присвячено питанню дослідження узгодженості оцінок експертів. Обґрунтовується недостатність обчислення коефіцієнта конкордації Кендела для якісного аналізу ситуації. Пропонується застосування методу карт Кохонена для виявлення структури експертних ранжувань та причин їх неузгодженості.

Ключові слова: інтелектуальний аналіз даних, аналіз узгодженості, експерти, ранжування, карти Кохонена.

E-mail: inna.zaporozhets@livenau.net

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Попередні дослідження автора були присвячені створенню методу автоматизованої оцінки зручності використання програмної системи при agile-розробці з метою врахування та прогнозування вимог користувачів [1]. Вказаний метод дозволяє не лише оцінювати поточний рівень зручності застосування програмних систем, але й керувати якістю процесів їхнього створення на основі математичних оптимізаційних моделей. Важливим для реалізації методу є аналіз експертних оцінок (а також оцінок користувачів) на узгодженість, що і є предметом дослідження у даній статті.

Кількісний та якісний аналіз узгодженості експертів

Експертиза представляє собою потужний засіб обробки низько формалізованих даних, який дозволяє виділити найбільш обґрунтовані твердження спеціалістів-експертів та використовувати їх для підготовки різних рішень [2]. Виробленню колективного рішення має передувати дослідження результатів експертного опитування з метою виявлення узгодженості суджень усіх експертів.

Оцінки експертів часто вимірюють у порядковій шкалі (типовими є задачі ранжування та класифікації об'єктів). Як показали численні дослідження, людина більш вірно і з меншими

I.V. Guchenko, postgraduate

Experts' base concordance analysis with Data Mining methods

The article is dedicated to concordance analysis of experts' rates. Inefficiency of Kendall's coefficient evaluation for qualitative analysis is grounded. Kohonen maps method is proposed for detection of experts' rankings' structure and causes of disagreement.

Key Words: Data Mining, concordance analysis, experts, ranking, Kohonen maps.

утрудненнями відповідає на питання якісного, порівняльного характеру, ніж кількісного [3].

Специфічними є підходи до перевірки узгодженості, що використовуються при оцінці об'єктів методом ранжування. У цьому випадку результатом роботи експерта є ранжування, що представляє собою послідовність рангів (для експерта j): $x_{1j}, x_{2j}, \dots, x_{nj}$.

Узгодженість між ранжуваннями двох експертів можна визначити за допомогою коефіцієнта рангової кореляції Спірмена:

$$r = 1 - \frac{6 \sum_{i=1}^n (x_{ij} - x_{ik})^2}{n(n^2 - 1)} = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2 - 1)},$$

де x_{ij} – ранг, наданий i -му об'єкту j -им експертом; x_{ik} – ранг, наданий i -му об'єкту k -им експертом; d_i – різниця між рангами, наданими i -му об'єкту; n – число парних членів ряду, або об'єм вибірки.

Величина r може змінюватися в діапазоні від -1 до $+1$. При повному співпадінні оцінок коефіцієнт дорівнює 1 , при найбільшому розходженні -1 .

При необхідності визначення узгодженості ранжувань більше, ніж двох експертів, розраховується коефіцієнт конкордації – загальний коефіцієнт рангової кореляції для групи, що складається з m експертів:

$$W = \frac{12 \cdot S}{m^2(n^3 - n)},$$

$$\text{де } S = \sum_{i=1}^n \left(\sum_{j=1}^m x_{ij} - \frac{1}{2} m(n+1) \right)^2.$$

Коефіцієнт конкордації змінюється в діапазоні від 0 до 1. Чим ближче коефіцієнт до 0, тим менш узгодженими є оцінки експертів.

В аналізі експертної бази на узгодженість застосовують найчастіше розрахунок коефіцієнта конкордації Кендела. Останній не дає можливості якісного аналізу ситуації, оскільки не вказує на експертів-«диссидентів», що є причиною неузгодженості, а також не дає інформації про об'єкти, що мають найбільші розбіжності при ранжуванні експертами. До того ж при обчисленні коефіцієнта конкордації перевіряється нульова гіпотеза, згідно якої ранжування незалежні та рівномірно розподілені на множині усіх ранжувань. Якщо ця гіпотеза приймається, то не можна говорити про узгодженість думок експертів. А якщо відхиляється? Також не можна. Оскільки, наприклад, може бути два (або більше) центри, навколо яких групуються думки.

На противагу недолікам обчислення коефіцієнту конкордації, було запропоновано метод узгодження кластеризованих ранжувань [3], що дозволяє «загнати» протиріччя всередину спеціальним чином побудованих кластерів (груп), у той час як упорядкування кластерів відповідає усім вихідним упорядкуванням.

Для більш повного аналізу узгодженості експертних оцінок пропонується застосовувати методи Data Mining для вирішення задачі кластеризації, додатково розраховуючи коефіцієнт конкордації Кендела.

Вибір методів Data Mining для аналізу структури ранжувань

Інтелектуальний аналіз даних (англ. Data Mining) — це процес виявлення в сирих даних, до цього невідомих, нетривіальних, практично корисних і доступних інтерпретації знань, необхідних для прийняття рішень в різних сферах людської діяльності.

Якщо класифікувати методи Data Mining згідно задач, то вирізняють описові та прогнозуючі методи. До методів, направлених на отримання описових результатів, відносять ітеративні методи кластерного аналізу, у тому числі: алгоритм k-середніх, k-медіани, ієрархічні методи кластерного аналізу, карти Кохонена, різноманітні методи візуалізації тощо [4].

За способом розбиття на кластери алгоритми бувають ієрархічні та неієрархічні. Класичні ієрархічні алгоритми працюють лише з категорійними атрибутами, коли будується повне дерево вкладених кластерів. Відповідно для задачі визначення узгодженості оцінок експертів, які визначені на порядковій (числовій) шкалі, ієрархічні методи не підходять. Серед неієрархічних методів найбільш відомі алгоритм k-середніх та його модифікація — k-медіан. Цей алгоритм занадто чутливий до викидів, які можуть викривляти середнє; повільно працює на великих базах даних; не підходить для застосування на даних, де можливі кластери, що перетинаються. З огляду на вказані недоліки відкидаємо алгоритм k-середніх для задачі оцінки узгодженості думок експертів.

Карти Кохонена, що застосовуються для кластеризації багатовимірних векторів, мають значні переваги: використання універсального апроксиматора — нейронної мережі; навчання без учителя; самоорганізація мережі; простота реалізації; гарантоване отримання відповіді. Вони призначені для роботи з будь-якими числовими даними (у тому числі і порядковими). Серед недоліків визначають питання мінімізації мережі та необхідність задання кількості кластерів (як і в алгоритмі k-середніх).

З огляду на наведений аналіз пропонується застосовувати карти Кохонена для вирішення задачі оцінки узгодженості думок експертів шляхом вивчення структури сукупності ранжувань.

Результати застосування карти Кохонена для аналізу узгодженості експертів

Мережі, що називаються картами Кохонена (Self-Organizing Kohonen Maps, SOM), — це один із різновидів нейронних мереж, однак принциповою відмінністю є використання неконтрольованого навчання. При цьому навчальна множина складається лише із значень вхідних змінних, у процесі навчання немає порівняння виходів нейронів з еталонними значеннями. Головна мета — не мінімізація помилки, а налаштування ваги (внутрішніх параметрів нейронної мережі) для найбільшого співпадіння з вхідними даними. Така мережа вчиться розуміти структуру даних.

Перевіримо та розкриємо переваги застосування карт Кохонена для перевірки узгодженості експертної бази на прикладі даних про оцінку десяти характеристик (об'єктів) десятима експертами (рис. 1).

Об'єкт (A _j)										
Експерт (E _i)	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
E1	2	9	1	6	5	4	8	7	2	3
E2	1	3	2	6	5	4	7	9	8	10
E3	2	4	1	6	6	5	7	9	8	3
E4	2	4	1	5	6	10	9	8	7	3
E5	3	4	1	6	5	8	7	9	2	4
E6	2	4	1	6	5	8	7	9	7	3
E7	1	3	2	6	5	7	8	9	10	4
E8	2	4	1	5	6	8	7	9	10	3
E9	1	3	3	8	6	5	7	9	2	4
E10	1	3	1	6	5	4	7	8	9	2

Рис. 1. Дані з оцінки експертами десяти умовних об'єктів

Оцінки виставлені в порядковій шкалі і представляють собою ранги.

Розрахунки, наведені далі в роботі, виконані за допомогою рішення SAS Enterprise Miner. У вказаному Data Mining інструменті елемент SOM/Kohonen містить три алгоритми: вектор квантування (дискретизації) Кохонена (Kohonen vector quantization); карта Кохонена (Kohonen self-organizing map); групова карта (Batch self-organizing map).

Вектор квантування Кохонена. Цей алгоритм має на меті відшукування підмножини, що представляє найбільший набір векторів даних. При цьому відбувається урізання вибірки векторів або заміна їх центрами мас (centroids). Балакрішнан, Купер, Якоб та Льюїс визначили, що алгоритм k-середніх визначає членів кластера точніше, ніж вектор квантування Кохонена [5].

Спираючись на вищенаведене, відкинемо алгоритм вектору квантування для перевірки експертної бази на узгодженість.

Карта Кохонена. Самоорганізовні карти виявились цінним інструментом візуалізації багатовимірних даних. Вони застосовуються здебільшого для перетворення нелінійних статистичних зв'язків між багатовимірними даними в прості геометричні зв'язки їх точкових образів на двовимірній (як правило) площині.

Така мережа є спеціальним випадком мережі, що навчається методом змагання, у якій визначається просторовий окіл для кожного вихідного елемента. Локальний окіл може бути квадратом, прямокутником чи колом. Початковий розмір околу встановлюється в межах від 1/2 до 2/3 розміру мережі й скорочується згідно певного закону (наприклад, згідно експоненційної спадної залежності). Під час навчання модифікуються усі ваги, пов'язані з

переможцем та його сусідніми елементами. Алгоритм навчання самоорганізовних карт досить стійкий щодо помилок.

Якщо для знаходження ядра регресії (при тренування самоорганізовної карти) використовується метод непараметричної регресії Надарая-Ватсона, то алгоритм групової карти (Batch self-organizing map) видає ті ж самі результати, що й карта Кохонена за винятком локальних мінімумів. [5].

Застосуємо карту Кохонена до аналізу структури експертних ранжувань. Це потребує попереднього самостійного визначення розміру карти (відповідно – кількості кластерів). Основні критерії: не повинно бути порожніх кластерів; середнє значення змінних в кластерах має суттєво відрізнитись (у протилежному випадку не має сенсу розглядати декілька кластерів замість одного). Визначимо розмір карти як 2x2, що означає виділення в структурі чотирьох кластерів, оскільки при розмірі 2x3 один з шести кластерів виявляється порожнім, а два кластери дуже подібні. Метод ініціалізації центрів – Separate, радіус (мінімальна відстань між центрами) – одиничний. Результати містять наступні вкладки: карта, змінні, відстані, профайли, статистика, вивід даних. Вкладка відстаней (Distances Tab) відображає графічне представлення розміру кожного кластеру і зв'язки між ними. Радіус залежить від найбільш віддаленого елемента кластеру. Отримана кількість кластерів – 4 – дає інформацію про те, що узгодженість експертів не є високою. Кластери не мають між собою спільних елементів, про що свідчить перегляд таблиці змінних та їх приналежності до кластерів (див. рис. 2).

Expert (Ei)	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Cluster ID	Distance	Row	Column	SOM ID
E1	2	9	1	6	5	4	8	7	2	3	1	1.0312210272	1	1	1:1
E2	1	3	2	6	5	4	7	9	8	10	2	0.5582239502	1	2	1:2
E3	2	4	1	6	6	5	7	9	8	3	3	0.3978074249	2	1	2:1
E4	2	4	1	5	6	10	9	8	7	3	1	1.1726873462	1	1	1:1
E5	3	4	1	6	5	8	7	9	2	4	1	0.8240422852	1	1	1:1
E6	2	4	1	6	5	8	7	9	7	3	1	0.7078788053	1	1	1:1
E7	1	3	2	6	5	7	8	9	10	4	2	0.8281329898	1	2	1:2
E8	2	4	1	5	6	8	7	9	10	3	3	0.3971408754	2	1	2:1
E9	1	3	3	8	6	5	7	9	2	4	2	0.9493180096	1	2	1:2
E10	1	3	1	6	5	4	7	8	9	2	4	0.1153390058	2	2	2:2

Рис. 2. Приналежність експертних ранжувань до кластерів

Отже, до першого кластеру належать ранжування експертів E1, E4-E6; до другого кластеру – E2, E7 та E9; до третього – E3 та E8; у четвертому кластері міститься лише експерт E10. Зрозуміло, що експерт E10 є так званим «дисидентом» і необхідно перевірити його кваліфікацію та причини такої відмінності.

Також цінну інформацію надає перегляд дерева та правил кластеризації (див. рис. 3).

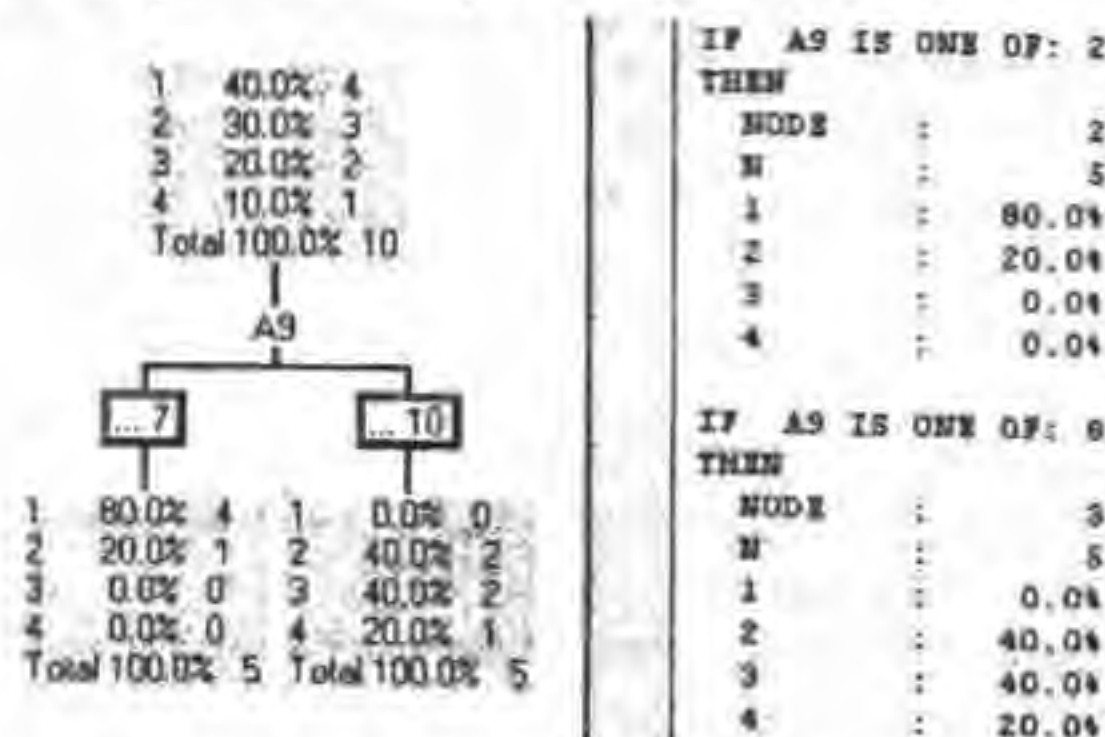


Рис. 3. Дерево кластеризації методом карти Кохонена

Очевидно, що розщеплення відбувається по об'єкту A9, який вносить суттєве протиріччя в судження експертів. Отримані висновки неможливо побачити при застосуванні лише коефіцієнта конкордації, який в даному випадку

дорівнює 0,68808. Розрахунки виконані за допомогою макроса MAGREE [6] в середовищі SAS. Значення "Prob>F" менше за .0001, що означає значимість результату з 99% рівнем достовірності. Якісна оцінка показників тісноти дається, як правило, за шкалою Чеддока. Таким чином, узгодженість експертної бази є помітною і цей результат є значимим. Завдяки отриманому за допомогою карти Кохонена якісному аналізу, можна усунути фактори, що заважають отримати високу узгодженість експертної бази.

Висновок

Важливість експертизи як засобу прийняття рішень ставить питання про якість аналізу узгодженості експертної бази. Загальноприйняті підходи, пов'язані з коефіцієнтами конкордації Кендела та рангової кореляції Спірмена, мають суттєве обмеження – узгодженість оцінюється одним числом, що не дає якісної інформації. Для вирішення цієї проблеми в роботі запропоновано використання методів та інструментів Data Mining, зокрема кластерного аналізу методом карт Кохонена для виявлення структури експертних ранжувань. У сукупності з обчисленням коефіцієнта конкордації вказаний метод дає повну картину узгодженості експертної бази.

Список використаних джерел

1. Тараненко К.Г., Гученко І.В. Автоматизований аналіз та оцінка зручності використання програмних систем. Системний аналіз та інформаційні технології: матеріали 12-ї Міжнародної наук.-техн. конференції SAIT 2010. – К.: ННК "ІПСА" НТУУ "КПІ", 2010. – 544 с.
2. Сидельников Ю.В. Моделирование поведения и интеллекта: экспертология – новая научная дисциплина // Автоматика и телемеханика. – 2000. – Вып. 2. – С. 107.

3. Горский В.Г., Орлов А.И., Гриценко А.А. Метод согласования кластеризованных ранжировок // Автоматика и телемеханика. – 2000. – Вып. 3. – С. 159-167.
4. Чубукова И.А. Data Mining. – М.: Интернет-Университет Информационных технологий; БИНОМ. Лаборатория знаний, 2006. – 382 с.: ил., табл.
5. Ian Witten, E. Frank. Data mining: practical machine learning tools and techniques – 2nd ed.: Elsevier Inc, 2005. – 525p.
6. <http://support.sas.com/kb/25/006.html>

Надійшла до редколегії 12.11.10

УДК 004.942:548.4

Д'яченко* Л.І., Мінов Є.В., Остапов С.Е., д.ф.-
м.н., проф., Фочук П.М., д. хім. н., проф.

Програмний комплекс моделювання точкових дефектів у напівпровідникових кристалах

Для моделювання точкових дефектів у напівпровідникових кристалах нами було розроблено програмний комплекс відновлення тривимірного зображення кристалу з серії знімків у інфрачервоних променях. В програмному комплексі існує можливість налаштування параметрів розпізнавання та отримання інформації про кількість дефектів певних груп у вигляді гістограм. Також можна відфільтрувати дефекти конкретної групи у результатуючому тривимірному зображенні.

Ключові слова: точкові дефекти, напівпровідники, кристали, моделювання.

*E-mail: lilya.d82@gmail.com

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

При дослідженні структури точкових дефектів у напівпровідникових кристалах велику допомогу можуть надати засоби комп'ютерного моделювання. Проте, спеціалізованих програм на ринку практично немає. Тому нами був розроблений спеціалізований програмний комплекс який дозволяє моделювати структуру точкових дефектів у напівпровідникових кристалах базуючись на серії знімків у інфрачервоному (ІЧ) випромінюванні.

Опис програмного забезпечення

Розроблений програмний комплекс складається з двох частин: розрахунку концентрацій точкових дефектів у напівпровідникових кристалах та моделюванні структури напівпровідникового кристалу на основі серії інфрачервоних знімків [1].

Розрахунок концентрацій дефектів здійснюється на основі чисельного розв'язку рівняння електронейтральності (РЕН) [2]. Дане РЕН можна ввести вручну або використати наявні конфігураційні файли. В програмі передбачені два конфігураційні файли: для кристалів телуриду кадмію та легованих індієм кристалів CdTe. Можна створювати власні конфігураційні файли для довільних кристалів.

L. D'yachenko, E. Minov, S. Ostapov, P. Fochuk

The program complex for point defects modeling at semiconductor crystals

The program complex for point defects modeling at semiconductor crystals was developed. It was intended for designing of crystal three-dimensional image from the infrared photos series. In the program complex there is a possibility of recognition parameters adjustment and obtaining the information of point defects from certain groups in the form of histograms. Also it is possible to filter the defects of specific group in the outgoing three-dimensional image.

Key Words: point defects, semiconductor, crystals, modeling.

Усі змінні, що використовуються для розв'язку рівняння та константи рівноваги повинні бути оголошені у верхньому вікні форми разом з їх значеннями.

Уведене рівняння електронейтральності потрапляє на лексичний аналізатор, який перевіряє наявність усіх необхідних даних для його розв'язку та відповідність прийнятому в програмі стандарту. Фізична правильність рівняння не може бути перевірена і покладається на користувача.

Сконструйоване рівняння конвертується з текстової в аналітичну форму і розв'язується методом дихотомії. Результати розв'язку подаються користувачеві у вигляді таблиці результатів, яка може бути експортована у Microsoft Excel, та роздрукована у вигляді графіку.

Крім розрахунку концентрації дефектів, розроблене програмне забезпечення надає можливість змоделювати процес охолодження кристалів CdTe, завантаживши початкові дані (тиск пари кадмію та температуру) з файлу вхідних даних формату *.xls. Файл вхідних даних складається з трьох стовбців: температура, тиск пари кадмію та дата вимірювання.

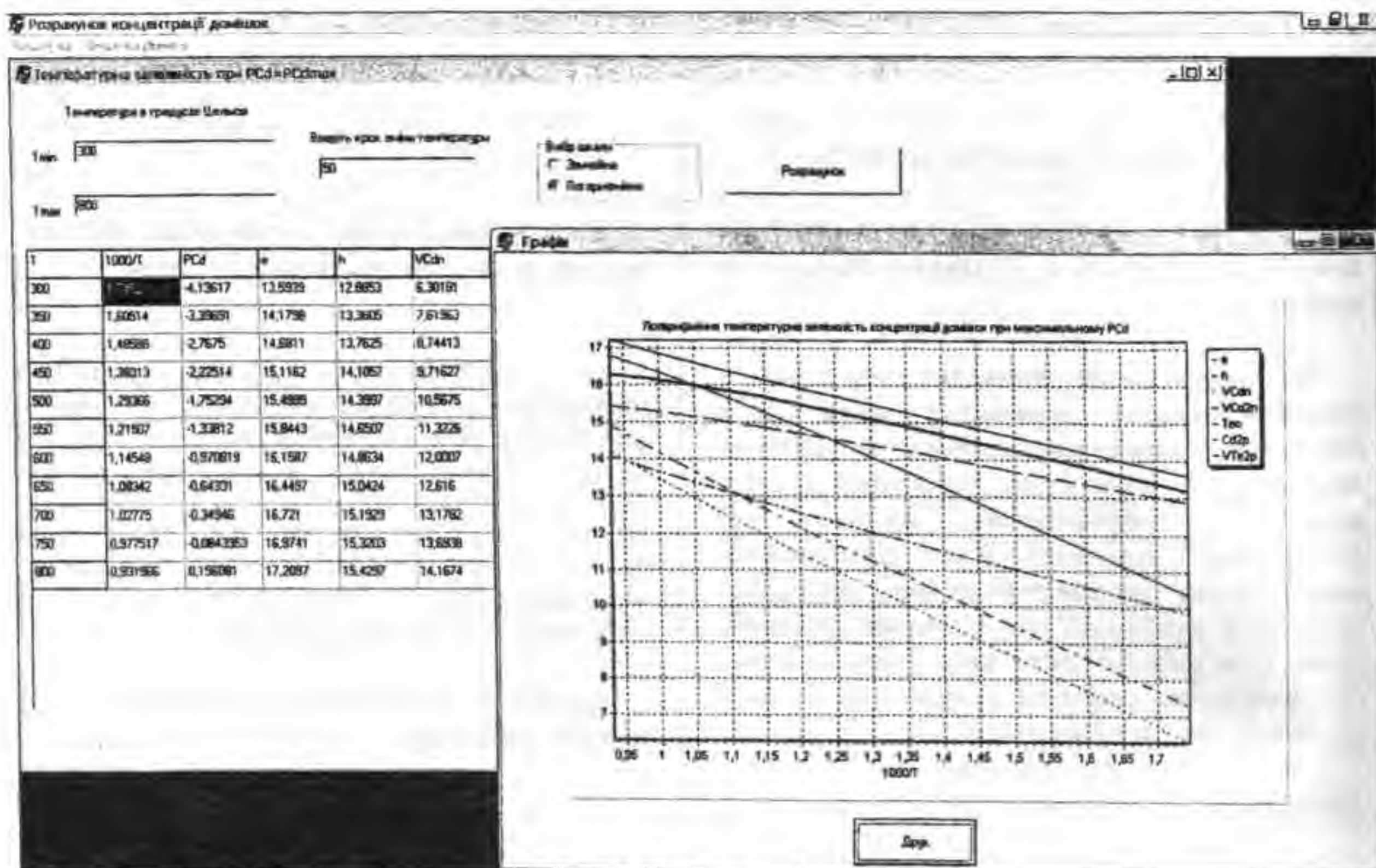


Рис. 1. Інтерфейс модуля розрахунків та візуалізації даних.

Моделювання структури напівпровідникового кристалу здійснюється на основі серії знімків у ІЧ випромінюванні, що зберігаються в форматі *.jpg. Програмний комплекс спочатку проводить розпізнавання дефектів на запропонованих знімках, а потім будує тривимірне зображення кристалу [3]. В процесі розпізнавання використовуються п'ять основних алгоритмів: normalize (контраст), blur (швидке розмиття), unsharp mask (нечітка маска), filtering (фільтрація по площі дефекту), threshold (поріг). Процес розпізнавання складається з наступних кроків:

- Нормалізація вхідних даних по межі чорного і білого (одноразово під час завантаження з файлу);
- Розмиття точок зображення заданого радіуса;
- Розрахунок нечіткої маски на основі результату розмиття і оригіналу;
- Фільтрування отриманої маски по концентрації дефектів на заданій площі;
- Приведення точок маски до білого за заданим порогом.

Результати роботи програми наведені на рис. 2 та 3.

Програмний комплекс надає засоби для редагування розпізнаного зображення: можна витерти дефекти, спричинені нерівностями

поверхні та дефекти самого знімку [4]. Проведені зміни можна задіяти до поточного шару або до всіх розглядуваних шарів, включаючи побудоване тривимірне зображення. Це дозволяє усунути помилки розпізнавання зображень та побудувати тривимірне зображення кристалу з точним відображенням дефектів, що знаходяться на кожному шарі.

На рис. 4 наведене побудоване тривимірне зображення кристалу, з можливістю перегляду кожного конкретного шару. Різні групи дефектів зображуються різним кольором для зручності. Також існує можливість відображати конкретну групу точкових дефектів. Це дозволяє переглянути тільки ті дефекти які цікавлять користувача на даний момент не показуючи інші групи точкових дефектів. Градація здійснюється за діаметром дефекту. За замовчуванням поділ здійснюється наступним чином: дефекти діаметром до 5 мкм, від 5 до 20 та більше 20 мкм. Побудоване тривимірне зображення можна повертати довільним чином, користуючись мишкою. Діаметр різних груп дефектів можна змінити натиснувши кнопку «Настройка» в нижньому лівому кутку вікна, яке зображене на рис. 4.

Головне меню
Навігатор
Додаток

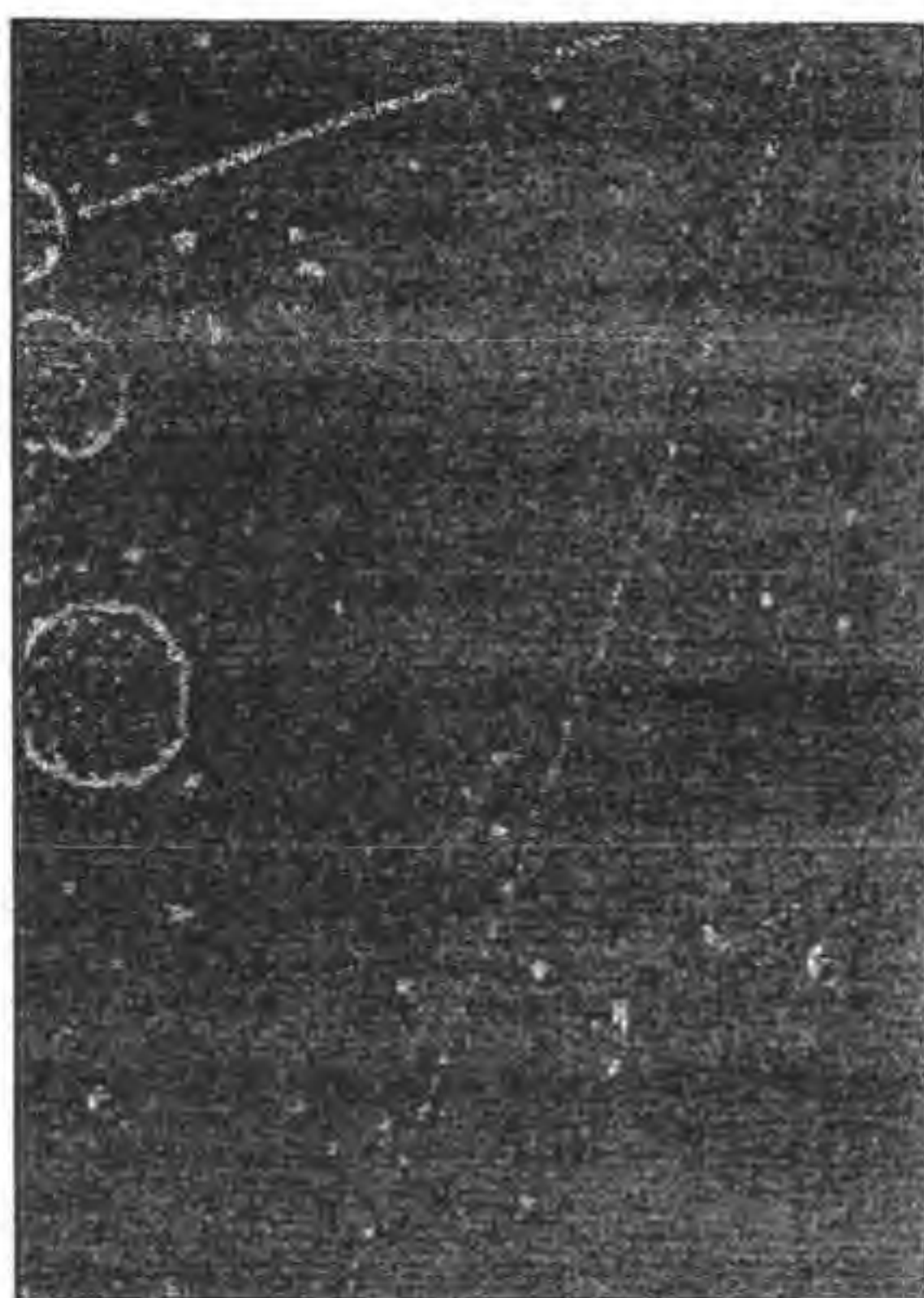


Рис. 2. Розпізнавання початкових знімків

В параметрах налаштування також можна змінити мінімальний розмір дефектів, що розпізнаються та інші важливі параметри розпізнавання.

Вікно налаштування параметрів розпізнавання наведено на рис. 5.

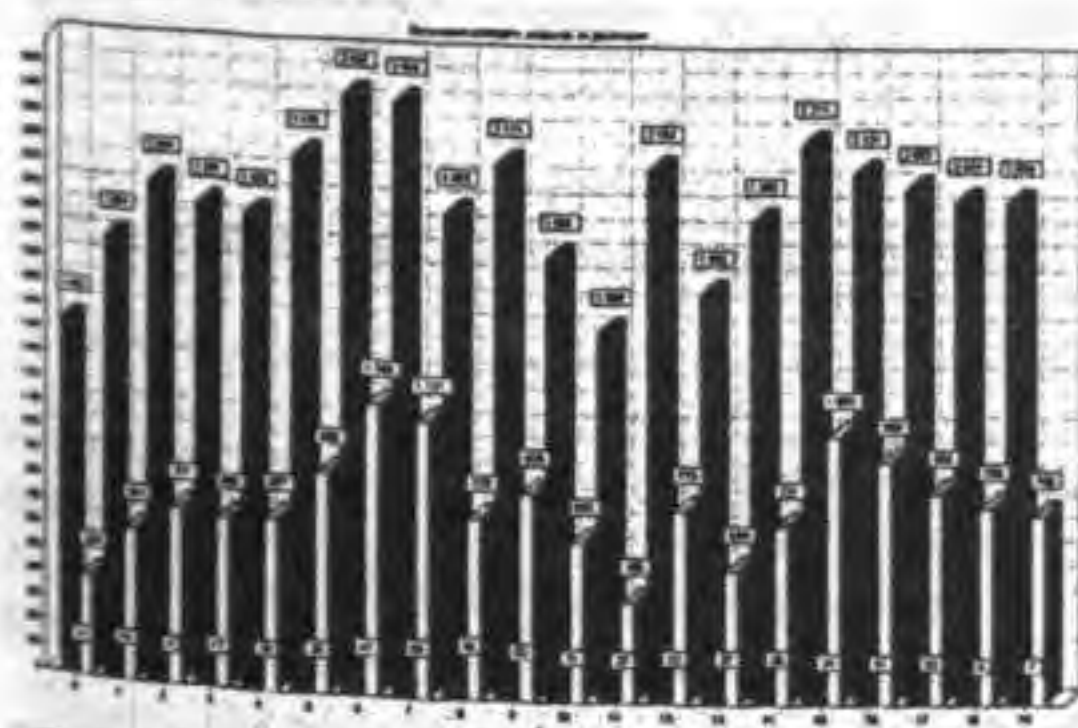


Рис. 3. Гістограма розподілу дефектів по шарах.

Розроблений програмний комплекс також дає можливість провести статистичну обробку отриманих результатів розпізнавання та побудувати гістограми розподілу точкових

дефектів за діаметром на кожному розглядуваному шарі [5]. Приклад побудови таких гістограм наведений на рис. 3. Дана гістограма дозволяє переглянути статистику по кількості дефектів трьох груп на кожному з розглядуваних шарів. Для кожної групи дефектів інформація виводиться іншим кольором.

Висновки

Отже, можна зробити висновок, що розроблений програмний комплекс надає широкий спектр можливостей для дослідження структури власних точкових дефектів напівпровідникових кристалів, а також для обробки результатів хімічних експериментів з реальними кристалами CdTe. Можливість отримання тривимірного зображення та статистичний аналіз розподілу дефектів значно спрощує процес візуалізації та оцінки дефектів у вирощених напівпровідникових кристалах. Наявність засобів редагування розпізнаних зображень дозволяє досягнути високої точності при побудові тривимірного зображення напівпровідникового кристалу.

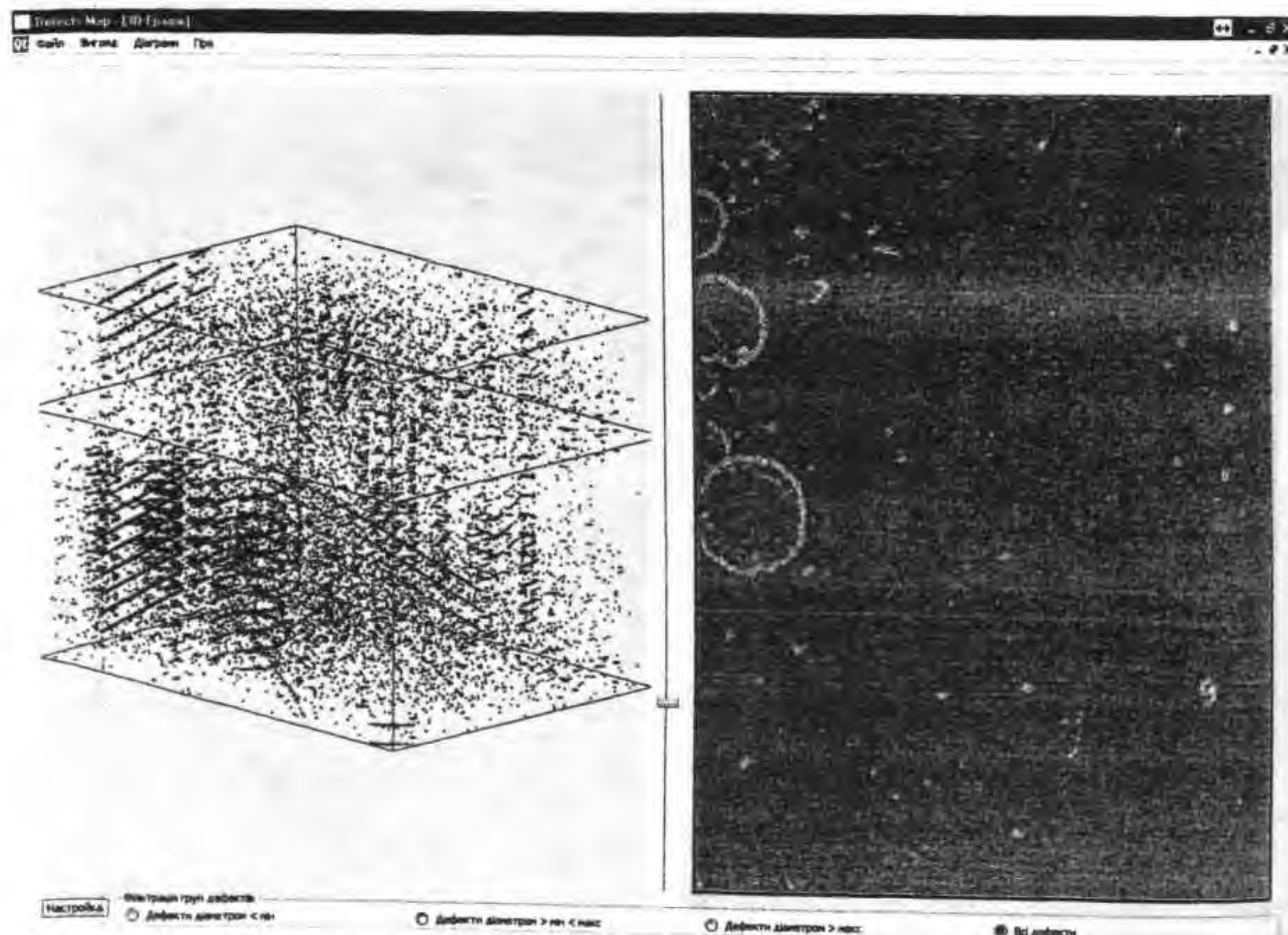


Рис. 4. Побудова тривимірного зображення кристалу

Ширина слою (мм)	1,50	Дільник розміру зображень	5
Довжина слою (мм)	2,04	Радіус розпізнавання	4,00
Мін. діаметр дефекту (мкм)	5,00	Поріг	280
Макс. діаметр дефекту (мкм)	20,00	Радіус фільтрації	35,00

Задіяти до всіх слоїв

Задіяти

Закрити

Настройка

Фільтрація груп дефектів

Дефекти діаметром < мін

Дефекти діаметром > мін < макс

Дефекти діаметром > макс

Всі дефекти

Рис. 5. Налаштування параметрів розпізнавання

Список використаних джерел

1. У.Прэтт. Цифровая обработка изображений. – В 2-х книгах, Кн.1.- М.: "Мир", 1982.-Т.1.- 310с.
2. P.Fochuk, R.Grill, O.Panchuk, The Nature of Point Defects in CdTe// J. Electron. Mater.- 2006. – V.35, No.6. - PP. 1354-1359.
3. И.К. Лурье, А.Г. Косиков. Теория и практика цифровой обработки

- изображений /Дистанционное зондирование и географические информационные системы. Под ред. А.М. Берлянта. -М.: Научный мир, 2003. - 168 с.
4. Р.Дуда, П.Харт. Распознавание образов и анализ сцен. М., Мир, 1976, 512 с.
5. А. Я. Архангельский. Программирование в Delphi 7.:Бином-Пресс,2003 г., 1152 с.

Надійшла до редколегії 09.12.10

УДК 004.384

А.С. Єпіфанов, аспірант

Автоматна інтерпретація генетичних послідовностей

Використання нового способу задання законів функціонування автоматів у формі символічних та числових графіків (геометричних образів законів функціонування автоматів) дозволяє проводити аналіз властивостей об'єктів та процесів, представлених математичними структурами у формах геометричних кривих та послідовностей, з відповідними їх інтерпретаціями. Спеціальний спектр числових показників рекурентного визначення послідовностей дозволяє давати достатньо глибоку числову оцінку будові послідовностей, а на цій базі отримувати оцінки складності автоматів, що породжують криві та послідовності. Властивості природніх об'єктів та процесів, представлених геометричними кривими та послідовностями, допускають оцінку складності по числовим показникам спектру. У роботі містяться результати проведеного аналізу фрагментів послідовностей ДНК людини та інших приматів, а також послідовностей РНК різних вірусів. Послідовність ДНК (РНК) розглядається як послідовність других координат точок геометричного образу автомата. У роботі здійснюється побудова автоматів по фрагментам ДНК, досліджується залежність числа станів автомату від потужності вхідного алфавіту. На основі побудови та аналізу спектрів числових показників для розглядуваних у роботі послідовностей ДНК и РНК побудовані класи еквівалентних по складності послідовностей.

Ключові слова: автомат, геометричний образ законів функціонування автомату, генетична послідовність, спектр динамічних параметрів.

*E-mail: epifanovas@list.ru
Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.
Introduction

The finite state machines as mathematical models were offered, since 1943 (by Mac-Calloc and Pitts) for the description of communications of sets of signals and conditions with a small number of

A. S. Epifanov, PhD student

Automata interpretation of gene sequences

Use of a new way of the task of laws of functioning of automaton (finite state machine) in the form of symbolical and numerical schedules (geometrical images of laws of functioning of automatic devices) allows to spend the analysis of properties of objects and the processes presented by mathematical structures in forms of geometrical curves and sequences, with their corresponding interpretations. The special spectrum of numerical parameters of recurrent definition of sequences allows to state deep enough numerical estimation to a structure of sequences, and on this basis to receive estimations of complexity of the state machine, generating curves and sequences. Properties of natural objects and the processes presented by geometrical curves and sequences, suppose an estimation of complexity on numerical parameters of a spectrum. Article contains results of the lead analysis of fragments of sequences of DNA of the human and other primates, and also sequences RNA of various viruses. The sequence of DNA (RNA) is considered as sequence of the second coordinates of points of a geometrical image of the state machine which at chosen set X of input signals and the fixed order on elements of set of input words X^* unequivocally defines laws of functioning of state machine. In clause is carried out construction of state machines by fragments of DNA, also is investigated dependence of number of conditions of state machine on number of input signals. On the basis of construction and the analysis of spectrum of numerical parameters for sequences of DNA(RNA), considered in the clause, are constructed classes of equivalent sequences on complexity.

Key Words: geometrical image of automaton, gene sequences

elements. It was reflected in ways of the task of state machines based on the obvious instruction of function (tables, finite graphs, matrixes, the logic equations with the variables set on finite sets). Functioning of state machines is based on recursion,

which allows to represent somehow big, but only initial, a fragment of process of functioning.

V.A.Tverdohlebov in work [1] is offered and developed the new approach for the task of laws of functioning of the discrete determined dynamic systems, based on numerical structures. The given approach allows to use powerful idealization of classical continuous mathematics: infinitesimal size, actual infinity, summation of infinite numbers, limiting transition etc. The Geometrical image γ_s defines completely laws of functioning of state machine A_s , that is, all phase picture of communications of input sequences with output signals. Concrete variants of processes of functioning, that is, phase trajectories, have geometrical images $\gamma_s(p)$, $p \in X^*$, in the form of sections γ_s on separate points. Geometrical images $\gamma_s(p)$ can be set numerical, instead of symbolical, the equations. Use of the apparatus of geometrical images allows to consider with automata interpretation geometrical curves on planes and numerical sequences. In work [1] are developed methods of synthesis of state machines on sequences and geometrical curves. The sequence of elements from the finite set, combined with linear order on set of input words, defines laws of functioning of the discrete determined dynamic system (state machine). In [3-4] contains results of the analysis more than 100000 sequences with various interpretation, including 8594 numerical sequences, setting approach of fundamental mathematical constants (π , e , φ , $\sqrt{2}$, etc.). In given clause are presented results on construction and analysis of state machines which laws of functioning are certain by initial pieces of geometrical images and a choice of number of entrance signals of the automatic device contain. For this purpose from bank of sequences of DNA [5] initial fragments of DNA of the people, a chimpanzee, a dolphin, a dog, a bat, the elephant, etc. length 100000 are taken. On each sequence laws of functioning of state machine are synthesized at various values of capacity of the set of input signals ($|X| = 2, 5, 10, 20, 50, 100, 1000, 2000, 5000, 10000$), state machines are minimized on number of conditions and the constructed class of state machines is broken into subclasses of equivalent automatic devices on complexity on number of conditions in the minimal state machines. Check of equivalence of conditions of state machines is carried out on the basis of use of the device of geometrical images of state machines. Classical methods of check of equivalence of conditions of the state machines (see, for example, A.Gill, [2]) at the specified dimensions of set of input signals and

conditions of the state machines practically cannot be realized (for example, at the analysis of sequence of length 100000 in case of $|X| = 100$ number of conditions of the automatic device equally 1000 and for check of equivalence of conditions is necessary to calculate reaction of the automatic device to 100^{999} input sequences of length 100).

Besides in the given work are presented results of construction and the analysis of spectrum of dynamic parameters (see [1]) for sequences taken from bank [5].

1. Geometrical images of laws of functioning of state machines

In work [1] Tverdohlebov V.A. is offered the formal mathematical apparatus of replacement of symbolical automaton models in the form of tables, graphs, the logic equations, by numerical structures in the form of geometrical figures, the numerical equations and sequences. This approach is intended for search of new ideas and methods of the organization of technical diagnosing of complex systems. Transformation of the symbolical form of automata model to numerical structure (a geometrical image of laws of functioning of the state machine) includes linear ordering automaton mapping $\rho_s = \bigcup_{p \in X^*} \{(p, \lambda(s, p))\}$, for initial state

machine $A_s = (S, X, Y, \delta, \lambda, s)$, where S , X and Y – accordingly sets of conditions, input and output signals, and $\delta: S \times X \rightarrow S$ – next-state function, $\lambda: S \times X \rightarrow Y$ – function of outputs. Automaton mapping ρ_s one-to-one transformed to automata mapping of a kind $\rho'_s = \bigcup_{p \in X^*} \{(p, \lambda'(s, p))\}$,

where $\lambda'(s, p)$ – last sign on sequence $\lambda(s, p)$. From a geometrical image γ_s of state machine A_s separate the sequence of the second coordinates of points of a geometrical image which one-to-one corresponds to a full geometrical image (at fixed the order on set X^* and size $m = |X|$). As a result laws of functioning of the state machine (that is, a phase picture) and concrete processes of functioning of the state machine (that is, phase trajectories) appear one-to-one certain by sequence of the second coordinates of points of a geometrical image. It allows to consider any sequence of elements from finite set as sequence of the second coordinates of points of a geometrical image and, hence, as the task of laws of functioning of the state machine. For that let a set of conditions of automaton a subset of set $\{s_p\}_{p \in X^*}$ and next-state function is defined by equality: $\delta(s_p, x) = s_{px}$

for any $p \in X^*$ and $x \in X$. Everyone point (px, y) , available in a geometrical image of state machine, where $p \in X^*$, $x \in X$ and $y \in Y$, defines equality $\lambda(s_p, x) = y$.

Overlapping of laws of functioning δ and λ of the automaton $A = (\{s_p\}_{p \in X^*}, X, Y, \delta, \lambda, s_e)$ with geometrical figures (curves on a plane) can be carried out on the basis of: a choice of points on a geometrical figure, a choice of a direction of detour of these points and construction of set Y of the second coordinates for the chosen points; constructions of sequence $y_1, y_2, \dots, y_i, \dots$ corresponding detour of the points chosen on a geometrical figure; a choice of the whole positive number m , sets $X = \{x_1, x_2, \dots, x_m\}$ and the initial linear order $x_1 \prec x_2 \prec \dots \prec x_m$ for definition of the linear order ω_1 on X^* ; construction of functions and with use of the rules of construction of functions specified above on points of a geometrical image. Use of the described geometrical approach allows to represent some properties of laws of functioning of the discrete determined dynamic systems (state machines) by properties of sequences or geometrical curves.

2. Spectrum of dynamic parameters of recurrent definition of sequences

The spectrum of dynamic parameters of the sequence is offered and developed for strict representation of properties of sequence in work [1] by Tverdohlebov V.A., describing sequence on interrelations (interposition) of elements in it. The spectrum of parameters represents rules of construction of sequence with use of recurrent forms of various orders. For this purpose from final set U recurrent forms $F(z_1, z_2, \dots, z_m) = z_{m+1}$ at which $z_i, 1 \leq i \leq m+1$, accept values from set U and which define parts or at once all sequence are compared with considered sequence $\xi = \langle u(1), u(2), \dots, u(t), \dots \rangle$ of elements.

Definition of sequence by recurrent form F is realized by sequence of recurrent forms on the basis of overlapping variables of the recurrent form with elements of sequence by a rule: for any $t, t > m$ (or t belongs to a considered interval of the whole positive numbers) $F(u(t-m), u(t-m+1), \dots, u(t-1)) = u(t)$.

Under the assumption independent and dependent variables of recurrent forms are certain on finite set U , that is, to recurrent form F there corresponds finite display of kind $F: U^m \rightarrow U$. It allows to set

effectively recurrent forms, finite families of recurrent forms and rules of their application at definition of sequences.

Spectrum $\Omega(\xi)$ for sequence ξ has 5 levels: $\Omega(\xi) = (\Omega_0(\xi), \Omega_1(\xi), \Omega_2(\xi), \Omega_3(\xi), \Omega_4(\xi))$. In a spectrum numerical values present orders of recurrent forms, lengths of pieces the sequences defined by separate recurrent forms and quantity of changes of recurrent forms. Each concrete variant of realization (representation by values of parameters) of any level $\Omega_i(\xi)$ defines splitting set U^* into subsets on properties of concurrence of characteristics. Subsets of such splitting we shall consider as classes of equivalence of sequences. Below is resulted formal definition of a spectrum $\Omega(\xi)$.

Let $U = \{u_1, u_2, \dots, u_k\}$ - final set and ξ Sequence of elements from set U : $\xi = \langle u(1), u(2), \dots, u(t), \dots \rangle$. We shall enter following designations.

For any sequence $\bar{\xi} \in U^v$, where U^v - set of all sequences of length v from elements from set U , the least order of the recurrent form defining sequence $\bar{\xi}$, we shall designate $m_0(\bar{\xi})$. For any sequence $\bar{\xi} \in U^v$ and $m \in N^+$, where $1 \leq m \leq m_0(\bar{\xi})$, the greatest length of an initial piece of the sequence $\bar{\xi}$, defined by the recurrent form of the order m , we shall designate $d^m(\bar{\xi})$. For any sequence $\bar{\xi} \in U^v$ and $m \in N^+$, where $1 \leq m \leq |\bar{\xi}| - 1$, number of changes of recurrent forms of the order m , required at definition of sequence $\bar{\xi}$, we shall designate $r^m(\bar{\xi})$. For any sequence $\bar{\xi} \in U^v$ and $m \in N^+$, where $1 \leq m \leq m_0(\bar{\xi})$ and j , where $1 \leq j \leq r^m(\bar{\xi})$ length of j -th piece in definition of sequence $\bar{\xi}$ we shall designate $d_j^m(\bar{\xi})$.

Using the entered designations we shall define a spectrum of the parameters describing sequence, as following structure:

$$\begin{aligned} - \Omega_0(\bar{\xi}) &= \langle m_0(\bar{\xi}) \rangle; \\ - \Omega_1(\bar{\xi}) &= \langle d^1(\bar{\xi}), d^2(\bar{\xi}), \dots, d^{\alpha}(\bar{\xi}) \rangle; \\ - \Omega_2(\bar{\xi}) &= \langle r^1(\bar{\xi}), r^2(\bar{\xi}), \dots, r^{\alpha}(\bar{\xi}) \rangle; \\ - \Omega_3(\bar{\xi}) &= \langle \Omega_3^1(\bar{\xi}), \Omega_3^2(\bar{\xi}), \dots, \Omega_3^{\alpha}(\bar{\xi}) \rangle, \end{aligned} \quad \text{where}$$

$\alpha = m_0(\bar{\xi})$ and $\Omega_3^j(\bar{\xi}) = \langle d_1^j(\bar{\xi}), d_2^j(\bar{\xi}), \dots, d_{n_j}^j(\bar{\xi}) \rangle$ (n_j - number of last piece in definition of sequence $\bar{\xi}$ as sequences of the pieces defined by separate recurrent forms of the order j). The fourth level $\Omega_4(\bar{\xi})$ a

spectrum $\Omega(\bar{\xi})$ adds to characteristics in previous levels an estimation of complexity of rules and variants of use of rules. Formally $\Omega_4(\bar{\xi}) = \Theta(\Omega_3(\bar{\xi}))$, where Θ - the operator of replacement in $\Omega_3(\bar{\xi})$ sizes of lengths of pieces in weights of the used recurrent forms for definition of pieces.

3. Synthesis of laws of functioning of automaton on sequence of DNA

In work [1] the new type of the automaton - $(H, m, d(H))$ - automaton is offered. Laws of functioning of the given type of the automaton are set by numerical sequence H which is necessary sequence of the second coordinates of points of a geometrical image. The initial piece of length $d(H)$ sequences H is considered. The size m - number of input signals of the automaton, quantity of output signals is defined by specificity of an initial piece of sequence H of length $d(H)$ (number of various values of elements in an initial piece of length $d(H)$).

In the given work is made construction and the analysis of classes of automaton, which laws of functioning are certain by sequence of DNA(RNA). From bank of sequences of DNA [5] initial fragments of length of 100000 signs on DNA of the people, a gorillas, a chimpanzee, a hedgehogs, cows, a dogs, mouse, hen, a dolphin and the elephant are taken. It is known, that the alphabet of genetic records consists of 4 letters A, C, G, T, which designate accordingly four kinds of the nitrogenous bases (adenine, cytosine, guanine and thymine), representing specificity of sequence of DNA. The taken sequences are coded on the basis of replacement of a sign adenine by signal y_1 , a sign cytosine by signal y_2 , guanine - by signal y_3 and thymine - by signal y_4 . Accordingly all automaton, constructed in this work, have output set $Y = \{y_1, y_2, y_3, y_4\}$.

At synthesis of laws of functioning of automaton on sequences of DNA corresponding sequences of the first coordinates of points of a geometrical image were defined by variants of a choice of number of input signals of the automaton and linear order ω , on set X^* of input words. In work is used one kind of the linear order, the so-called modified lexicographic order (see the monography [1]). At synthesis of laws of functioning of the automaton

the way of regularization of next-state functions is essential. Probably cyclic regularization, regularization in an initial state (reset), generation of a state by random numbers generator (from set of possible sets). In case of, when $\frac{d(H)}{|X|} \neq \left\lceil \frac{d(H)}{|X|} \right\rceil$ where

$|X|$ - capacity of the set of input signals of the automatic device, and $d(H)$ - the length of an initial piece of sequence H (on which laws of functioning of the automaton are under construction), regularization is required and for function of outputs. In the given work regularization of next-state functions is carried out by all specified ways, and value of capacity of the set of input signals and lengths of initial pieces of sequences are chosen in such a manner that $\frac{d(H)}{|X|} = \left\lceil \frac{d(H)}{|X|} \right\rceil$, therefore

regularization it is not required to function of outputs. On each sequence (digital) are constructed 300 automaton (at consideration of a set of the initial pieces having lengths 10000, 20000, ..., 100000 signs, use of 3 ways of regularization of next-state functions and 10 various values of capacity of the set of input signals $|X| = 2, 5, 10, 20, 50, 100, 1000, 2000, 5000, 10000$). In view of the big dimension of the constructed automaton as an illustration used method of synthesis of the automaton on sequence are resulted in table 1 and 2 laws of functioning of the automaton (the table of next-state function and table of function of outputs), constructed on a piece of DNA of length of 100 signs (taken of chromosome Y of the man) at $|X| = 20$.

As a result of the analysis of properties of the constructed automaton are noted following specific properties. Dependence of number of states of the automaton after minimization from a way of regularization of next-state functions is noted. So at cyclic of regularization of next-state functions it is noted, that at all constructed automaton (on all analyzed DNA) the number of classes of equivalence practically coincides with number of states, i.e. almost all such automaton already are minimal on number of states. Reduction of number of conditions after minimization is noted at of regularization of next-state functions in an initial state, and at and of regularization of next-state functions on the basis of use of the generator of random numbers (states gets out of set of possible states) the number of classes of equivalence

coincides with number of states, i.e. automatic devices are minimal on number of states.

regularization and of regularization with random number generator.

Table 1.

The table of next-state function of automaton (at cyclic regularization of next-state function), constructed by a fragment of sequence of DNA of length of 100 signs.

δ	s_0	s_1	s_2	s_3	s_4
x_1	s_1	s_1	s_1	s_1	s_1
x_2	s_2	s_2	s_2	s_2	s_2
x_3	s_3	s_3	s_3	s_3	s_3
x_4	s_4	s_4	s_4	s_4	s_4
x_5	s_0	s_0	s_0	s_0	s_0
x_6	s_1	s_1	s_1	s_1	s_1
x_7	s_2	s_2	s_2	s_2	s_2
x_8	s_3	s_3	s_3	s_3	s_3
x_9	s_4	s_4	s_4	s_4	s_4
x_{10}	s_0	s_0	s_0	s_0	s_0
x_{11}	s_1	s_1	s_1	s_1	s_1
x_{12}	s_2	s_2	s_2	s_2	s_2
x_{13}	s_3	s_3	s_3	s_3	s_3
x_{14}	s_4	s_4	s_4	s_4	s_4
x_{15}	s_0	s_0	s_0	s_0	s_0
x_{16}	s_1	s_1	s_1	s_1	s_1
x_{17}	s_2	s_2	s_2	s_2	s_2
x_{18}	s_3	s_3	s_3	s_3	s_3
x_{19}	s_4	s_4	s_4	s_4	s_4
x_{20}	s_0	s_0	s_0	s_0	s_0

Table 2.

The table of function of outputs of automaton (at $|X|=20$), constructed by a fragment of sequence of DNA of length of 100 signs.

λ	s_0	s_1	s_2	s_3	s_4
x_1	y_2	y_1	y_2	y_3	y_2
x_2	y_1	y_1	y_1	y_3	y_3
x_3	y_2	y_4	y_2	y_3	y_1
x_4	y_2	y_1	y_4	y_3	y_2
x_5	y_2	y_2	y_2	y_3	y_1
x_6	y_4	y_4	y_1	y_4	y_4
x_7	y_3	y_1	y_2	y_4	y_2
x_8	y_1	y_4	y_3	y_1	y_2
x_9	y_2	y_3	y_4	y_3	y_3
x_{10}	y_1	y_3	y_4	y_3	y_1
x_{11}	y_3	y_1	y_2	y_3	y_3
x_{12}	y_2	y_3	y_4	y_2	y_1
x_{13}	y_2	y_3	y_3	y_4	y_2
x_{14}	y_4	y_2	y_1	y_4	y_4
x_{15}	y_2	y_2	y_3	y_2	y_3
x_{16}	y_3	y_1	y_3	y_1	y_3
x_{17}	y_4	y_1	y_2	y_1	y_3
x_{18}	y_4	y_1	y_1	y_2	y_1
x_{19}	y_2	y_4	y_3	y_1	y_3
x_{20}	y_4	y_1	y_4	y_4	y_1

The increase in a difference between number of states at the automaton before minimization at regularization of next-state functions in an initial state in comparison with cyclic regularization of next-state functions observed in a class of the automata constructed by a fragment of sequence of DNA of the mouse and a hedgehog.

The analysis of the constructed classes of automata has shown, that use at construction of automata regularization of next-state functions in an initial state gives function of transitions the least number of states after minimization, than cyclic of

4. Construction and the analysis of spectrums for fragments of sequences of DNA

Base model of the discrete determined dynamic systems are the finite determined automata. Comparison on complexity of such automata can be made on the basis of comparison of the mathematical structures representing specificity of laws of functioning of the automaton. As such structures geometrical images of laws of functioning of automata can be used. As it has been shown above, specificity of a geometrical image at the fixed set of input signals of automaton X and the linear

order ω_1 on set X^* of input words, is presented by sequence of the second coordinates of its points. As such sequences in the given work are considered sequences of DNA of various alive essences. For an estimation of complexity of sequences is used the spectrum of dynamic parameters offered and developed by V.A.Tverdohlebov in work [1]. The lead computing experiment has included construction of spectrums for more than 5000 fragments of sequences of DNA of various length (from 500 up to 100000 signs). As a result of the analysis of the received results is noted specificity of analyzed DNA from the point of view of the used mathematical apparatus. For example, the initial piece of length 10000 fragments of DNA of the man corresponding gene BRCA2 (see [5]) is described without contradictions by the recurrent form of the order 15 (i.e. $m_0=15$). Thus, use of recurrent forms of orders 3 and 4 allows to describe without contradictions the same initial piece of length 23, use of recurrent forms of orders 10 and 11 - the same initial piece of length 765, and with orders 13 and 14 the piece of length 6191 is defined. From the point of view of the recurrent description the fragment of DNA of the rabbit of the length 10000, corresponding gene SLC35F1 has specific properties. The length of the initial piece described without contradictions by recurrent forms of orders $m=8,9,10,11,12$, has the same value - 48 signs. With orders of the recurrent form $m=13, 14, 15, 16, 17$ the initial piece of identical length - 73 signs also is described. The length of the initial piece described without contradictions by recurrent forms of orders

$m=18,19,20,21, \dots, 39$, matters - 1914 signs. Such specific change of length of an initial piece at increase in the order of the recurrent form is noted on all analyzed interval up to 10000 signs which without the contradiction is defined by the recurrent form of the order exceeding almost in 100 times similar size for definition of person BRCA2 specified above a gene. Unfortunately, the volume of the given work does not allow to present detailed results of research with use of a spectrum even one chromosome of any of the alive essences chosen for the analysis.

Brief conclusions

The results stated in clause show an opportunity of use of the apparatus of geometrical images for the analysis of properties of sequences of DNA. Automatic interpretation of sequences of DNA allows to reveal new, deeper and thin properties of DNA. The classes of the discrete determined dynamic systems (automatons) certain on the basis of a choice of sequence of DNA and number of input signals of automatons are constructed and analyzed. It is analyzed more than 3000 automatons, classes of equivalent automatons on complexity are constructed. For an initial fragment of each sequence of DNA, analyzed in work, values of parameters of a spectrum are calculated and on the basis of concurrence of values of parameters of a spectrum is lead classification of sequences of DNA.

References

1. Tverdokhlebov V.A. The geometrical images of laws of functioning of finite state machines, "Science book", Saratov, 2008.-183 p.
2. Gill A., Introduction in the theory of finite state machines, Publishing house the Science, Moscow, 1966.
3. Epifanov A.S. The analysis of phase pictures of discrete dynamic systems, "Science book", Saratov, 2008.-156p.
4. Epifanov A.S. "The analysis of geometrical images of laws of functioning of finite state machines", Upravlenie bol'simi sistemami, Vol.24, Moscow, 2009.pp.81-98.
5. <http://www.ensembl.org/index.html>

Надійшла до редколегії 22.12.10

УДК 004.4

Іванов Є.В., аспірант

Операційна семантика програм обробки складноіменних даних

У статті надано операційну семантику програмам композиційно-номінативної мови обробки складноіменних даних.

Ключові слова: композиційно-номінативний підхід, номінативні дані, операційна семантика.

*E-mail: ivanov.eugen@gmail.com

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

1. Вступ

Теорія програмування, як наука, знаходиться у стадії становлення, у зв'язку з чим багато її понять не мають адекватних математичних уточнень, або існуючі уточнення є неповними. Тому у деяких випадках важко формулювати у точних термінах проблеми, важливі для неї. Один з підходів, що може бути застосований для подолання вказаних труднощів є композиційно-номінативний підхід до уточнення понять програмування [1-3]. Його метою є розробка ієрархії адекватних моделей програм різного рівня абстракції і загальності. Згідно цього підходу, дані уточнюються як класи номінативних даних, програми – як функції над номінативними даними (номінативні функції), а засоби конструювання програм – як композиції програм (оператори над класами номінативних функцій). У даній роботі розглядається специфічна проблема, пов'язана із композиційно-номінативним підходом. У роботах [5-7] у рамках композиційно-номінативного підходу розглянуто проблему стабільності програм при структурних перетвореннях даних, побудовано спеціальний клас номінативних даних зі складними іменами – клас складноіменних даних, а також побудовано відповідну функціонально повну і структурно адекватну формальну мову програм над такими даними – мову $SICON_{as}$ (просту композиційно-номінативну мову програм з асоціативним розіменуванням).

У роботах [5-7] програми розглядалися як функції над даними і фактично було надано денотативну семантику програм обробки складноіменних даних. В той же час для реалізації обробки складноіменних даних корисним є корисною наявністю визначеної

Іє. V. Ivanov, PhD student

Operational semantics of programs over complex-named data

In the article an operational semantics of programs of a composition-nominative language for complex-named data processing is given.

Key Words: composition-nominative approach, nominative data, operational semantics.

операційної семантики цих програм. У зв'язку з цим у даній роботі надається операційна семантика програм мови $SICON_{as}$.

2. Складноіменні дані і мова $SICON_{as}$

У цьому розділі надамо визначення класу складноіменних даних і композиційно-номінативної мови $SICON_{as}$ програм обробки таких даних.

Клас складноіменних даних [5] побудований на основі класу номінативних даних зі складними іменами, тобто даних, імена у яких є рядками у певному алфавіті.

Нагадаємо [1-3], що номінативні дані є ієрархічно побудованими об'єктами, в основі яких лежить відношення ім'я $\mapsto$ значення. Такі дані можна подавати текстовим записом, подібним до наведеного нижче (де v_1, v_2, v_3 – імена): $[v_1 \mapsto 1, v_2 \mapsto [v_1 \mapsto 2, v_3 \mapsto 3]]$.

Формально номінативні дані визначаються індуктивно на основі множини імен V та класу базових значень (атомів) W таким чином:

$ND_0(V, W) = W \cup \{\emptyset\}$ – номінативні дані рангу нуль,

$$ND_{k+1}(V, W) = ND_k(V, W) \cup (V \xrightarrow{*} ND_k(V, W))$$

для $k \geq 0$ – дані рангу не більше $k+1$, де $A \xrightarrow{*} B$ позначає клас (часткових) функцій зі скінченним графіком.

Тоді усі номінативні дані складають клас $ND(V, W) = \bigcup_{k \geq 0} ND_k(V, W)$.

Основними операціями над номінативними даними є унарні параметричні операції іменування і розіменування, і бінарна операція накладання.

Позначимо літерою d номінативне дане.

Операція іменування $\Rightarrow v$ з параметром $v \in V$ визначається рівністю $\Rightarrow v(d) = [v \mapsto d]$.

Операція розіменування $v \Rightarrow$ з параметром $v \in V$ визначається умовою $v \Rightarrow (d) \equiv d(v)$, де $\equiv$ – сильна рівність [1,5].

Накладання ∇ визначене тільки на неатомарних даних і задається рівністю $d_1 \nabla d_2 = d_2 \cup d_1|_{\arg(d_1) \setminus \arg(d_2)}$, де $\arg(d)$ – множина імен (аргументів), на яких визначено d . Ця операція створює дане, в якому присутні усі компоненти другого аргументу і ті компоненти першого, які не були перевизначені в другому.

Номінативні дані класу $ND(V, W)$ засновані на неструктурованих (простих) іменах. Якщо допустити, що імена мають структуру у формі конкатенації простих імен, то отримаємо клас даних зі складними іменами, що позначається як $NDV(V, W) = ND(V^*, W)$, де V^* – множина не порожніх слів в алфавіті V . У даній роботі ми будемо розглядати випадок, коли множина базових імен V скінченна.

Неформально, клас складноіменних даних $NDVC(V, W)$ є підкласом даних з $NDV(V, W)$, для всіх елементів якого виконується принцип однозначності структурного іменування [5-7], який означає, що одному складному імені у даному відповідає одне значення. Наприклад, у даному $[v_1 v_2 \mapsto 1, v_1 \mapsto [v_2 \mapsto 2]]$ складному імені $v_1 v_2$ відповідає два різних значення (1 і 2), тому для нього ОСІ не виконується, а для даного $[v_1 v_2 \mapsto 1, v_1 \mapsto [v_1 \mapsto 2]]$ ОСІ виконується.

Перш ніж надати формальне визначення $NDVC(V, W)$, введемо поняття шляху в даному $d \in NDV(V, W)$: іменний шлях в d – це послідовність імен $(v_1, v_2, \dots, v_k)$ з V^* ($k \geq 1$), така, що визначено значення $(\dots((d(v_1))(v_2))\dots(v_k))$.

Формальне визначення класу складноіменних даних подається таким чином: $NDVC(V, W) = W \cup G(V, W)$, де $G(V, W)$ – дані $d \in NDV(V, W) \setminus W$ такі, що має місце властивість: якщо для двох деяких шляхів $(v_1, v_2, \dots, v_n)$ та $(v'_1, v'_2, \dots, v'_m)$ в d , слово $v_1 v_2 \dots v_n$ є префіксом слова $v'_1 v'_2 \dots v'_m$, то $n \leq m$ і $v_i = v'_i$ для всіх $i \in \{1, 2, \dots, n\}$. Змістовно це означає, що у даному немає двох шляхів (один з яких не є початком іншого), таких, що конкатенація імен вздовж одного шляху є префіксом конкатенації імен вздовж іншого шляху.

Домовимося, що надалі літерою d (можливо з індексами) будуть позначатися дані класу $NDVC(V, W)$, якщо явно не вказано інакше.

Основними операціями над складноіменними даними є операція іменування, визначена вище для довільних номінативних даних, унарна параметрична операція асоціативного розіменування і бінарна операція структурного накладання. Асоціативне розіменування $v \Rightarrow_a$ з параметром $v \in V^*$ визначається індуктивно за довжиною слова-параметру. Введемо позначення $d(v) \downarrow$ – дане d визначено на імені v .

Якщо $|v|=1$ (де $|v|$ – довжина слова), то $v \Rightarrow_a(d) = d(v)$ у випадку, коли $d(v) \downarrow$, $v \Rightarrow_a(d) = [u \mapsto d(vu) | u \in V^*, d(vu) \downarrow]$, якщо існує слово $u \in V^*$, таке, що $d(vu) \downarrow$, і $v \Rightarrow_a(d) \uparrow$ в інших випадках.

Якщо $|v| > 1$, то $v \Rightarrow_a(d) \equiv v_2 \Rightarrow_a(v_1 \Rightarrow_a(d))$, де $v_1 \in V$ і $v_2 \in V^*$ – такі слова, що $v = v_1 v_2$.

Структурне накладання [5] аналогічне накладанню, введеному вище, але враховує складеність імен. На прикладах ця операція діє таким чином:

$$1) [v_1 \mapsto d_1] \nabla [v_2 \mapsto d_2] = [v_1 \mapsto d_1, v_2 \mapsto d_2],$$

якщо $v_1, v_2 \in V$, $v_1 \neq v_2$;

2) $[v_1 v_2 \mapsto d_1] \nabla [v_1 \mapsto d_2] = [v_1 \mapsto d_2]$, значення під іменем v_1 перевизначає значення під іменами, що є продовженнями v_1 ;

3) $[v_1 \mapsto d_1] \nabla [v_1 v_2 \mapsto d_2] = [v_1 \mapsto d_1 \nabla [v_2 \mapsto d_2]]$ (при $d_1 \notin W$), тобто значення під іменем $v_1 v_2$ модифікує значення під іменем v_1 .

Розглянемо мову програм $SICON_w$ (проста композиційно-номінативна мова із асоціативним розіменуванням). Кожна програма в цій мові трактується як часткова функція класу $NDVC(V, W) \rightarrow NDVC(V, W)$, що може бути отримана з базових функцій за допомогою композицій.

Базовими функціями є сім'я операцій іменування $\Rightarrow v$ (для $v \in V^*$), сім'я операцій асоціативного розіменування $v \Rightarrow_a$ (для $v \in V^*$) і функція-константа $\emptyset$ зі значенням $\emptyset$ (порожнє, тобто ніде не визначене дане).

Композиціями мови є найпростіші композиції структурного програмування – множення (послідовне виконання), цикл і присвоєння.

Множення – це бінарна композиція, яка за функціями f, g будує функцію h (позначається $h = f * g$), таку, що $h(d) \equiv g(f(d))$.

Цикл – це бінарна композиція, яка за функціями p (умова) і g (тіло циклу) будує функцію h (позначається $h = p * g$), таку, що $h(d) = d$, якщо $p(d) \downarrow = \emptyset$, $h(d) = g^{(k)}(d)$, якщо $p(g^{(k)}(d)) \downarrow = \emptyset$ і $p(g^{(i)}(d)) \downarrow \neq \emptyset$ для всіх $i \in \{1, 2, \dots, k-1\}$ (тут $g^{(k)}(d) = g(g(\dots g(d)\dots))$ – k -кратне застосування функції g до даного d), $h(d)$ не визначено, в інших випадках.

Присвоєння Asg^v , залежне від параметру $v \in V^*$ – унарна композиція, що задається формулою $\text{Asg}^v(f)(d) \equiv d \nabla_v [v \mapsto f(d)]$.

Незважаючи на простоту, мова SICON_{as} є достатньо потужною: в ній можна промодельовувати усі частково рекурсивні функції (тобто ця мова є функціонально повною) і можна промодельовувати похідними композиціями багато композицій, що використовуються у програмуванні – різноманітні умовні оператори, цикли, тощо.

3. Операційна семантика програм мови SICON_{as}

Позначимо Prog – множина термів програм мови SICON_{as} , тобто термів, побудованих з символів базових функцій за допомогою символів композицій.

Визначимо структурну операційну семантику з великими кроками (натуральна семантика або англ. big-step operational semantics) [4] програм мови SICON_{as} . Спочатку ми визначимо таку семантику вважаючи операції над складноіменними даними елементарними, а потім деталізуємо її таким чином, щоб вона відображала складність цих операцій.

Визначимо відношення $\Downarrow$ на $(\text{Prog} \times \text{NDVC}(V, W)) \times \text{NDVC}(V, W)$ переходу із конфігурації (пари програма, дане) у заключне дане. При цьому $\langle P, d \rangle \Downarrow d'$ означає, що програма P починаючи роботу на d , завершується із результатом d' .

Відношення $\Downarrow$ визначимо як найменше за включенням відношення, що задовольняє правилам, наведеним нижче:

1) Множення: якщо $\langle P_1, d_1 \rangle \Downarrow d_2$, то $\langle P_1 \square P_2, d_1 \rangle \Downarrow \langle P_2, d_2 \rangle$.

2) Присвоєння: якщо $d_1 \notin W$, $\langle P, d_1 \rangle \Downarrow d_2$, $d_1 \nabla_v [v \mapsto d_2] = d_3$, то $\langle \text{Asg}^v(P), d_1 \rangle \Downarrow d_3$.

3) Цикл:

якщо $\langle P_1, d_1 \rangle \Downarrow \emptyset$, то $\langle P_1 * P_2, d_1 \rangle \Downarrow d_1$.

якщо $\langle P_1, d_1 \rangle \Downarrow d$, $d \neq \emptyset$, то $\langle P_1 * P_2, d_1 \rangle \Downarrow \langle P_2 \square P_1 * P_2, d_1 \rangle$.

4) Порожня константа: $\langle \emptyset, d \rangle \Downarrow \emptyset$.

5) Іменування: $\langle \Rightarrow v, d \rangle \Downarrow [v \mapsto d]$.

6) Розіменування: якщо $v \Rightarrow_o (d_1) \downarrow = d_2$, то $\langle v \Rightarrow_o, d_1 \rangle \Downarrow d_2$.

Коректність наведеної операційної семантики:

Теорема 1. $\langle P, d \rangle \Downarrow d'$ тоді і тільки тоді, коли $f(d) \downarrow = d'$, де $f: \text{NDVC}(V, W) \rightarrow \text{NDVC}(V, W)$ – інтерпретація терму P .

Доведення проводиться індукцією за структурою терму P .

У наведеній операційній семантиці не розкрито виконання операцій над складноіменними даними. Враховуючи, що складноіменні дані мають структуровану будову, такий погляд не завжди є доцільним. Тому ми деталізуємо семантику, щоб вона відображала складність операцій над даними.

Для визначення деталізованої операційної семантики використаємо подання складноіменних даних, побудоване у роботі [7]. Це подання конструктивне в тому сенсі, що дане подається термом у (скінченній) сигнатурі, а не розглядається як часткова функція.

Будемо вважати, що множина базових імен V скінченна і містить n елементів – $v_1, v_2, \dots, v_n$.

Нехай $C = \{e\} \cup \{\psi_{i_1, i_2, \dots, i_n} \mid i_1, i_2, \dots, i_n \in \{0, 1, 2\}\}$ – сигнатура, де e – 0-арний функціональний символ, $\psi_{i_1, i_2, \dots, i_n}$ – функціональний символ арності $\text{sgn } i_1 + \dots + \text{sgn } i_n$. Розширимо сигнатуру C базовими значеннями: покладемо $C_* = C \cup \{\omega\} \cup W$, де ω – новий унарний символ, елементи з W розглядаються як 0-арні.

Тоді покладемо D – множина усіх термів t у сигнатурі C_* , таких, що $t \in W$ і якщо $\omega(t')$ – підтерм t , то $t' \in W$ (тобто аргументом ω може бути тільки базове значення). Символ ω змістовно відмічає базові значення у термі.

У [7] побудована бієкція $\gamma: \text{NDVC}(V, W) \rightarrow D$, яка, неформально кажучи, є легко обчислюваною. Терми з D будемо розглядати як представлення складноіменних даних. У [7] побудовані системи переписування термів для

операцій іменування і асоціативного розіменування і структурного накладання над таким поданням. Кожна з цих систем переписування термів (СПТ) є конструктор-ортогональною [7] і має вигляд $R = (F_{def}, C_1, P)$, де F_{def} – сигнатура нових функціональних символів, які визначає R , $C_1 \subseteq C \cup \{\omega\}$ – сигнатура конструкторів для визначення символів з F_{def} , P – множина правил переписування. СПТ R відповідає відношення переписування термів сигнатури $F_{def} \cup C_\omega$ за один крок $\rightarrow_1$ і його рефлексивно-транзитивне замикання $\rightarrow_*$.

Для кожного $v \in V^*$ покладемо $R_{\Rightarrow v}$ – СПТ, побудована у [7] для іменування ($R_{\Rightarrow v}$ може бути ефективно побудована за v), в якій визначається новий функціональний символ $\Rightarrow \bar{v}$. Позначимо $\rightarrow_{\Rightarrow v}$ – відповідне відношення переписування. Тоді $\Rightarrow \bar{v}(D_1) \rightarrow_{\Rightarrow v} D_2$ тоді і тільки тоді, коли $\Rightarrow v(\gamma^{-1}(D_1)) = \gamma^{-1}(D_2)$ (СПТ моделює іменування).

Аналогічно покладемо $R_{\Rightarrow}$ і $R_{\bar{v}}$ – СПТ, побудовані у [7] для асоціативного розіменування і структурного накладання, що визначають відповідно функціональні символи $\bar{v} \Rightarrow$ і $\bar{v}$ (останній бінарний).

Визначимо деталізовану операційну семантику $SICON_{as}$ на основі вказаних систем переписування термів:

1) Множення: якщо $\langle P_1, D_1 \rangle \Downarrow D_2$, то $\langle P_1 \sqcap P_2, D_1 \rangle \Downarrow \langle P_2, D_2 \rangle$.

2) Присвоєння: якщо $D_1 = e$ або $D_1 = \psi_{i_1, \dots, i_n}(t_1, \dots, t_n)$ для деяких індексів $i_1, \dots, i_n$ і термів $t_1, \dots, t_n$, $\langle P, D_1 \rangle \Downarrow D_2$, $\langle \Rightarrow v, D_2 \rangle \Downarrow D_3$, $D_1 \bar{v}, D_3 \rightarrow_v D_4$, то $\langle Asg^*(P_1), D_1 \rangle \Downarrow D_4$.

3) Цикл: якщо $\langle P_1, D_1 \rangle \Downarrow e$, то $\langle P_1 * P_2, D_1 \rangle \Downarrow D_1$, якщо $\langle P_1, D_1 \rangle \Downarrow D$, $D \neq e$, то $\langle P_1 * P_2, D_1 \rangle \Downarrow \langle P_2 \sqcap P_1 * P_2, D_1 \rangle$.

4) Порожня константа: $\langle \bar{\emptyset}, D \rangle \Downarrow e$.

5) Іменування: якщо $\Rightarrow \bar{v}(D_1) \rightarrow_{\Rightarrow v} D_2$, то $\langle \Rightarrow v, D_1 \rangle \Downarrow D_2$.

6) Розіменування: якщо $\bar{v} \Rightarrow (D_1) \rightarrow_{\Rightarrow v} D_2$, то $\langle v \Rightarrow, D_1 \rangle \Downarrow D_2$.

Коректність цієї семантики визначається таким твердженням:

Теорема 2. $\langle P, D \rangle \Downarrow D'$ тоді і тільки тоді, коли $f(\gamma^{-1}(D)) \Downarrow \gamma^{-1}(D')$, де f – інтерпретація P .

Доведення аналогічне доведенню теореми 1.

4. Висновки

У статті надано операційну семантику програмам обробку складноіменних даних мови $SICON_{as}$. Отримані результати можуть бути корисними при застосуванні композиційно-номінативного підходу для розробки програмних систем або створення формальних моделей існуючих програмних систем.

Список використаних джерел

1. Басараб И.А. Композиционные базы данных / И.А. Басараб, Н.С. Никитченко, В.Н. Редько – К.: Либідь, 1992. – 191 с.
2. Nikitchenko N.S. A composition nominative approach to program semantic / N.S. Nikitchenko // Report IT-TR: 1998-020, Technical University of Denmark, 1998. – 103 p.
3. Никитченко Н.С. Композиционно-номинативный подход к уточнению понятия программы / Н.С. Никитченко // Проблемы программирования. – 1999. – № 1. – С. 16–31.
4. Plotkin G. A structural approach to operational semantics / G. Plotkin // Technical Report DAIMI FN-19, University of Aarhus, 1981.
5. Нікітченко М.С. Властивості композиційно-номінативних мов програм

- з асоціативним розіменуванням / Є.В. Іванов, М.С. Нікітченко // Матеріали XVI всеукраїнської наукової конференції "Сучасні проблеми прикладної математики та інформатики". – Львів. – 2009. – С. 157-158.
6. Нікітченко М.С. Стабільність та монотонність програм відносно структурних трансформацій даних / М.С. Нікітченко, Є.В. Іванов // Матеріали VII міжнародної наукової конференції УкрПрог. – Київ. – 2010. – С. 58-67.
7. Ivanov Ie. Modeling programs over complex-named data by term rewriting systems / Ie. Ivanov, M. Nikitchenko, L. Feraud, M. Strecker // Тези доповідей міжнародної наукової конференції TAAPSD'2010. – Київ. – 2010. – С. 40-49.

Надійшла до редколегії 15.12.10

УДК 62-50:519.7

Ковальов О.М., д.ф.-м.н., проф., Савченко О.Я.,
д.ф.-м.н., проф., Козловський В.А., к.ф.-м.н.,
с.н.с., Щербак В.Ф., к.ф.-м.н., с.н.с.

Обернені системи керування в алгоритмах перетворення інформації

Розглянуто деякі аспекти застосування
обернених систем керування для перетворення
та шифрування інформації. Вивчаються
неперервні та дискретні (скінченні автомати)
динамічні системи. Запропоновано схему
побудови оберненої системи та її модифікацій з
хаотичною поведінкою траєкторій. Досліджено
умови оберненості для динамічних систем над
скінченними полями.

Ключові слова: система керування, автомат,
оберненість, шифрування.

*E-mail: kozlovskii@iamm.ac.donetsk.ua, kozlovskii@list.ru

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

В останні десятиліття, з появою роботи [1],
формується напрямок досліджень, який
пов'язано з використанням в телекомунікаційних
технологіях динамічних систем, що мають
хаотичну поведінку (див., наприклад [2,3]).
Пристрій, що перетворює за допомогою хаосу
сигнал у передавачі з одного виду в інший,
називається хаотичним кодером. З його
допомогою можна змінювати інформацію таким
чином, що вона виявиться недоступною
сторонньому спостерігачу, але в той же час може
бути легко повернута до вихідного виду
спеціальною динамічною системою - хаотичним
декодером, що знаходиться на прийомному боці
комунікаційної системи. В роботі розглянуто
загальний підхід до побудови схем перетворення
інформації, який пов'язано із застосуванням
спеціального класу динамічних систем, відомих в
математичній теорії керування як нелінійні
обернені системи керування [4-6]. Такі системи
дозволяють відновлювати вхідний вплив на
нелінійну динамічну систему по інформації про її
реакцію, яка задається функцією виходу. При
цьому багатовимірність динамічної системи,
наявність регульованих внутрішніх зв'язків,
можливість забезпечити хаотичну поведінку
траєкторій дозволяє конструювати різні
декодери, у яких відновлення вихідного

A.M. Kovalev, prof., dr., A.Ya. Savchenko, prof.,
dr., V.A. Kozlovskii*, dr., V.F. Scherbak, dr.

Inverse control systems for algorithms of transformation of information

Some aspects of inverse control systems
application for information encryption are
described. Continuous and discrete (finite
automata) dynamical systems are considered. The
scheme of inverse system design and its
modifications for chaotic behavior of trajectories is
proposed. Invertibility conditions for the dynamical
systems over finite fields are studied.

Key Words: control system, automaton,
invertibility, encryption.

повідомлення може відбуватися за різними
схемами.

Обернені системи керування

Однією із основних конструкцій, пов'язаних
з оберненими задачами в теорії керування, є
динамічні системи «вхід-вихід»:

$$\dot{x} = f(x, u), x(0) = x_0, x \in R^n, u \in R^m \quad (1)$$

$$y = h(x, u), y \in R^k, \quad (2)$$

де $x(t)$ - стан системи, $u(t)$ - вхід (керування
або зовнішній вплив), $y(t)$ - вихід системи, що
задається функцією часу, значення якої
вважаються відомими на кожній траєкторії
системи (1). Сенс обернених задач пов'язано з
прикладними проблемами, які потребують
відновлення складових математичної моделі (1)
за інформацією про її вихід $y(t)$. Зокрема,
поширеною є задача спостереження, метою якої є
відновлення стану $x(t)$ при відсутності
інформації про початковий стан x_0 . Відомою є
також задача функціональної ідентифікації -
визначення значень функції входу $u(t)$.

В роботі розглянуто можливість
використання існуючих схем рішення задачі
ідентифікації в алгоритмах шифрування -
розшифрування інформації. Передбачається, що
перетворення та передача інформації
проводяться за наступною схемою. Як пристрій

кодування використовується динамічна система (1), яка перетворює вхід $u(t)$, який промодульовано цифрованим інформаційним повідомленням, у розв'язок $x(t, u, x_0)$ системи диференціальних рівнянь (1) із початковим значенням x_0 . Отримана функція часу $x(t, u, x_0)$ за допомогою виходу (2) переводиться у функцію $y(t) = h(x(t, u, x_0), u)$ - сигнал, що буде переданий приймальною стороною.

Якщо побудоване таким чином передатне відображення $u(t) \rightarrow y(t)$ є взаємно однозначним, то існує, принаймні теоретично, можливість визначення вихідного повідомлення $u(t)$ за наявною інформацією про реакцію системи - вихід $y(t)$.

В теорії керування обернення передатного відображення здійснюється шляхом побудови так званої оберненої системи керування [4]. Обернена система керування також є динамічною системою «вхід-вихід», яка за інформацією про сигнал $y(t)$ та його похідні, до деякого порядку α включно, $\dot{y}(t), \dots, y^{(\alpha)}(t)$ (вхід до оберненої системи), відтворює вихідне повідомлення $u(t)$ (вихід оберненої системи):

$$\begin{aligned} \dot{x} &= F(t, x, y(t), \dot{y}(t), \dots, y^{(\alpha)}(t)), \quad x(0) = x_0, \\ u(t) &= g(t, x, y(t), \dot{y}(t), \dots, y^{(\alpha)}(t)). \end{aligned}$$

Розглянемо схему побудови оберненої системи для одноканальної системи «вхід-вихід», тобто системи (1) зі скалярним входом $u(t)$ (функція, яка містить закодоване тим або іншим чином повідомлення, що підлягає шифруванню) та скалярним виходом $y(t)$ (сигналом, який буде розповсюджено по комунікаційних мережах). Зазначимо, що аналіз багатоканальних систем, для яких вимірності входу та виходу однакові, може бути проведено відповідно до приведеної нижче схеми. Структурно більш складні ситуації, при яких ці вимірності не співпадають, розглянуто в [4].

Одним з ключових параметрів при побудові обернених систем є відносний ступінь виходу, який визначається в результаті наступної процедури. Диференціюємо згідно системи (1) рівняння виходу (2) до такого порядку, коли в правих частинах відповідної похідної вперше явно ввійде вхід $u(t)$.

Маємо

$$\begin{aligned} \xi_1 &= y = h(x) = h_1(x), \\ \xi_i &= y^{(i-1)} = h_i(x) = \frac{\partial h_{i-1}(x)}{\partial x} f(x, u), \quad i = 2, \dots, \alpha \end{aligned}$$

$$y^{(\alpha)} = h_{\alpha+1}(x, u).$$

Число $\alpha \leq n$ називають відносним ступенем виходу (2) для системи (1).

Зробимо заміну змінних

$$\xi_i = h_i(x), \quad \eta_{j-\alpha} = x_j, \quad i = 1, \dots, \alpha, \quad j = \alpha + 1, \dots, n.$$

При умові, що ця заміна не є виродженою, систему (1) у нових змінних можна переписати у вигляді

$$\begin{aligned} \dot{\xi}_1 &= \xi_2, \\ \dot{\xi}_2 &= \xi_3, \\ &\dots \\ \dot{\xi}_\alpha &= H(\xi_1, \xi_2, \dots, \xi_\alpha, \eta, u), \\ \dot{\eta} &= G(\xi_1, \xi_2, \dots, \xi_\alpha, \eta, u). \end{aligned} \quad (3)$$

Змінні $\xi_1, \dots, \xi_\alpha$ та $\eta_1, \dots, \eta_{n-\alpha}$ визначають відповідно зовнішню і внутрішню динаміку системи (3). Оскільки в якості змінних $\xi_1, \dots, \xi_\alpha, \dot{\xi}_\alpha$ виступають вихід $y(t)$ та його відповідні похідні, то будемо вважати їх відомими функціями часу. Похідна порядку α від сигналу $y(t)$ вже залежить явним чином від входу u

$$\dot{\xi}_\alpha = H(\xi_1, \xi_2, \dots, \xi_\alpha, \eta, u) \quad (4)$$

Розв'язуючи рівняння (4) відносно u , одержуємо

$$u = H^{-1}(\xi_1, \xi_2, \dots, \xi_\alpha, \dot{\xi}_\alpha, \eta). \quad (5)$$

Остаточно, підставляючи (5) у групу рівнянь «внутрішньої» динаміки системи (3), одержуємо замкнуту систему диференціальних рівнянь порядку $n - \alpha$

$\dot{\eta} = G(\xi_1, \dots, \xi_\alpha, \eta, H^{-1}(\xi_1, \dots, \xi_\alpha, \dot{\xi}_\alpha, \eta)) = F(t, \eta)$ (6)
Система (6) і є системою, оберненою до системи «вхід-вихід» (1), (2). Входом до неї є відомі функції часу $\xi_1, \dots, \xi_\alpha, \dot{\xi}_\alpha$, а вихід задається співвідношенням (5). Якщо взяти початкові значення $\eta(0) = (x_{\alpha+1}(0), \dots, x_n(0))$, то, вирішуючи задачу Коші для системи (6), знаходимо чисельні значення компонент вектора $\eta(t, \eta_0)$. В результаті усі змінні в правій частині (4) стають відомими функціями часу, що дозволяє однозначно визначити вихідне повідомлення $u(t)$.

Отримане представлення (6), (5) є альтернативним описом динаміки вхід-вихід, яка задається системою (1), (2), за умови, що початкове значення стану x_0 для систем диференціальних рівнянь (1) і (6) співпадають. З урахуванням того, що входом для (6) є вектор виходу (2) разом зі своїми похідними до певного

порядку, а виходом – вхід до (1), це представлення природно вважати зворотним до вихідного. При цьому, якщо вихідна система формує передатне відображення $u(t) \rightarrow y(t)$, то для зворотної системи відповідне передатне відображення має вигляд $y(t), \dots, y^{(\alpha)}(t) \rightarrow u(t)$.

Зауваження. Нами було зроблено припущення про те, що функції $y(t), \dot{y}(t), \dots, y^{(\alpha)}(t)$ є відомими функціями часу. Проте, слід зазначити, що в разі, коли $y(t)$ – реальний неперервний сигнал, який приймальною стороною вимірюється деяким фізичним пристроєм, то в ньому, як правило, присутні адитивні збурення. Відомо, що в таких випадках чисельне визначення похідних пов'язано з великими похибками. З метою уникнути процедури чисельного диференціювання при побудові оберненої системи, наведена вище схема може бути модифікована наступним чином: по каналах зв'язку замість $y = h(x, u)$ передається сигнал

$$y(t) = \xi_\alpha = H(\xi_1, \xi_2, \dots, \xi_\alpha, \eta, u)$$

і тоді змінні $\xi_1, \xi_2, \dots, \xi_\alpha$ мають бути відновлені в результаті чисельного інтегрування. Обернена система для такого виходу матиме наступний вигляд:

$$\begin{aligned} \dot{\xi}_1 &= \xi_2, \\ \dot{\xi}_2 &= \xi_3, \\ &\dots \\ \dot{\xi}_\alpha &= y \\ \dot{\eta} &= G(\xi_1, \xi_2, \dots, \xi_\alpha, \eta, H^{-1}(\xi, y)), \\ u &= H^{-1}(\xi_1, \xi_2, \dots, \xi_\alpha, y, \eta) \end{aligned}$$

Оберненість є необхідною властивістю динамічних систем, які можуть бути використані в алгоритмах перетворення інформації за зазначеною схемою. Стандартною вимогою до такого роду алгоритмів є також хаотичне поведінка траєкторій. Оскільки відповідні властивості оберненості та хаотичності визначаються правими частинами диференціальних рівнянь $f(x, u)$ і функціями виходу $h(x, u)$, то розглянемо задачу про відповідний синтез цих функцій.

Синтез динамічних систем із властивостями оберненості та хаотичності.

Особливістю використання оберненої системи в алгоритмах кодування інформації є

можливість в значній мірі регулювати структуру передатного відображення $u(t) \rightarrow y(t)$ за рахунок додаткових алгебраїчних зв'язків між змінними в рівняннях внутрішньої динаміки. Іншими словами, задаючи ту чи іншу функціональну залежність $u(x, v(t))$ між станом x та інформаційною частиною входу $v(t)$ можна перенастроювати приймач (декодер), забезпечуючи одночасно із оберненістю деякі додаткові властивості систем, що розглядаються.

Нехай як передавач (кодер) обрана деяка система (1), (2). Тоді задача одержання запропонованих динамічних властивостей переданого сигналу може бути вирішена в результаті синтезу тієї чи іншої функціональної залежності $u(x, v(t))$. Припустимо, наприклад, що метою є спрощення динамічних рівнянь приймача. Тоді функція $u(x, v(t))$ в системі (3) може бути обрана з умови

$$H(\xi_1, \xi_2, \dots, \xi_\alpha, \eta, u) = \tilde{H}(\xi_1, \xi_2, \dots, \xi_\alpha, v),$$

де інформаційне повідомлення задається функцією $v(t)$. При цьому система (3) матиме вигляд

$$\begin{aligned} \dot{\xi}_1 &= \xi_2, \\ \dot{\xi}_2 &= \xi_3, \\ &\dots \\ \dot{\xi}_\alpha &= \tilde{H}(\xi_1, \xi_2, \dots, \xi_\alpha, v), \\ \dot{\eta} &= G(\xi_1, \xi_2, \dots, \xi_\alpha, \eta, u). \end{aligned} \quad (7)$$

В цьому випадку для розшифрування повідомлення приймачою стороною достатньо моделювати спрощену зовнішню динаміку у просторі, який задано координатами $\xi_1, \xi_2, \dots, \xi_\alpha$.

Для забезпечення хаотичного розташування траєкторій зовнішня динаміка передавача може бути зведена до деякої еталонної системи диференціальних рівнянь. Наприклад, як такі можуть бути використані системи диференціальних рівнянь Лоренца, Ресслера. Відомо, що при значеннях параметрів цих систем у визначених діапазонах, їхні траєкторії мають властивість хаотичності [6, 7].

Рівняння Ресслера:

$$\begin{aligned} \dot{x}_1 &= -x_2 - x_3 \\ \dot{x}_2 &= x_1 + ax_2 \\ \dot{x}_3 &= c + x_3(x_1 - b) \end{aligned} \quad (8)$$

Рівняння Лоренца:

$$\begin{aligned}\dot{x}_1 &= \sigma(x_2 - x_1), \\ \dot{x}_2 &= -x_1x_3 + rx_1 - x_2, \\ \dot{x}_3 &= x_1x_2 - bx_3,\end{aligned}\quad (9)$$

Нехай, наприклад, для системи (8) $y(t) = x_1$, тоді, за умови $a = \text{const}$ безпосередніми обчислюваннями отримуємо:

$$y^{(3)} = \ddot{y}(a - b + \dot{y} - ay) + \dot{y}(ab - 1 + y + ay) - by - c - a(\dot{y}^2 + y^2)$$

Тоді лінійна заміна змінних

$$(x_1, x_2, x_3)^T = A \cdot (\zeta_1, \zeta_2, \zeta_3)^T,$$

$$\text{де } A = \begin{pmatrix} -a & 1 & 0 \\ 1 & 0 & 0 \\ -1 & a & -1 \end{pmatrix}, \quad \det A = 1,$$

не змінюючи статистичних властивостей траєкторій, приводить систему до «трикутного» вигляду

$$\begin{aligned}\dot{\zeta}_1 &= \zeta_2, \\ \dot{\zeta}_2 &= \zeta_3, \\ \dot{\zeta}_3 &= c - b\zeta_1 + (ab - 1)\zeta_2 + (a - b)\zeta_3 - \\ &\quad - a(\zeta_1^2 + \zeta_2^2) + (1 + a)\zeta_1\zeta_2 - a\zeta_1\zeta_3 - a\zeta_2\zeta_3\end{aligned}\quad (10)$$

При цьому параметри b і c перетвореної системи можуть бути функціями часу, в які можна включити інформаційну складову $v_1(t), v_2(t)$: $b(t) = b_0(t) + v_1(t)$, $c(t) = c_0(t) + v_2(t)$.

З використанням такого представлення можна формувати хаотичний характер сигналу. Дійсно, нехай зовнішня динаміка передавача (3) зведена до вигляду

$$\begin{aligned}\dot{\zeta}_1 &= \zeta_2, \\ \dot{\zeta}_2 &= \zeta_3, \\ \dot{\zeta}_3 &= u(\zeta_1, \zeta_2, \zeta_3, v_1, v_2).\end{aligned}\quad (11)$$

Тоді, якщо в передавачі синтезовано залежність $u(\zeta_1, \zeta_2, \zeta_3, v_1, v_2)$ відповідно до останнього рівняння системи (10), то тим самим гарантовано забезпечується хаотичний характер сигналу $y = \zeta_1(t)$. Вказане рівняння системи (10) містить як невідомі: константу a , функції часу $b_0(t), c_0(t), v_1(t), v_2(t)$. Таким чином константи $\zeta_1(0), \zeta_2(0), \zeta_3(0), a$, функції часу $b_0(t), c_0(t)$, та одна із функцій $v_1(t), v_2(t)$ виступають як закриті параметри (ключ відповідного шифру).

Аналогічним чином для генерування хаотичного сигналу із збереженням властивості оберненості динамічної системи (кодера) можна використовувати систему диференціальних

рівнянь Лоренца (9). Перетворення змінних у цьому випадку нелінійне, однак хаотична поведінка траєкторій зберігається.

Нехай $y(t) = x_1(t)$ – сигнал, коефіцієнти r, σ є константами, а коефіцієнт b змінний і його промодульовано інформаційним повідомленням: $b(t) = b_0(t) + v(t)$.

Тоді в результаті відповідної кількості диференціювань отримуємо рівняння:

$$\begin{aligned}y^{(3)} - \frac{y^2 \dot{y}}{y} + (1 + \sigma)(\ddot{y} - \frac{\dot{y}^2}{y} + y\dot{y}) - \\ - y\dot{y}^2 - \sigma y^3 = b(\ddot{y} - \sigma(r + 1)y)\end{aligned}$$

Нелінійна невідокремлена заміна змінних в області, де $x_1 \neq 0$

$$\begin{aligned}x_1 &= \zeta_1, \\ x_2 &= \zeta_1 + \frac{\zeta_2}{\sigma}, \\ x_3 &= r - 1 - \frac{\zeta_2(1 + \sigma) - \zeta_3}{\sigma \zeta_1}\end{aligned}\quad (12)$$

приводить систему (9) до вигляду

$$\begin{aligned}\dot{\zeta}_1 &= \zeta_2, \\ \dot{\zeta}_2 &= \zeta_3, \\ \dot{\zeta}_3 &= \frac{(1 + \sigma)\zeta_2 - \zeta_3}{\zeta_1} \zeta_2 + (b - \sigma - 1)\zeta_3 - \\ &\quad - \zeta_1\zeta_2(\sigma + 1 - \zeta_1) + \sigma\zeta_1^3 - b(r - 1)\sigma\zeta_1.\end{aligned}\quad (13)$$

Остання форма представлення показує, що за інформацією про сигнал $y = x_1$ та його похідні $y, \dot{y}, \ddot{y}, y^{(3)}$ можна визначити значення b , а, отже, і вихідне повідомлення $v(t)$. Ключовими параметрами алгоритму шифрування в цьому випадку є константи $\zeta_1(0), \zeta_2(0), \zeta_3(0), r, \sigma$ та функція часу $b_0(t)$.

Оскільки алгебраїчна заміна (12) не міняє статистичних властивостей траєкторій системи Лоренца, то вибір в системі (11) функції $u(\zeta_1, \zeta_2, \zeta_3, v)$ у вигляді правої частини останнього рівняння (13) одночасно із можливістю відновлення значень $v(t)$ гарантує хаотичний характер сигналу.

Регулювання складності процедури відновлення значень входу можна також за рахунок змін рівнянь внутрішньої динаміки системи (3). Зокрема, з метою підвищення криптозахисту повідомлень, можна розглянути задачу не спрощення, а ускладнення динаміки передавача і приймача. У цьому випадку процедура синтезу функції $u(x, v(t))$ може бути

використана для збільшення числа змінних стану за рахунок збільшення вимірності їх стану та введення додаткових залежностей, які задано системою диференціальних рівнянь

$$\dot{t} = \Phi(\xi, \eta, \tau, v).$$

Таким чином, можна регулювати в широких межах складність поведінки систем і процес відновлення вихідного сигналу за рахунок таких параметрів, як вимірність, наявність інформації про параметри системи та її початковий стан і спосіб модулювання вхідного сигналу інформаційним повідомленням.

Оберненість скінченних динамічних систем

Неперервні динамічні системи при їхній реалізації на комп'ютері замінюються дискретними обчислювальними моделями. Обчислення з обмеженою точністю можна виконати, переходячи від диференціальних рівнянь до скінченно-різницевого. Якщо, у зв'язку з скінченною розрядністю будь-якого комп'ютера, обмежуватися скінченним числом значень всіх параметрів і змінних, які є у вихідній системі, то отриману систему рівнянь можна розглядати як опис деякого скінченного автомата.

Скінченний автомат це п'ятірка об'єктів $A = (S, X, Y, \delta, \lambda)$ [8], де S – (скінченна) множина станів, X – (скінченний) вхідний алфавіт, Y – (скінченний) вихідний алфавіт, $\delta: S \times X \rightarrow S$, $\lambda: S \times X \rightarrow Y$ – функції переходів і виходів відповідно. Функції δ й λ розширюються на множину X^* слів скінченної довжини звичайним чином.

Автомат можна описати таблицею, що задає функції переходів і виходів, або графом переходів, або системою рівнянь (використовуючи операції k -значної логіки, зокрема, двозначної, або операції у скінченному кільці або полі).

Перші два способи опису (особливо у вигляді графа переходів), наочно подають інформацію про структуру й ряд властивостей автомата.

Однак ця перевага губиться при великих потужностях його алфавітів. У цьому випадку більш зручним і компактним є опис автомата у вигляді системи рівнянь у належній алгебрі. При цьому передбачається, що автомат працює в дискретному часі t , $t \in T = \{0, 1, 2, \dots\}$ і описується так званими канонічними рівняннями:

$$\begin{aligned} s(t+1) &= \delta(s(t), x(t)), \\ y(t) &= \lambda(s(t), x(t)). \end{aligned} \quad (14)$$

Перехід від диференціальних рівнянь до їхніх скінченно-різницевого аналогів приводить до рівнянь, у яких присутні 4 арифметичні операції. Скінченність числа значень величин, що входять до них, і збереження форми рівнянь, які відбивають зв'язок між цими величинами, роблять природним розгляд цих рівнянь як рівнянь у деякому скінченному полі $GF(q)$ (або, у більш простих ситуаціях, у скінченному кільці Z_q).

Якщо діапазон зміни величин, що зустрічаються в згаданих вище скінченно-різницевого рівняннях, обмежений і скінченний, то можна підібрати підходящий порядок q поля, при якому зазначені величини будуть ін'єктивно відображатися в елементи поля $GF(q)$, а інтерпретація арифметичних операцій, як операцій у полі $GF(q)$, дасть опис автомата відповідною системою рівнянь.

Розглянемо такий перехід на прикладі системи Лоренца [7].

Система Лоренца описується рівняннями:

$$\dot{x} = -\sigma x + \sigma y, \dot{y} = rx - y - xz, \dot{z} = -bz + xy,$$

де σ, r і b – позитивні константи.

У вихідну систему вводиться вхідний сигнал $u(t)$, вихід $v(t)$ покладається рівним $y(t)$, в результаті чого друге рівняння набуває вигляду:

$$\dot{y} = rx - y - xz + au.$$

Дискретизація приводить до таких скінченно-різницевого рівнянь:

$$\begin{aligned} x(t+h) &= x(t) + h\sigma(y(t) - x(t)) \\ y(t+h) &= y(t) + h(rx(t) - y(t) - x(t)z(t) + au(t)) \\ z(t+h) &= z(t) - bz(t) + x(t)y(t) \\ v(t) &= y(t) + h(rx(t) - y(t) - x(t)z(t) + au(t)) \end{aligned}$$

Обернена система містить ті ж перше й третє рівняння, а друге замінюється на наступне:

$$y(t+h) = v(t)$$

В оберненій системі входом стає $v(t)$, а виходом $u(t)$, який визначається рівнянням:

$$u(t) = ((v(t) - y(t))/h - rx(t) + y(t) + x(t)z(t))/a$$

Покладаючи у розглянутих рівняннях $h=1$ і вводячи узагальнюючі коефіцієнти, отримаємо систему параметризованих рівнянь. Якщо таку систему розглядати у деякому скінченному полі $GF(q)$, то вона описує клас скінченних динамічних систем – автоматів, які будемо називати автоматами Лоренца [9]. У змінних автоматних алфавітів відповідна система матиме вигляд:

$$\begin{aligned} s_1(t+1) &= as_1(t) + bs_2(t) \\ s_2(t+1) &= cs_1(t) + ds_2(t) + es_1(t)s_3(t) + \alpha u(t) \\ s_3(t+1) &= fs_3(t) + gs_1(t)s_3(t) \\ y(t) &= s_2(t+1) \end{aligned} \quad (15)$$

Основною властивістю систем, що здійснюють перетворення інформації, є оберненість, якщо потрібне однозначне відновлення вихідного повідомлення. Це накладає на скінченну динамічну систему структурні й поведінкові обмеження, які відбивають різні типи оберненості, такі, наприклад, як структурна оберненість, що приводить до групових автоматів, або оберненість по поведінці, що приводить до автоматів без втрати інформації (БВІ-автомати).

Для вище визначених обернених неперервних динамічних систем автоматними аналогами можна вважати БВІ-автомати. Для автоматів властивість оберненості за поведінкою виражається в тому, що можливо відновлення вхідної послідовності за відомою вихідною (при відомому початковому стані). Таке відновлення може здійснюватися з певною затримкою, тобто перший символ вхідного слова однозначно визначається за вихідним словом довжини, не меншої деякого k . У цьому випадку такі автомати називаються автоматами без втрати інформації порядку k (БВІ- k -автомати). Автомат A є БВІ-автоматом, якщо при відомих початковому стані й вихідному символі однозначно визначається відповідний вхідний символ, тобто рівняння $\lambda(s, x) = y = \lambda_s(x)$ однозначно розв'язується відносно x при відомих стані s і виході y (позначаємо $x = \lambda_s^{-1}(y)$).

Автомат Лоренца з функцією виходів

$$y(t) = cs_1(t) + ds_2(t) + es_1(t)s_3(t) + \alpha x(t)$$

є БВІ-автоматом при $\alpha \neq 0$ у будь-якому полі $GF(q)$, тому що рівняння, що описує функцію виходів, однозначно розв'язується щодо входу $x(t)$ при відомих $y(t)$ і $s(t) = (s_1(t), s_2(t), s_3(t))$. Властивість автомата A «бути БВІ-автоматом» визначає існування оберненого за поведінкою для A автомата. Ясно, що якщо A є БВІ-автомат, то рівняння $\lambda(s, p) = q$ розв'язується однозначно щодо вхідного слова p довільної довжини при будь-яких відомих стані s і вихідному слові q .

Поняття оберненості динамічної системи можна поширити на випадок множини траєкторій подібно до того, як це зроблено в [4], і яке узагальнює звичайне поняття оберненості системи. Введемо відповідне узагальнення

автомату без втрати інформації. Автомат A назовемо k -кратним без втрати інформації порядку m автоматом (скорочено k -БВІ- m автоматом), якщо можна однозначно визначити перший символ вхідного слова p довжини m по реакціях на нього k різних станів автомата, причому існує таке вхідне слово, що для його однозначного визначення необхідно знати реакції на нього k різних станів автомата, а m найменше число, при якому можливо таке відновлення. Звичайний БВІ-автомат є 1-БВІ-1 автомат. Якщо автомат є k -БВІ автоматом, то це означає, що система рівнянь $\lambda(s_i, p) = q_i$, $i = 1, \dots, k$, завжди має єдиний розв'язок відносно p при попарно різних станах s_i і їхніх відомих реакціях q_i на вхідне слово p довільної довжини.

Розглянемо наступну конструкцію, що зводить вивчення k -БВІ-автоматів до вивчення звичайних БВІ-автоматів. Стан s автомата A назовемо станом без втрати інформації, якщо рівняння $\lambda(s, x) = y = \lambda_s(x)$ однозначно розв'язне відносно x при відомому y .

Визначимо по автомату $A = (S, X, Y, \delta, \lambda)$ автомат $A_k = (S_k, X_k, Y_k, \delta_k, \lambda_k)$: $S_k = S^k$ – k -а декартова ступінь множини станів автомата A , $X_k = X$, $Y_k = Y^k$ – k -а декартова ступінь множини вихідних символів Y ; якщо $s = (s_1, \dots, s_k) \in S^k$, $y = (y_1, \dots, y_k) \in Y^k$, то $\delta_k(s, x) = (\delta(s_1, x), \dots, \delta(s_k, x))$; $\lambda_k(s, x) = (\lambda(s_1, x), \dots, \lambda(s_k, x)) = y$.

У цьому автоматі виділимо множину AD_k тих його станів $s = (s_1, \dots, s_k) \in S^k$, у яких усі компоненти попарно різні.

Справедливе твердження [9].

Теорема 1. Автомат $A = (S, X, Y, \delta, \lambda)$ є k -БВІ-автоматом тоді й тільки тоді, коли в автоматі $A_k = (S_k, X_k, Y_k, \delta_k, \lambda_k)$ всі стани AD_k є станами без втрати інформації.

Варто помітити, що якщо вихідний автомат груповий, то підавтомат, породжуваний в A_k станами AD_k , є груповий БВІ-автомат. У цьому випадку породжений підавтомат відразу ж доставляє конструкцію до вихідного автомата через обернення зазначеного БВІ-автомата. Властивість «бути автоматом без втрати інформації» перевіряється за допомогою спеціальної таблиці, що неприйнятно при великих потужностях алфавітів автомата. Нижче наводиться достатня умова оберненості при

фіксації стану для аналітичного варіанту завдання автомата.

Нехай вимірності вхідних і вихідних векторів є m , а векторів станів n , $m \leq n$. Будемо вважати, що у всіх розглянутих автоматних вектор-функціях елементи представлені у вигляді поліномів від відповідних змінних. (Частинною) похідною полінома $p(x_1, \dots, x_i, \dots, x_k)$ над $GF(q)$ за змінною x_i назовемо поліном $\frac{\partial p}{\partial x_i}$ над цим полем,

отриманий за звичайними правилами обчислення похідної. Нехай $y(t) = (y_1(t), y_2(t), \dots, y_m(t))$.

Позначимо через $\frac{\partial y}{\partial x}$ матрицю, елементи якої $y_i = \frac{\partial y_i}{\partial x_j}$. Назвемо її матрицею Якобі скінченної

динамічної системи в поліноміальному поданні (над полем $GF(q)$).

Теорема 2. Якщо $\det \frac{\partial y}{\partial x} = f(s)$, f залежить тільки від s , і $f(s) \neq 0$, то стан s автомату, заданого рівняннями (14), є БВІ-станом.

Доведення. Нехай $\det \frac{\partial y}{\partial x} = f(s)$. Приведемо матрицю до трикутного вигляду. Тоді на головній діагоналі будуть стояти елементи, що залежать тільки від s і $f(s) = f_1(s) \dots f_j(s) \dots f_m(s)$. Отже, система рівнянь може бути однозначно розв'язана щодо змінних x методом послідовного виключення змінних, тобто, відповідний стан s є станом без втрати інформації. Точніше, знайдені на попередніх 1, 2, ..., j -ому кроках змінні $x_1, x_2, \dots, x_j$ (якщо необхідно, всі змінні можуть бути перенумеровані належним чином), після підстановки в $j+1$ -е рівняння приводять останнє до виду $w_{j+1}(y, s) = v_j(s) + f_j(s)x_{j+1}$. Тоді,

$$x_{j+1} = \frac{w_{j+1}(y, s) - v_j(s)}{f_j(s)}, \quad f_j(s) \neq 0 \text{ за умовою.}$$

Таким чином, існує єдиний розв'язок системи рівнянь (14).

Зауваження. Теорема 2 дає локальну умову оберненості, яка перетворюється в глобальну, якщо $\det \frac{\partial y}{\partial x} \neq 0$ при будь-якому s , тобто є або константою, або поліномом, який не має корінь у полі $GF(q)$. У цьому випадку з теореми 2 випливає достатня умова для автомата «бути БВІ-автоматом».

Можна різними способами вводити інформаційні входи в автономну динамічну систему. Звичайно розглядається адитивне введення входу [6]. У цьому випадку узагальнені рівняння автомата приймають вигляд:

$$s(t+1) = F(s(t)) + G(x(t));$$

$$y(t) = M(s(t+1)),$$

де F, G, M – вектор-функції належних вимірностей.

У відомих з літератури випадках M і G є лінійними. Такими системами є запропоновані у [6] системи перетворення інформації на основі систем Лоренца, Ресслера. Побудовані за цими системами відповідні автомати Лоренца, Ресслера задовольняють умові глобальної оберненості з вищенаведеного зауваження.

Зауважимо, що вимірності векторів x і y збігаються, але можуть не збігатися з вимірністю вектора s .

За аналогією з адитивним введенням інформаційного сигналу можна визначити й чисто мультиплікативний варіант такого введення. У цьому випадку рівняння автомата приймають вигляд:

$$s(t+1) = F(s(t)) * G(x(t));$$

$$y(t) = M(s(t+1)).$$

Теорема 2 дозволяє запропонувати варіанти перетворювачів на основі системи Лоренца, що відрізняються від адитивного способу підмішування інформаційного сигналу, як це зроблено в системі (15). Нижче наведена така система:

$$s_1(t+1) = s_1(t) + hA_1(s_2(t) - s_1(t)) + B_1s_1(t)$$

$$s_2(t+1) = s_2(t) + h(A_2(s_1(t) - s_2(t) - s_1(t)s_3(t) + B_2s_1^2(t) + B_3f(s)x_2(t))$$

$$s_3(t+1) = s_3(t) + h(s_1(t)s_2(t) - A_3s_3(t))$$

$$y_1(t) = s_1(t+1)$$

$$y_2(t) = s_2(t+1)$$

Передбачається, що зазначений у системі поліном $f(s)$ не має коренів у $GF(q)$.

Система задовольняє умовам теореми 2 і є оберненою. Вона істотно складніше системи 2.

Таким чином, теорема 2 може бути використана для синтезу скінченних обернених динамічних систем на основі різних способів введення інформаційного сигналу в автономні системи-прототипи, причому більш складним образом, ніж це було зроблено вище для автомата Лоренца.

Ця теорема може бути корисною і при аналізі узагальненої оберненості скінченних динамічних систем-перетворювачів. При описі автомата A

системою рівнянь у полі $GF(q)$ легко побудувати відповідну систему рівнянь автомата A_k . Така система отримується дублюванням системи вихідного автомата з перейменуванням змінних так, щоб всі змінні станів і виходів автомата стали різними. Якщо у вихідній системі вектор змінних станів $\epsilon s = (s_1, \dots, s_n)$, то в новій системі визначаємо стан як вектор $s^{(k)} = (s_1^1, s_2^1, \dots, s_n^1, \dots, s_1^k, s_2^k, \dots, s_n^k)$. Аналогічно визначається і вектор виходів. У цьому випадку при виконанні умов теореми 1 може бути використана локальна умова оберненості для всіх станів нової системи, що задовольняють умові попарної нерівності координат вектора стану: $\prod_{i \neq j, j \neq i} (s_i^j - s_j^i) \neq 0$. Таку узагальнену оберненість за декількома траєкторіями можна використовувати при передачі зашифрованої інформації по декількох незалежних каналах.

Список використаних джерел

1. Pecora L.M. Synchronization in chaotic systems. / L.M. Pecora, T.L. Carrol // Physical Review Letters. – 1990. – Vol. 64(8).: – P. 821-824.
2. Feldman U. Communication by chaotic signals: the inverse system approach. / U. Feldmann, M. Hasler, W. Schwarz // Int. J. Circ. Theory Appl. – 1996. – Vol. 24, – P. 551-579.
3. Дмитриев А.С. Динамический хаос как парадигма современных систем связи / А.С. Дмитриев, А.И. Панас, С.О. Старков. // Зарубежная радиоэлектроника. Успехи современной радиоэлектроники. – 1997. – №10. – С. 4-26.
4. Ковалев А.М. Управляемость, наблюдаемость, идентифицируемость динамических систем. / А.М. Ковалев, В.Ф. Щербак. – Киев: Наук. думка. - 1993. - 285 с.

Розшифрування буде можливим лише за наявності інформації, переданої по усіх каналах.

Висновки

Запропоновано схеми перетворення інформації без втрати інформації, на основі способів побудови обернених динамічних систем в теорії керування. Розглянуто як неперервні так дискретні динамічні системи у скінченних полях. Показано, що в таких системах можна впливати на складність перетворення завдяки різним способам включення інформаційного сигналу, як керуючого, у відповідну динамічну систему. Дослідженні властивості оберненості можуть бути використані при створенні нових алгоритмів перетворення та захисту інформації.

5. Kovalev A.M. Criteria for the functional controllability and invertibility of nonlinear systems / A.M Kovalev // J.Appl. Math.Mech. – 1998. – Vol. 62(1). – P. 103-113.
6. Sobhy M.I., Secure computer communication using chaotic algorithms / M.I. Sobhy, A. Shehata // Int. J. of Bifurcation and Chaos. – 2000. – Vol.10, №12. – P. 2831-2839.
7. Данилов Ю.А. Лекции по нелинейной динамике / Ю.А. Данилов. – Москва: Постмаркет, 2001. – 184 с.
8. Кудрявцев В.Б. Введение в теорию автоматов / В.Б. Кудрявцев, С.В. Алешин, А.С. Подколзин. – Москва: Наука, 1985. – 320с.
9. Козловский В.А. Автоматные аналоги динамических хаотических систем / В.А. Козловский, Л.А. Толмачевская // Труды ИПММ НАН Украины. – 2003. – Т. 8. – С. 59-69.

Надійшла до редколегії 15.12.10

УДК 004.4

Кульчицький Ю.М., аспірант

Паралельне програмування на основі скелетонів

Основою метою прискорення обчислень за допомогою паралельних систем є одночасність виконання дій багатьма процесорами чи багатьма комп'ютерами, що об'єднані в мережу. Але в дійсності це не вирішує в такий спосіб проблему прискорення, її просто перекладено на реалізацію багатопроцесорних, кластерних і розподілених систем, які вносять додаткові складнощі. Таким чином, занурюючись в деталі паралельного програмування, ми втрачаємо і нівелюємо основну мету паралельного програмування — розпаралелювання задачі й досягнення високих показників швидкості обчислень. Таку мету паралельних обчислень покликано реалізувати скелетони — алгоритмічні форми, елементарні одиниці програмування, з яких повинні будуватися паралельні програми. У праці розглядаються скелетони паралельних типів даних як функції високих порядків та операції над ними, базовані на алгебрі списків; а також визначається клас скелетонів (катаморфізми) для паралельних реалізацій.

Ключові слова: скелетон, катаморфізм, алгебра списків, функтор.

*E-mail: yuscoolya@ukr.net

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Паралельні обчислення є основною формою обчислень для складних виробничих та наукоємних задач (які, зазвичай, є і надзвичайно ресурсоємними), забезпечуючи таке необхідне пришвидшення отримання результатів, порівняно з послідовними обчисленнями.

Зараз важко собі уявити, які методи розв'язання задач з біоінформатики (наприклад, укладки білку) чи економіки (наприклад, симуляцій в математичних фінансах) довелося б

Yu.M. Kulchytsky, postgraduate student

Skeleton-based Parallel Programming

The main purpose of accelerating computations using parallel systems is the simultaneity of execution of actions by many processors or multiple computers that are networked. But in reality this does not solve the problem of acceleration; it is simply being transferred onto implementation of multiprocessor, cluster, and distributed systems, which bring additional complexities. Thus, plunging into details of parallel programming, we lose and eliminate the main purpose of parallel programming — paralleling the problem itself and in this way achieving high indicators of computing speed. This purpose of parallel computing are called to implement skeletons — algorithmic forms, elementary units of programming, which should be the building blocks of parallel programs. The paper contains description of data-parallel skeletons (as higher-order functions) and operations on them (based on the algebra of lists), as well as the definition of the class of skeletons (catamorphisms) for parallel implementations.

Key Words: skeleton, catamorphism, algebra of lists, functor.

вигадувати людству, якби не було можливості розпаралелити обчислення. Паралельні обчислення застосовуються в багатьох класичних задачах, зокрема:

- в спектральних методах;
- для симуляцій методу Монте Карло;
- у комбінаторній логіці (наприклад, в криптографічних техніках);
- в обході графів (наприклад, в алгоритмах сортування);
- у динамічному програмуванні;

- у методі гілок і границь;
- у графічних моделях (таких, як пошук прихованих моделей Маркова та побудова Баєсівських мереж);
- та в багатьох інших задачах

Перспективним напрямком досліджень було і залишається удосконалення та розвиток засобів паралельних обчислювальних систем, програм, програмних комплексів.

Основою метою прискорення обчислень за допомогою паралельних систем є одночасність виконання дій багатьма процесорами чи багатьма комп'ютерами, що об'єднані в мережу. Але в дійсності це не вирішує в такий спосіб проблему прискорення, її просто перекладено на реалізацію багатопроцесорних, кластерних і розподілених систем, які вносять додаткові складнощі. Таким чином, занурюючись в деталі паралельного програмування, ми втрачаємо і нівелюємо основну мету паралельного програмування — розпаралелювання задачі й досягнення високих показників швидкості обчислень.

Таку мету паралельних обчислень покликано реалізувати скелетони — алгоритмічні форми, елементарні одиниці програмування, з яких повинні будуватися паралельні програми [1]. Ідея використання таких форм паралельної абстракції є еволюційним кроком розвитку від мов низького рівня до мов високого рівня. За розвитком скелетонам передують функціональне програмування, що поєднує у собі функції високого порядку і параметричний поліморфізм [2], отже з деяким наближенням скелетонами можна назвати поліморфні функції високих порядків.

Поняття «скелетон» вперше використав Мюррей Коул (Murray Cole) в 1989 році. Але необхідність такої концепції обговорювалася багатьма спеціалістами ще з кінця 70-х років минулого століття.

Синонімами скелетонів іноді називають шаблони, патерни, парадигми і т.п. Проте, скелетон — це алгебраїчна сутність, а шаблон, наприклад, є лише засобом представлення цієї сутності в певній мові програмування.

Часто трапляється, що програміст, фокусуючись на паралелізмі застосувань, забуває про апаратну особливість виконання на одній машині, внаслідок чого, програма залишається

паралельною настільки, наскільки вона була запрограмована як паралельна. Новий погляд — це використання паралельних типів даних, визначення й довизначення старих [3].

Звичайно, різні види таких функцій-скелетонів (багатозначні, клієнт-серверні) відображатимуть відповідну специфіку, а тому в нашій інтерпретації орієнтуватимемося на основний клас скелетонів — скелетони паралельних даних.

Основна ідея програмування за допомогою скелетонів — це поділ паралельного програмування на рівень реалізацій та застосування. Вважається, що програміст визначить структуру майбутньої програми заздалегідь, а основна реалізація та взаємодія блоків, з яких вона складається, буде реалізована в компіляторах чи бібліотеках. Звичайно, у таких пошуках конструктивного алгоритмування не обійтись без формалізмів і теорем, що встановлюють сам спосіб програмування за допомогою скелетонів [4].

У праці будуються скелетони паралельних типів даних на списках як функції високих порядків, а також операції над ними, базовані на алгебрі списків; визначається клас скелетонів (катаморфізми) для паралельних реалізацій.

Скелетони паралельних типів даних на списках

Список — найчастіше використовуваний тип даних, за допомогою якого можна подати інші класичні типи даних, такі як, масив, дерево, граф і т.п.

Скелетоном називатимемо множину операцій над списком (-ами).

Множиною рівностей над цими операціями вважатимемо основні перетворення, що відбуваються в процесі трансформації програми.

Множина рівностей і множина операцій і утворюють алгебру (в нашому випадку алгебру на списках).

Нотація цієї алгебри базується на конструктивному алгоритмуванні, так званому формалізмі Бірда-Міртенса (Bird-Meertens Formalism).

Так, звична для нас $f(x)$ або $f \in A \rightarrow B$ (де A — область визначення, а B — область допустимих значень), позначатиметься дуже просто — fx чи $f * x$.

Для множини A введемо функцію пошуку аргументу id , за означенням вважатимемо, що $id * f = f$ і $f * id = f$.

Для двох функцій $f \in A \rightarrow B$ і $g \in B \rightarrow C$, їхня композиція $f * g \in A \rightarrow C$ — результат застосування функції f над аргументом, тобто для будь-якого $a \in A$, $(f * g)a = g(fa)$. Фундаментальною властивістю такої композиції є необхідна асоціативність: $f * (g * h) = (f * g) * h$. Асоціативність дозволяє нам писати вираз без дужок (тобто без урахування) порядку виконання.

У реальному програмуванні id визначає стан, при якому не відбувається жодних обчислень.

Асоціативність визначає такі обчислення, при яких порядок виконання команд не важливий, а це і є основа паралельних обчислень.

Нашим типом даних вважатимемо множину еквівалентних чисел порядку натуральних чисел. Друге обмеження таке, що цей тип похідний від алгебри, і всі конструкції нашого типу даних — це операції алгебри.

Розглянемо таку алгебру — алгебру непорожніх списків.

Наш непорожній тип даних складатиметься з таких застосувань конструкцій:

- функція $sole$, $sole \in A \rightarrow [A]$, наприклад для одного елемента $a \in A$ $sole \in a \rightarrow [a]$ (список з одного елемента)
- асоціативна бінарна операція $concatenation$ ($++$), що об'єднує два списки в один:

$$[a_1, \dots, a_m] ++ [b_1, \dots, b_n] = [a_1, \dots, a_m, b_1, \dots, b_n]$$

На основі вище означеного непорожнього типу даних непорожній список визначимо нижче наведені функції вищого порядку.

Визначення 1. map -функціонал породжує для кожної функції на списку елементів, $f \in A \rightarrow B$, функцію на списках, $map f \in [A] \rightarrow [B]$, яка застосовує функцію f до кожного елемента списку аргументів. Тобто, для списку $[a_1, \dots, a_m]$,

$$map f [a_1, \dots, a_m] = [fa_1, \dots, fa_m]$$

Звичайно, базовими властивостями функції map є ідентичність і композиція, вище зазначена основа паралельних перетворень, тобто:

$$map id = id$$

$$map (f * g) = map f * map g$$

Визначення 2. red -функціонал породжує для бінарної асоціативної (базової) операції над елементами списку, $\otimes \in A \times A \rightarrow A$, функцію на списках, $red(\otimes) \in [A] \rightarrow A$, таку, що:

$$red(\otimes)[a_1, \dots, a_m] = a_1 \otimes \dots \otimes a_m$$

Наступний наш крок — узагальнення всіх функцій, що мають відношення до семантики паралельних обчислень і визначення цілого класу функцій над не порожніми списками. Називатимемо їх катаморфічними.

Визначення 3. Функція $h \in [A] \rightarrow X$ називається списково катаморфічною відносно бінарної операції $\otimes$, якщо:

$$h(l1 ++ l2) = (hl1) \otimes (hl2)$$

У деяких працях стосовно формалізму Бірда-Міртенса використовується термін «гомоморфізм» для позначення вище наведених функцій. Проте, таке позначення вживається в алгебрі у більш загальному сенсі, для позначення функції між двома алгебрами, що зберігає структуру цих алгебр. Тому ми надамо перевагу терміну «катаморфізм», який є спеціальним видом гомоморфізму.

Оскільки операція $++$ асоціативна, то і операція $\otimes$ повинна бути також асоціативною.

Основну властивість катаморфізму розкриває наступна теорема.

Теорема 1 (Факторизація). Функція h над не порожніми списками є катаморфічною відносно бінарної операції $\otimes$, якщо її можна факторизувати таким чином:

$$h = map \phi * red(\otimes), \text{ де } \phi a = h[a]$$

Доведення. Для будь-якого $a \in A$ виконується $h = map \phi * red(\otimes)$, де $\phi a = h[a]$. За **Визначенням 1**: $h = [ha] * red(\otimes)$. Оскільки операція $*$ — асоціативна, то $h = red(\otimes) * [ha]$.

Для будь-яких $l_1, l_2 \in A$ завжди знайдеться $a = l_1 ++ l_2$, так як a довільне. Враховуючи *Визначення 2* $h = h l_1 \otimes h l_2$, тобто $h(l_1 ++ l_2) = (h l_1) \otimes (h l_2)$, а за *Визначенням 3* маємо, що h — списково-катаморфічна. Теорему доведено.

З теореми про факторизацію випливає, що *map*-функціонал визначається однозначно такими рівностями:

$$\text{map } f[a] = [fa],$$

$$\text{map } f(l_1 ++ l_2) = (\text{map } f l_1) ++ (\text{map } f l_2),$$

для будь-якого елемента $a \in A$ а і списків $l_1, l_2 \in [A]$.

А *red*-функціонал визначається однозначно такими рівностями:

$$\text{red}(\otimes)[a] = a,$$

$$\text{red}(\otimes)(l_1 ++ l_2) = (\text{red}(\otimes) l_1) \otimes (\text{red}(\otimes) l_2),$$

для будь-якого елемента $a \in A$ а і списків $l_1, l_2 \in [A]$.

Практичне значення цієї теореми полягає в тому, що *map* $\phi * \text{red}(\otimes)$ — це кандидат на паралельну реалізацію. Реалізація міститиме два рівні: спершу, функція ϕ обраховується паралельно на кожному процесорі (*map*-функціонал); потім, остаточний результат конструюється з часткових результатів у процесорах за допомогою *red*-функціоналу. Отже, якщо ми зможемо довести, що поставлена перед нами задача — списково-катаморфічна, то ми зможемо розв'язати її в такий спосіб. Теорема факторизації застосовна тільки тоді, коли ми можемо представити нашу обчислювальну задачу як списковий катаморфізм.

Інша особливість цієї унікальної властивості полягає в тому, що і для двох катаморфічних функцій розпаралелювання відбуватиметься аналогічно.

Теорема 2 (Узагальнення). Для будь-якого спискового катаморфізму h стосовно асоціативної операції $\otimes$ виконується наведена нижче рівність:

$$\text{red}(++) * h = \text{map } \phi * \text{red}(\otimes), \text{ де } \phi = h[a]$$

Доведення. Права частина рівності має формат катаморфізму, згідно теореми про факторизацію. Завдяки властивості унікальності, необхідно довести, що ліва частина рівності є списковим катаморфізмом. Для будь-яких $l_1, l_2 \in A$, таких, що $l_1 ++ l_2 = a$ (чи те саме, що $[l_1, l_2]$),

$\text{red}(++) * h(l_1 ++ l_2) = \text{red}(++) h(l_1 \otimes l_2)$, за *Визначенням 3*. За наслідком для *red*-функціоналу з *Теореми 1*:

$$\text{red}(++) (h l_1 \otimes h l_2) = (\text{red}(++) * h) l_1 \otimes (\text{red}(++) * h) l_2$$

Враховуючи асоціативність операції $*$ і *Визначення 2*, одержуємо:

$$\text{red}(++) (h l_1 \otimes h l_2) = \text{red}(\otimes) [h * \text{red}(++) l_1, h * \text{red}(++) l_2]$$

За *Визначенням 1* права частина останньої рівності тотожна:

$$\text{red}(\otimes) \text{map } h[\text{red}(++) l_1, \text{red}(++) l_2],$$

Що, в свою чергу, знову за *Визначенням 2* та, враховуючи асоціативність операції $*$, тотожно:

$$\text{map } h[l_1, l_2] * \text{red}(\otimes)$$

Отже, для будь-якого $a \in [l_1, l_2] \in A$: $\text{red}(++) * h = \text{map } \phi * \text{red}(\otimes)$, де $\phi = h[a]$. Теорему доведено.

Поняття катаморфізму покриває великий клас задач. Щоправда, воно не покриває усі обчислювальні задачі: наприклад, катаморфічні обчислення є завжди скінченними. Більш того, існують прості скінченні задачі, що не є катаморфічними, а, так звані «майже катаморфічними» [1]. Приклади таких задач не є предметом розгляду даної праці.

Категоріальні типи даних

Скелетони паралельних даних і їх типові представники функціонали *red* та *map* включають в себе обчислення і взаємодію.

Аналогічно типам даних, скелетони також можна генерувати і накопичувати. Ключем до розуміння загальних властивостей функціоналів є теорія категорій, що надає рекурсивне

представлення таких типів даних як списки і дерева, наприклад.

Оскільки головна мета — це реалізовані скелетони, на основі яких будується паралельна програма, то для компілятора скелетонів потрібно передавати сигнатури операції й інші параметри необхідні для успішної її реалізації на різних платформах. Також, необхідно визначити механізм визначення сигнатур, а таким в теорії категорій є поняття функтора.

Визначення 4. Категорія C складається з колекції $\text{obj } C$ об'єктів, і, для кожних двох об'єктів $A, B \in \text{obj } C$, містить колекцію $A \xrightarrow{C} B$ морфізмів, таку, що:

- для кожних двох морфізмів, $f \in A \xrightarrow{C} B$ і $g \in A \xrightarrow{C} B$ існує їхня композиція $f * g \in A \xrightarrow{C} C$ з асоціативною операцією *
- для кожного об'єкта A цієї категорії існує морфізм пошуку аргументу $\text{id} \in A \xrightarrow{C} A$, такий що $f * \text{id} = \text{id} * f = f$

Таким чином, ми розглядаємо типи даних як об'єкти зручної категорії, а морфізми встановлюють еквівалентні за функціональністю програми, ґрунтуючись на семантичній еквівалентності.

Найбільш популярним частковим прикладом категорії є множина, а морфізмом, відповідно, — функції.

Наступне означення дає поняття над-перетворень, збираючи в одне поняття всі конструктори типу даних. І таке поняття назовемо функтором.

Визначення 5. Функтор T між двома категоріями C і D , який позначається $T \in C \Rightarrow D$, складається з двох перетворень, а саме:

- перетворення об'єктів $A \in \text{obj } C$ в об'єкти $T A \in \text{obj } D$
- перетворення морфізмів $f \in A \xrightarrow{C} B$ в морфізми $T f \in T A \xrightarrow{D} T B$, такі що, $T \text{id} = \text{id}$ та $T (f * g) = T f * T g$

Зазначимо, що пошук аргументу ($\text{id } T$) перевизначено для того, щоб врахувати перетворення і над типами, і над функціями.

Очевидно, визначаючи функтор, ми намагалися досягти того, що в поліморфічних типах даних, де вигляд типу визначає параметр, вигляд цього типу даних буде змінюватися в залежності від застосованого над ними функціоналу, а перетворення функціоналів за допомогою функтора може одночасно переслідувати й ціль незмінності структури даних. Наприклад, список однаково задає граф і дерево, але в залежності від того, як функція буде обробляти цю структуру даних, відповідно, отримаємо різні варіанти. Тому, у цьому випадку важливо, щоб функтор перетворив функціонал в залежності від конкретного типу даних у структурі.

Отже, використовуючи теорію категорій, і, знайшовши розв'язок (алгоритм, програму) для однієї задачі, ми можемо розв'язати цілий клас задач, якщо знайдемо і застосуємо відповідні функтори.

Висновки

Основна ідея програмування за допомогою скелетонів — це поділ паралельного програмування на рівень реалізацій та застосування. Вважається, що програміст визначить структуру майбутньої програми заздалегідь, а основна реалізація та взаємодія блоків, з яких вона складається, буде реалізована в компіляторах чи бібліотеках. Звичайно, у таких пошуках конструктивного алгоритмування не обійтись без формалізмів і теорем, що і були розглянуті у цій праці.

Ми дослідили основу паралельного програмування на основі скелетонів паралельних даних, а саме:

- розглянули тип даних список, побудували на ньому скелетони паралельних типів даних як функції високих порядків, розглянули операції над цими функціями, базуючись на алгебрі списків
- визначили клас скелетонів, які назвали катаморфізмами, що добре застосовні при паралельній реалізації
- користуючись теорією категорій, отримали узагальнення і визначення структури скелетонів

Скелетони — це готові алгоритмічні форми, з яких будуються паралельні програми. Кожен скелетон має готову реалізацію, таким чином

непотрібно щоразу адаптувати програму під конкретне застосування. Основною перевагою скелетонів є те, що вони поєднують взаємодію (координацію) і розпаралелювання на рівні даних, на високому рівні абстракції [5].

Використання такої абстракції як скелетон — це пошук загальних схем побудови паралельних програм, які за допомогою формальних математичних схем і загальних класифікованих схем-наборів розв'язання типових проблем дозволяють легко створювати паралельні програми для класів задач.

Програмування за допомогою скелетонів — це узагальнена форма паралельного програмування з багатьма варіантами. Основною

перевагою скелетонів є те, що, написавши програму один раз, можна розв'язати таким чином цілий клас задач, і не потрібно писати кожного разу код для реалізації часткового випадку однієї задачі під конкретну розподілену систему. Використання такої абстракції як скелетон — це пошук загальних схем побудови паралельних програм. Як результат, маємо математичні формальні схеми і загальні класифіковані схеми-набори розв'язання типових модельних задач, що легко програмуються.

Автор висловлює подяку Сергію Горлачу за підтримку мотивації цього дослідження, також Семену Гороховському за критику й допомогу в написанні цієї статті.

Systems — 1994. — Vol. 16, Issue 3, Parallel programming with control abstraction — pp. 524-576

4. Jörg Fischer, Sergei Gorlatch, and Holger Bischof. Foundations of data-parallel skeletons // In Patterns and Skeletons for Parallel and Distributed Computing. — Springer-Verlag. — 2003. — pp. 1-27.

5. Ricardo Pena, Fernando Rubio. Parallel functional programming at two levels of abstraction // Proceedings of the 3rd ACM SIGPLAN international conference on Principles and practice of declarative programming. — Florence, Italy. — 2001. — pp. 187-198.

Список використаних джерел

1. Fethi A. Rabhi, Sergei Gorlatch, editors. Patterns and Skeletons for Parallel and Distributed Computing // Springer-Verlag. — 2003. — 450 pp.
2. Marco Aldinucci, Marco Danelutto. Skeleton-based parallel programming: Functional and parallel semantics in a single shot // Computer Languages, Systems & Structures. — 2007. — Vol. 33, Issues 3-4, Semantics and Cost Models for Cost Models for High-level Parallel Programming. — pp. 179-192.
3. Lawrence A. Crowl, Thomas J. LeBlanc. ACM Transactions on Programming Languages and

Надійшла до редколегії 15.12.10

УДК 681.3.06

Лавріщева К.М., д.-р ф.-м. н., проф.,
Слабоспицька О.О., к.ф.-м.н.,
Колесник А.Л., Коваль Г.І., к.ф.-м.н.

Теоретичні аспекти керування варіабельністю в сімействах програмних систем

Визначено вимоги до керування варіабельністю – здатністю артефактів сімейства програмних систем (СПС) бути зміненими, налаштованими чи конфігурованими для застосування в певному контексті. На підтримку вимог запропоновано узгоджені моделі варіабельності в структурі й артефактах СПС. Вона доповнює її відомі кількісні виміри – змінні артефакти та їх варіанти – виміром кількісних оцінок (рівня і ступеню відповідності потребам зацікавлених сторін), які обґрунтовують управлінські рішення з розроблення СПС. Для вірогідності оцінок універсальні моделі – мережу Байєса, Аналітичну ієрархію, Дерево цінності – узгоджено згідно з ситуацією оцінювання. Описано підтримку визначених вимог функціями керування варіабельністю, запровадженими в модельному середовищі.

Ключові слова: варіабельність, трасовність, сімейство програмних систем, керування варіабельністю, модель оцінювання.

*E-mail: slabospickaya@mail.ru, gagaips@ukr.net
Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Потреби сучасного ринку програмної продукції у різних предметних областях (ПрО) стимулюють організації-розробники до швидкого створення та випуску на ринок високоякісних програмних систем (ПС) і вчасного реагування на зміни вподобань замовників, появу нових середовищ (Web, Grid), архітектурних платформ тощо. Це, в свою чергу, ставить перед програмною інженерією задачі вдосконалення наявних і створення нових методологій розроблення ПС. Одна з новітніх методологій ґрунтується на ідеї побудови сімейств функціонально подібних ПС (СПС) для окремих ПрО. СПС – це сукупність ПС, які мають спільну множину характеристик, відповідних потребам певних функціональних сегментів ПрО, розрізняються способами втілення цих характеристик в окремих ПС і розробляються з використанням готових ресурсів.

K.M. Lavrischeva, Dr. Sci. (Phys.-Math.), prof.,
O.A. Slabospickaya, Cand. Sci. (Phys.-Math.),
A.L.Kolesnik, G.I.Koval, Cand. Sci. (Phys.-Math.)

The Theoretical View for Software Family Variability Management

The demands are elaborated for the Management of Variability being seen as the ability of software system or artifact from Software Family (SF) to be changed, customized or configured for use in particular context. Consistent models are proposed for the demands support, namely for the links of variability both in SF structure and core assets and also the dedicated Variability model. It expands well-known qualitative variability measures – such as variation points and its variants – with quantitative estimates (both for variability level and relevance as to stakeholders' needs) that substantiate the managerial decisions as to SF. The universal models, namely Bayesian Net, Analytical Hierarchy and Value Tree are configured accordingly to the assessment situation for the estimates' consistency. The above demands support is described with the Variability Management functions being introduced in the model environment being proposed.

Key Words: software family, variability, traceability, variability management, assessment model.

Програмні системи, члени СПС, не створюються «з нуля», а породжуються на підставі загальної моделі СПС, яка конкретизується згідно зі специфікацією вимог до ПС [1–3]. Ресурси СПС накопичуються в репозиторіях інтегрованого середовища його розроблення (ICP) і обиратися, адаптуватися та складатися разом під час створення ПС.

Ключовою проблемою інженерії сімейства ПС, як специфічного об'єкту програмної інженерії, є забезпечення належної варіабельності (змінюваності) усіх артефактів процесу розроблення як сімейства, так і його членів. Це підвищує їх повторну використовність та чутливість до передбачуваних змін ділових процесів, ПрО та умов середовища функціонування, оскільки розширює контекст застосовності ПС – членів СПС. Але водночас зростає складність ICP та, як наслідок, тривалість і вартість складання ПС. Дотримання балансу між очікуваними

перевагами та витратами від реалізації тих чи інших варіантів рішень з варіабельності обумовлює потребу керування нею. Однак, декларуючи підтримку такого керування [4], переважна більшість дослідників зосереджуються на окремих технологічних і реалізаційних аспектах забезпечення варіабельності [5, 6], залишаючи поза увагою проблему обґрунтованого кількісного оцінювання та моніторингу її рівня.

На підтримку вирішення цієї проблеми в роботі пропонується вдосконалення методології розроблення СПС [4] теоретичним апаратом узгодженого моделювання варіабельності в структурі та артефактах СПС, а також функціями обґрунтованого керування варіабельністю у пропонованому модельному середовищі.

Проблема досліджується в рамках фундаментального проекту Інституту програмних систем НАН України «Розробка теоретичного фундаменту генеруючого програмування та інструментальних засобів його підтримки» [1].

Середовище керування варіабельністю

Визначальною особливістю процесу розроблення СПС для певної ПрО є наявність у ньому двох складників – процесів інженерії ПрО та інженерії програмних застосунків (ПЗ) (рис. 1).

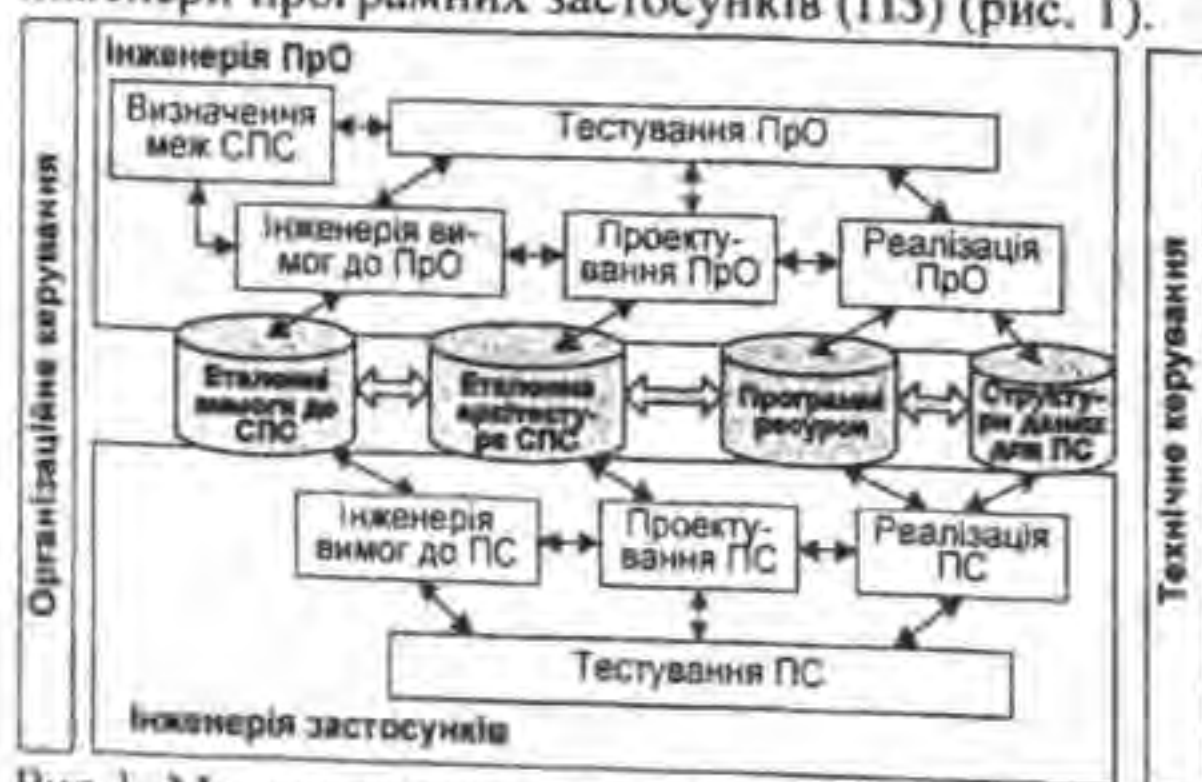


Рис. 1. Модель процесу розроблення СПС

Процес інженерії ПрО призначений для визначення спільних і відмітних особливостей програмних систем у складі СПС з позицій функціональних і нефункціональних вимог до них (їх якості, архітектури, середовища функціонування тощо) та створення артефактів, які є реалізацією спільних характеристик ПЗ та забезпечують варіабельність СПС, необхідну для породження множини ПЗ за вимогами фахівців у ПрО. Артефакти у ПрО, що є ресурсами повторного використання (РПВ), піддаються тестуванню і утворюють платформу СПС. Ця платформа поряд з програмними компонентами включає моделі вимог, архітектури, тестів тощо. При побудові систем оброблення даних,

характерними ознаками яких є складні структури даних у ПрО і взаємодія з СКБД для ведення великих об'ємів інформації, до складу платформи входять також моделі даних, якими маніпулюють ПЗ – члени СПС.

Процес інженерії застосунків призначений для породження ПЗ на загальній платформі з множини РПВ шляхом зв'язування (усунення) закладеної в них варіабельності відповідно до вимог щодо кожної ПЗ.

Процеси виконуються паралельно, взаємодіють через робочі продукти цих процесів у репозиторії ІСР, якими вони обмінюються в узгоджений спосіб.

Процеси організаційного керування необхідні для координації робіт за процесами інженерії ПрО та інженерії застосунків, а процеси технічного керування – для планування і контролю робіт зі створення РПВ, з одного боку, та використання цих РПВ під час побудови ПЗ, з іншого боку.

Центральне місце серед процесів технічного керування займає керування варіабельністю.

Функції керування варіабельністю

Варіабельність – здатність сімейства ПЗ, окремої системи або артефакту до розширення, змінювання, пристосування або конфігурування з метою використання у конкретній ПрО [4].

Варіабельність має забезпечуватися на рівнях вимог до ПЗ, моделі характеристик СПС, архітектури, коду, документації, тестів тощо. Загалом, вона може бути реалізована як у СПС, так і в конкретних ПЗ. У першому випадку забезпечується варіабельність виробничої лінії, яка підтримує існування сімейства як множини ПЗ. Вона стосується множини базових елементів будь-якої природи (програмних і не програмних), які містяться у репозиторії ІСР. У другому випадку забезпечується варіабельність складових ПЗ (включаючи документацію), що обумовлює еволюційність ПЗ після її породження і «відлучення» від СПС. В обох випадках реалізація варіабельності спрямована на забезпечення життєздатності СПС і ПЗ.

Об'єктами керування варіабельністю і, водночас, її проявами в СПС, є:

- точка варіантності, або варіантний артефакт, – формальне подання елементарного артефакту СПС певного типу, способи реалізації якого є різними для різних ПЗ;

- варіант – елементарний артефакт СПС одного типу з відповідним йому варіантним артефактом, який являє собою його подання у певній ПЗ;

– обмеження – предикат, визначений на декартовому добутку декартових квадратів множин точок варіантності та варіантів;

– залежність – предикат, що визначає припустимі взаємозв'язки між точкою варіантності і множиною варіантів для неї.

Адаптація відомого циклу керування Е.Дьюмінга [7] до сучасних уявлень про ефективну організацію програмної індустрії [8], дозволяє виокремити у процесі розроблення СПС чотири функції кількісного керування варіабельністю:

1) планування реалізації варіабельності в артефактах СПС (на рівнях інженерії Про та інженерії застосунків [9]) (F_1);

2) реалізація варіабельності в артефактах СПС [6] (F_2);

3) системний моніторинг задовільності стану СПС в аспекті варіабельності (F_3);

4) актуалізація СПС за результатами моніторингу (F_4).

Для уніфікації керування варіабельністю у процесах інженерії Про та інженерії застосунків, доцільно прийняти такі вимоги до виконання функцій $F_1 - F_4$:

1) обґрунтованість – наявність об'єктивних підстав прийняття рішень щодо $F_1 - F_4$ (D_1);

2) узгодженість – однаковість способу вироблення й реалізації рішень на всіх рівнях абстракції та на всіх етапах процесу розроблення СПС (D_2);

3) масштабованість – незалежність способу вироблення й реалізації цих рішень від обсягу функціональних можливостей, охоплених СПС (D_3);

4) трасовність – можливість відстеження зв'язків між проявами варіабельності на всіх рівнях абстракції та на всіх етапах процесу розроблення СПС (D_4);

5) візуалізованість – зазначених проявів варіабельності та зв'язків між ними (D_5).

Забезпечення сформульованих вимог $D_1 - D_5$ для функцій $F_1 - F_4$ потребує адекватного модельного середовища їх реалізації.

Необхідним елементом цього середовища має бути спеціальна інтегрована модель варіабельності в СПС (VM), яка дозволить, за даними про поточну або прогнозовану структуру СПС, визначати оцінку рівня варіабельності та ступеня її відповідності потребам Про за відносною шкалою. Але свою роль з підтримки функцій $F_1 - F_4$ – надання оцінок варіабельності в ролі підстави рішень з їх виконання – модель VM зможе виконувати тільки разом із адекватною моделлю самого СПС – джерела зазначених даних.

Відомі два типи підходів до моделювання варіабельності. За одним з них пропонується інтегрувати варіабельність в існуючі моделі (як-от FD або UML), за іншим – будувати спеціальні моделі варіабельності.

Моделі варіабельності [10], які пропонуються наразі у підходах першого типу, обмежено придатні тільки для функції F_3 – перевірки відповідності структури СПС (в аспекті варіабельності) потребам Про [9]. До того ж, вони використовують принципово незіставні нотації для подання варіабельності артефактів СПС різних типів, що істотно ускладнює забезпечення вимог узгодженості та трасовності. Серед спеціальних моделей варіабельності, єдиною відомою авторам моделлю, практично позбавленою останнього недоліку і тому обраною для перетворення у VM, є так звана ортогональна модель варіабельності (OVM) [4].

Відповідна їй модель СПС має узгоджено подавати склад і взаємозв'язки всіх проявів варіабельності – як у структурі СПС, так і в усіх артефактах процесів інженерії Про та інженерії застосунків, зіставляючи їх із проявами спільних властивостей ПС з СПС.

Відсутність наразі моделей СПС із зазначеною особливістю обумовлює доцільність розроблення її оригінальної версії.

Таким чином, пропонується внести до складу модельного середовища для функцій $F_1 - F_4$:

– тривимірну модель варіабельності, яка зіставляє вимірам її ортогональної моделі третій вимір – вкладену оціночну модель;

– модель варіабельності в СПС із визначеною вище особливістю.

Узгоджені моделі варіабельності в структурі та артефактах СПС

Аналіз внутрішньої структури підпроцесів інженерії Про та інженерії застосунків за моделлю Похла [4] висвітлює чотири типи (t) їх елементарних артефактів, а саме: характеристика ПС, передбачена до реалізації ($t=1$), компонент архітектури ПС ($t=2$), програмний артефакт – (компонент, аспект тощо) ($t=3$), таблиця бази даних ($t=4$). Зазначимо, що артефакти типу $t=1$ поділяються на дві групи: враховані потреби ділових процесів Про та вимоги до ПС з боку аналітиків СПС.

Модель структури СПС має відповідати таким вимогам:

– бути придатною для подання артефактів у різних взаємопов'язаних функціональних сегментах Про;

– бути придатною для подання артефактів (ресурсів) СПС відповідно до сфери

відповідальності учасників розроблення СПС (як мінімум, замовників, аналітиків у ПрО, бізнес-аналітиків, архітекторів, програмістів);

– включати нотації, прийнятні для моделювання артефактів СПС розробниками СПС відповідно до типу артефактів (як мінімум, характеристик ПрО, вимог до ПС, архітектури СПС, збірки з компонентів, сховищ даних тощо);

– визначати постійні (властиві всім ПС) і розбіжні (властиві окремим ПС) характеристики

$$SV = \langle \langle CF; \langle DR, TC \rangle; \langle AR, TD \rangle; \langle CM, FR, TS, TA \rangle; \langle ER, TF \rangle; Constr; Dep \rangle, \quad (1)$$

де $CF = \langle SF, LF \rangle$ – діаграма характеристик (у поширеній нотації [11]), яка подає множину сталих і змінних властивостей ПС (SF), пропорованих їх потенційним споживачам у ПрО, та обмеження ($Constr$) і залежності (Dep) між ними за допомогою зв'язків обов'язкового і варіантного підпорядкування (LF);

$DR = \langle SF \cup SR, LF \cup LR \rangle$ – аналогічна діаграма, де характеристики $f \in SF$ деталізовано і/або доповнено характеристиками $r \in SR$, що подають сталі та змінні "технічні" вимоги до ПС з боку аналітиків СПС, обмеження й залежності яких подаються зв'язками LR ;

$TC = \{(r, f), r \in SR, f \in SF\}$ – двосторонні зв'язки трасовності між "технічними" вимогами до ПС та їх властивостями для споживача, які конкретизуються вимогами;

$AR = \langle AC, LC \rangle$ – еталонна архітектура СПС [4], тобто опис взаємозв'язків (LC) між формальними поданнями програмних компонентів (AC) для реалізації характеристик $f \in DR$ у певній нотації для програмної архітектури (UML, xADL тощо);

$TD = \{(ac, f), ac \in AC, f \in SF \cup SR\}$ – двосторонні зв'язки трасовності між поданнями компонентів $ac \in AC$ для реалізації характеристик $f \in DR$ та цими характеристиками;

CM – формальні подання базових програмних компонентів для реалізації компонентів архітектури при складанні за допомогою елементів компонентного каркаса (FR) [12];

$TS = \{(ts, cm), cm \in CM\}$ – формальні описи тестів (ts) для компонентів з поданнями cm ;

$TA = \{(cm, ac), cm \in CM, ac \in AC\}$ – двосторонні зв'язки трасовності між програмними компонентами та компонентами архітектури AR ;

$TL = \{(fr, lc), fr \in FR, lc \in LC\}$ – двосторонні

на всіх рівнях абстракції і впродовж всіх етапів їх подання в артефактах процесу розроблення СПС та відображувати обмеження й залежності щодо цих артефактів.

Ієрархію рівнів варіабельності артефактів СПС, відповідних типам $t=1, \dots, 4$, встановлює

Означення 1. Модель варіабельності в структурі СПС – це кортеж

зв'язки трасовності між елементами каркаса для складання компонентів, та взаємозв'язками компонентів архітектури;

ER і TF – відповідно, ER-модель БД для оброблення ПС з СПС, та двосторонні зв'язки трасовності її елементів з компонентами $cm \in CM$.

Взаємозв'язки проявів варіабельності в структурі СПС за моделлю FV (1) показано на рис. 2а).

Кожна з наведених рис. 2а п'яти "горизонтальних" площин відображає подання потреб потенційних споживачів, формоване у відповідному підпроцесі розроблення СПС (див. рис. 1) в артефактах відповідного типу – від характеристик потреб ПрО до підмножин полів таблиць БД.

Варіантність цих артефактів має два аспекти: розбіжність варіантів їх втілення в ПС для певного функціонального сегменту ПрО і в ПС для різних функціональних сегментів. Два незалежні напрями долучення вершин до формальних подань варіантності – графів $DR, AR, (FR, CM), ER$, відповідні аспектам варіантності, – утворюють виміри "горизонтальних" площин на рис. 2а).

Третім виміром моделі (1) є трасовність артефактів суміжних рівнів, реалізована зв'язками з множин TD, TA, TF і подана на рис. 2а) вертикальними стрілками.

Зауваження 1. У моделі структури СПС (1) зв'язки зі складу множин TF, TA, TD реалізують неперервну трасовність довільного артефакту СПС типу $t > 1$ знизу догори, зіставляючи йому хоча б один артефакт-відповідник кожного з типів i^* ($1 \leq i^* < t$), для реалізації (безпосередньої або опосередкованої) якого призначено трасований артефакт.

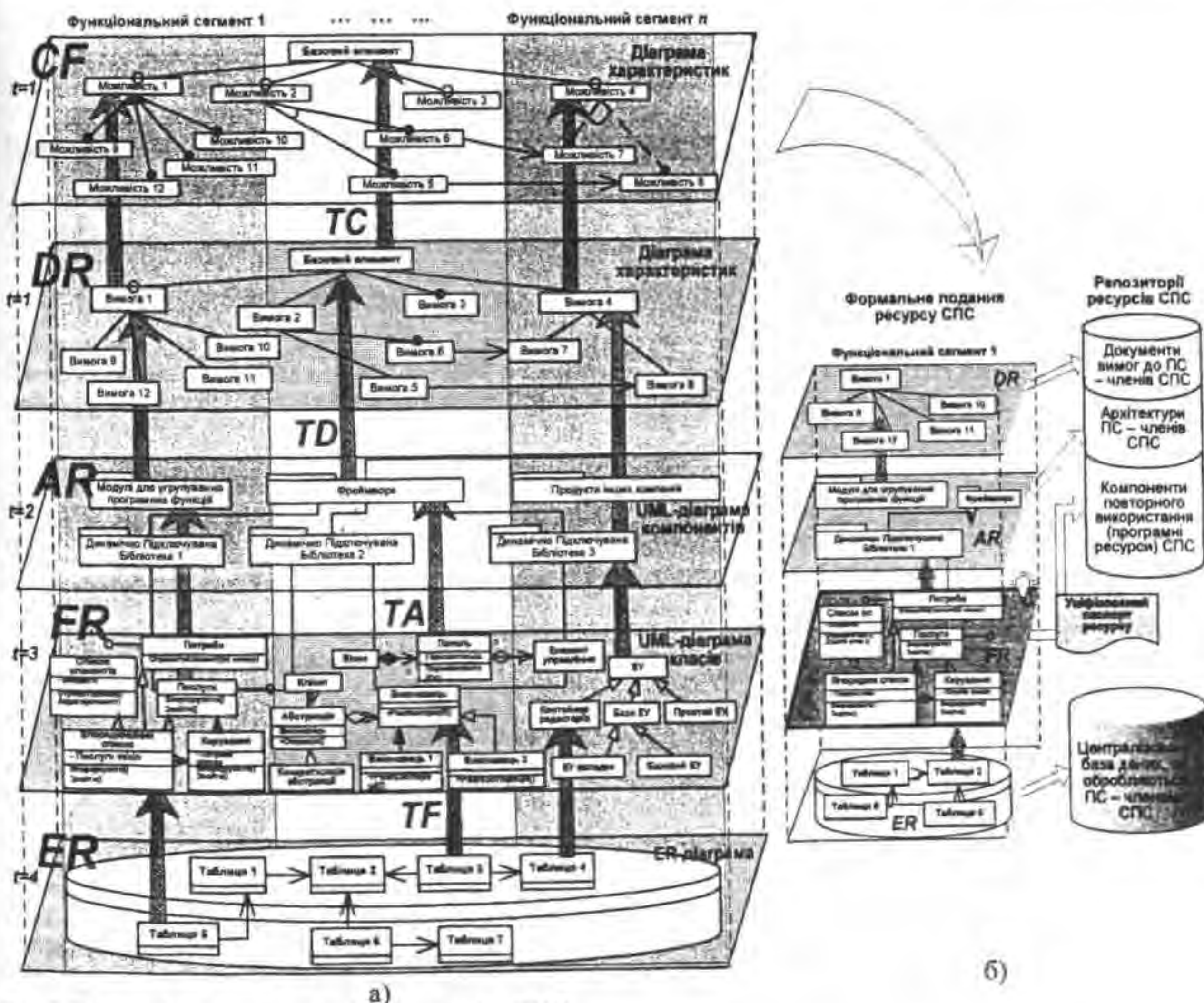


Рис.2 Тривимірна модель варіабельності в СПС

Зауваження 2. У моделі структури СПС (1) може існувати артефакт СПС довільного типу $t=1,2,3$, для якого не визначено жодного артефакту-відповідника типу t_* ($t < t_* \leq 4$) за зв'язками зі складу множин TD , TA , TF , призначеного для його реалізації (безпосередньої або опосередкованої).

Такий артефакт зветься надалі нетрасовним (якщо $t_* = t + 1$) або нетрасовним на рівні t_* .

Із сформульованих зауважень випливає неможливість появи на будь-якому рівні моделі (1) вершин, не підпорядкованих певній характеристиці $f \in SF \cup SR$, а також дублювання вершин (за необхідності, долучаються нові зв'язки трасовності). Це означає неможливість дублювання артефактів у процесі розроблення СПС та створення "зайвих" артефактів, не призначених для реалізації в ПС певних характеристик з меж СПС (див. рис. 2).

Співвідношення між варіабельністю в структурі СПС та в його артефактах, зокрема в

ресурсах, висвітлює

Означення 2. Модель варіабельності в артефактах СПС – це трійка

$$FA = \langle CAS; REP; DBF \rangle; CAS = \bigcup_{i=1, \dots, 4} AF_i; \quad (2)$$

$$REP = \langle RPR \cup RPA \cup RPE \rangle; RPE = CM' \cup TS', \quad (3)$$

де $AF_i = \{af_i\}$ – формальні подання артефактів СПС типу $t=1, \dots, 4$ (af_i);

RPR , RPA , RPE – репозиторії ресурсів СПС типу t : документів вимог до ПС ($t=1$), типових архітектур ПС ($t=2$) та програмних ресурсів ($t=3$);

DBF – фізичне подання БД для оброблення ПС – членами СПС;

CM' , TS' – множини програмних компонентів повторного використання, описаних у CM в (1), та тестів для них.

Для узгодження подання варіабельності в артефактах СПС та в структурі СПС за моделлю (1) пропонується уніфікований розгляд довільного артефакту типу t (af_i) як наскрізного фрагмента моделі (1), виокремленого

неперервними зв'язками трасовності (з множин $TD \cup TA \cup TF$) вершин певного "цільового" підграфу r_i графу DR – тих характеристик ПС, для реалізації яких призначено af_i . На рис. 26) наведено таке подання програмного артефакту (af_3) для реалізації характеристик з Функціонального сегменту 1 ПрО для СПС. Відповідний їм підграф елементів каркасу і програмних компонентів, що безпосередньо реалізується артефактом af_3 , виділено на рис. 26) сірим кольором (для af_3 він являє собою елемент $\langle fr_3, cm_3 \rangle$ з його подання (4)).

Запропонований розгляд артефактів фіксує

Означення 3. Формальне подання артефакту СПС типу t (af_t) – це кортеж

$$f(af_t) = \langle r_t; \langle ar_t, td_t \rangle; \langle fr_t, cm_t, ts_t, ta_t \rangle; \langle er_t, tf_t \rangle \rangle, \quad (4)$$

де складниками є підграфи "горизонтальних" графів моделі (1) на всіх її рівнях та "вертикальні" зв'язки трасовності між вершинами цих підграфів суміжних рівнів.

Зокрема, у виразі (4):

r_t – підграф DR , що подає характеристики, для реалізації яких призначено артефакт af_t ;

ar_t – підграф графу AR , де вершини пов'язані неперервними зв'язками трасовності: $td_2 \subseteq TD$ – з вершинами графу r_2 або ж, коли $t \neq 2$,

$$tr \subseteq \begin{cases} TF \cup TA, t > 2 \\ TD, t = 1 \end{cases}$$

– з вершинами підграфу gr_t рівня t з моделі (1), який безпосередньо реалізується артефактом af_t у процесі розроблення СПС;

$$gr_1 = r_1, gr_2 = ar_2; gr_3 = \langle fr_3, cm_3 \rangle; gr_4 = er_4;$$

$td_1 \subseteq TD$ – зв'язки трасовності між вершинами підграфів ar_1 і r_1 ;

$cm_t \subseteq CM$, $fr_t \subseteq FR$ – формальні подання програмних компонентів та елементів каркаса їх складання, пов'язаних неперервними зв'язками трасовності: $ta_3 \subseteq TA$ – з вершинами підграфу ar_3 ;

$$tr \subseteq \begin{cases} TA \cup TD, t = 1 \\ TA, t = 2 \\ TF, t = 4 \end{cases}$$

– з вершинами підграфу gr_t ($t \neq 3$);

$ts_t \subseteq TS$ – формальні описи тестів для компонентів, описаних в cm_t ;

$ta_t \subseteq TA$ – зв'язки трасовності між вершинами підграфів $\langle fr_t, cm_t \rangle$ і ar_t ;

er_t – фрагмент ER-моделі БД для оброблення ПС з СПС, де вершини пов'язані висхідними неперервними зв'язками трасовності: $tf_4 \subseteq TF$ – з

вершинами підграфу $\langle cm_3, fr_3 \rangle$; $tr \subseteq TF \cup TA \cup TD$ – з вершинами підграфу gr_t ($t \neq 4$);

$tf_t \subseteq TF$ – зв'язки трасовності між вершинами підграфів er_t і $\langle fr_t, cm_t \rangle$.

Зауваження 3. Елементарний артефакт, нетрасовний на певному рівні в моделі FV (1), є нетрасовним на тому ж рівні і в усіх поданнях артефактів СПС типу t $f(as_t)$ (4), до яких належить. Однак артефакт, трасований в (1), може стати нетрасовним (на певному рівні t) у поданні (4) деякого артефакту as_t . Тоді підмножини його потенційних відповідників на позиціях $t+1, \dots, 4$ у поданні (4) пусті.

За умов повторного використання артефакту af_t в ролі ресурсу СПС, для постійного відстеження позиції af_t на підтримку функцій F_1 – F_4 доцільно зберігати його у відповідному репозиторії (див. (3)) разом з уніфікованим паспортом (як показано на рис. 26)).

Пропоновану структуру паспорту визначає

Означення 4. Уніфікований паспорт ресурсу $rs \in CAS$ (2), (4) – це поповнюваний кортеж

$$CT(as) = \langle wc; \{ \langle wu_t, us_t, vb_t, r_t \rangle \}; id \rangle, \quad (5)$$

де wc , wu_t – відповідно, трудовитрати на створення rs та середні трудовитрати на його повторне використання протягом чергового періоду t розроблення СПС;

$us_t = \langle fc, fh, ph \rangle$ – використовність rs протягом періоду t , складена частотами: безпосереднього надання rs замовнику/споживачу (fc), повторного застосування в проактивному розробленні ПС ринкового призначення (fh) та для вдосконалення самого процесу розроблення СПС (ph);

$vb_t = \langle fvp, fvr \rangle$ – варіабельність rs протягом періоду t , складена частотою точок варіантності (fvp) і середньою частотою варіантів для них (fvr) згідно з $f(rs)$ (4);

r_t – рейтинг корисності rs у період t , експертно коригований аналітиками СПС на підставі апріорного значення у впорядкуванні ресурсів того ж типу, що й rs , за спаданням величин $(fc+fh+ph)/(wc+wu_t)$ для них.

id – реєстраційні реквізити (ідентифікатор, дата реєстрації в репозиторії, персоналії й роль її суб'єкта).

Інтегрована модель варіабельності

Для забезпечення сформульованих вимог D_1 – D_4 стосовно функцій F_1 – F_4 , принцип побудови ортогональної моделі варіабельності [4] застосовано до розробленої моделі варіабельності в структурі СПС. Саме, з моделей SV (1) і AV (2)–(4) вилучено вершини всіх рівнів, які подають

спільні риси всіх ПС з СПС (тобто вершини, не пов'язані у відповідних графах жодними зв'язками варіантного підпорядкування). Отримані в результаті двовимірні графи розподілено на множини всіх точок варіантності в СПС (VP) та варіантів для них (VAR) разом із збереженими зв'язками трасовності з множин TF, TA, TD (див. (1)). У цих множинах виокремлено "фрагменти", відповідні типам відображуваних артефактів – рівням моделей SV (1). Для додаткової підтримки вимоги обґрунтованості рішень з керування варіабельністю, кожному з отриманих "блоків" зіставлений третій вимір кількісних оцінок варіабельності (на рівні, відповідному блоку згідно з моделлю SV (1)) за деякою шкалою відношень, дані для яких має надавати цей блок. Враховуючи необхідність постійного дотримання балансу між структурованістю множини точок варіантності й варіантів в СПС та труднощами ведення відповідного ІСР, що зафіксований на початку роботи, передбачено два класи оцінок варіабельності СПС – її рівні та рівні її відповідності потребам СПС.

Співвідношення між окресленими вимірами, поєднаними в отриманому гібридному конструкті, названому інтегрованою моделлю варіабельності, показано на рис. 3.



Рис.3 Склад інтегрованої моделі варіабельності

Його формальне подання надає

Означення 5. Інтегрована модель варіабельності – це структурований кортеж

$$VM = \langle EVM, VP, VAR \rangle; EVM = \langle VL, VR \rangle; (6)$$

$$VP = \langle VpR, VpA, VpE, VpB \rangle;$$

$$VAR = \langle VR, VA, VE, VB \rangle;$$

$$VL = \langle vrl, val, vel, vbl \rangle, VR = \langle vrr, var, ver, vbr \rangle, (7)$$

де VpR, VpA, VpE, VpB – множини точок варіантності в структурі СПС, які подають, відповідно, характеристики ПС, компоненти архітектури, елементи каркаса і поля таблиць БД

(VpB), із збереженими зв'язками трасовності, обмеження ($Constr$) й залежності (Dep) з (1);

VR, VA, VE, VB – множини варіантів для точок варіантності з множин, відповідних їм за номером у (6), на яких також збережено ими зв'язками трасовності, обмеження ($Constr$) й залежності (Dep) з (1);

VL – вкладена оціночна модель рівня варіабельності, деталізована за типами її проявів – відповідно, у вимогах (vrl), компонентах архітектури (val), програмних артефактах (vel), оброблюваних даних (vbl);

VR – аналогічна оціночна модель рівня відповідності варіабельності в СПС потребам Про зі сфери охоплення СПС, деталізована відповідно для вимог (vrr), компонентів архітектури (var), програмних артефактів (ver), оброблюваних даних (vbr).

Для подання фрагментів (7) вкладеної оціночної моделі пропонуються три універсальні моделі переваг. Вибір серед них визначається поточною ситуацією оцінювання варіабельності в процесі розроблення СПС.

Зокрема, індивідуальне стохастичне прогнозування варіабельності за високого ступеню невизначеності її чинників (на початку етапів процесу інженерії Про та інженерії за стосунків (див. рис. 1)), найбільш доцільне за допомогою Байєсівської мережі. Для індивідуального або колективного вибору оптимального серед кількох варіантів рішень з планування або реалізації варіабельності на цих етапах найбільш придатна аналітична ієрархія Т.Сааті. Нарешті, колективне оцінювання дійсного рівня варіабельності та ступеню її відповідності потребам Про наприкінці етапів процесу інженерії Про та інженерії застосунків може ефективно виконуватися за допомогою дерева цінності. Для збереження спадкоємності отримуваних оцінок у процесі розроблення СПС передбачається узгодження зазначених моделей шляхом зіставлення їм одного набору термінальних вершин. Орієнтовний їх склад відповідає структурі уніфікованого паспорту ресурсу $CT(rs)$ (5), який і надає фактичні дані для їх оцінювання.

Оцінки рівнів варіабельності за пропонованими моделями VL і VR (7) зручно накопичувати у спеціальному профілі варіабельності СПС. Його структуру фіксує

Означення 6. Профіль варіабельності СПС – це структурований кортеж

$$VPR = \{ \langle WU, US, VB, R \rangle \},$$

де компоненти мають зміст, аналогічний компонентам уніфікованого паспорту ресурсу

$CT(rs)$ (5), а саме:

WC і WU_t – відповідно, трудовитрати на створення СПС та середні трудовитрати на створення ПС для споживачів протягом чергового періоду t розроблення СПС;

US_t – середня використовність артефактів СПС протягом періоду t , складена усередненими частотами їх безпосереднього надання замовнику/споживачу, повторного застосування в проактивному розробленні ПС ринкового призначення та для вдосконалення самого процесу розроблення СПС;

VB_t – показник варіабельності СПС протягом періоду t , складений усередненими значеннями відповідних показників з паспортів ресурсів СПС;

R_t – п'ятірки артефактів з найвищим та найнижчим рейтингом у період t за їх паспортами $CT(rs)$ (5).

Висновки

Вперше в інженерії СПС розроблено теорію моделювання варіабельності в СПС.

Запропоновані модельне середовище та функції керування варіабельністю СПС з урахуванням вимог узгодженості, масштабованості, трасовності й візуалізованості складають ядро цілісного процесу обґрунтованого кількісного керування варіабельністю.

Поповнення ортогональної моделі варіабельності оціночною моделлю в її інтегрованій моделі надає можливість накопичення оцінок варіабельності за різних змістовних ситуацій при розробленні СПС та вироблення формальних описів цих ситуацій у термінах значень рівня варіабельності та ступеню її відповідності потребам Про.

Модельне середовище керування варіабельністю дає можливість формулювати і розв'язувати оптимізаційні задачі ресурсно-ефективного планування розроблення СПС – розподілу коштів і термінів за обмежень на значення критеріїв оціночної моделі варіабельності.

Надані моделі складають підґрунтя для врахування обмежень та залежностей між проявами варіабельності в процесі обґрунтованого кількісного керування нею.

Список використаних джерел

1. Лавріщева К.М. Генерувальне програмування програмних систем і сімейств // Проблеми програмування. – 2009. – № 1. – С. 3-16.
2. Чернецки К., Айзенекер У. Порождающее программирование. Методы, инструменты, применение. – Издательский дом Питер. – М. – СПб. – Харьков. – Минск. – 2005. – 730 с.
3. Лавріщева К.М., Коваль Г.І., Слабостицька О.О., Колесник А.Л. Особливості процесів керування при створенні сімейств програмних систем // Проблеми програмування. – 2009. – №3. – С.40-49.
4. Pohl K., Böckle G., Linden F.J. Software Product Line Engineering: Foundations, Principles and Techniques. – New York: Springer-Verlag. – 2005. – 437 p.
5. Berg K., Bishop J., Muthig D. Tracing Software Product Line Variability – From Problem to Solution Space // in Proc. of SAICSIT'2005, P. 111-120.
6. Колесник А.Л. Механізми забезпечення варіабельності в сімействах програмних систем // Проблеми програмування. – 2010. – №1. – С. 35-44.
7. Walton M. The Deming Management method // New York: Dodd, Mead. – 1986. – 262 p.
8. Андон П.І., Лавріщева К.М. Розвиток фабрик програм в інформаційному світі // Вісник НАН України. – 2010. – №10. – С.15-41.
9. Trinidad P., Benavides D., Ruiz-Cortés A. Improving Decision Making in Software Product Lines Product Plan Management // in: Proc. of the ADIS 2004 Workshop on Decision Support in Software Engineering, CEUR Workshop Proceedings. – 2004. – Vol. 120. – P. 51-58.
10. Sinnema M., Deelstra S. Classifying Variability Modeling Techniques // Elsevier J. on Inf. and Softw. Technology. – 2007. – 49. – P. 717-739.
11. Asikainen T., Männistö T., Soininen T. A Unified Conceptual Foundation for Feature Modelling // in Proc. of the 10th Int. Software Product Line Conf. (SPLC 2006). – 2006. – P. 31 – 40.
12. Грищенко В.Н., Лавріщева Е.М. Методы и средства компонентного программирования // Кибернетика и системный анализ. – 2003. – №1. – С. 39-55.

Надійшла до редколегії 15.12.10

УДК 004.774.6CMS

Лілікович С.О., Логінов А.В.

Аналіз систем шаблонізації сучасних CMS

У статті розглядаються питання використання систем управління шаблонами, принципи побудови шаблонізаторів. Мета статті – висвітлення рішень, які дозволяють підвищити швидкість роботи шаблонізатора.

Ключові слова: шаблонізатор, маркер, кешування.

S.O. Lilikovich, A.V. Loginov

Analysis of Template Engine Systems for Modern CMS

The issues of template engine systems' use are considered in the paper. The purpose of the paper is to describe the solutions which let to increase a speed of template engine's work.

Key words: template engine, marker, caching.

*E-mail: Livich@yandex.ru. Loginov_anatoly@mail.ru

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

У наш час більшість компаній володіють web-ресурсом. Незважаючи на відмінності у сферах діяльності настає момент, коли на підтримку сайту йде більше й більше часу, тому стійкою тенденцією останніх років є використання систем керування контентом (CMS), – програм, що дозволяють ефективно управляти змістом сайту [1]. Існує досить велика кількість літератури, присвяченої питанню створення сайтів за допомогою CMS, при цьому проблема зміни дизайну вже розробленого сайту на основі CMS системи висвітлена недостатньо, що робить розгляд даного питання актуальним.

Одним з рішень подібної проблеми є використання систем керування шаблонами: з їхньою допомогою можна чітко відокремлювати логіку від даних, використовуючи єдині шаблони.

Програми, що виконуються на web-серверах, зокрема, програми, написані мовою PHP, можуть містити в собі крім логіки, ще й HTML теги, різні конструкції, що не мають відношення до обробки інформації, вони тільки оформлюють її. Змішування логіки і представлення даних може призвести до негативних наслідків: програма стає незрозумілою іншим програмістам, ускладнюється робота дизайнера, уповільнюється швидкість роботи інтерпретатора PHP. У таких випадках, коли необхідно чітко відокремити представлення даних від логіки, використовуються системи керування шаблонами або просто «шаблонізатори» [2].

Таким чином, дизайнер може незалежно працювати з представленням даних, а програміст – з їхньою обробкою.

Простих статичних шаблонів буває недостатньо.

SMARTY – вважається своєрідним стандартом для систем керування шаблонами в широких колах програмістів. Однак є певні недоліки цього шаблонізатора, перш за все, громіздкість програмного рішення: великі вимоги до ресурсів серверу, занадто низька швидкість обробки.

Після аналізу принципів роботи популярних шаблонізаторів (SMARTY, Templier, HSTemplate) стає зрозуміло, що HTML шаблон для них перестає бути таким і переростає в набір інструкцій шаблонізатора упереміш із HTML кодом. Інструкції шаблонізатора прийнято називати маркерами. Маркери мають у кожному шаблонізаторі свій вид, наприклад «{\$var}» в SMARTY означає, що тут потрібно замість «{\$var}» підставити значення змінної var, яке повинне було бути передане SMARTY заздалегідь.

Більшість шаблонізаторів дозволяють також виконувати деякі інструкції в циклах, створювати умови, тобто мають обгортки до стандартних PHP-конструкцій. Через це в колах розробників широко поширена думка, що шаблонізатор у цілому – це обгортка для PHP, що робить його «безглуздим», адже шаблони можна реалізувати й на самому PHP (нативні шаблонізатори), включаючи їх до коду програми. Проте, шаблонізатори й сьогодні використовуються в багатьох веб-проектах, тому що використання PHP у шаблонах ускладнює роботу дизайнера, порушує MVC – класичну парадигму побудови систем даного роду.

Ідея розробки власної системи керування шаблонами виникла після використання шаблонізатора SMARTY в авторському проєкті Jaspen CMS [3-5]. Програмний продукт SMARTY не показав задовільних результатів: на обробку нескладного шаблону могло йти до чотирьох секунд (час обробки без кешування).

SMARTY має два головних недоліки: активне використання процесору (на деяких хостінгах, наприклад, на популярних в теперешній час – «хмарних», час CPU оплачується окремо), і швидкість роботи (іноді обробка доходить до 3-10 секунд).

Перевагою SMARTY є плагіни, тобто можливість легко розширити функціонал програми за рахунок доповнень.

Вимоги до систем подібного роду формуються продуктом, у якому планується використовувати систему керування шаблонами. Так, розроблений шаблонізатор JTE використовується в CMS системі, тобто в складній і вимогливій до швидкодії розробці.

У більш простих системах, де логіка в шаблонах не потрібна, або там, де потрібно тільки досить обмежений набір можливостей, можна використовувати й більш прості, а, виходить, і більш нересурсомісткі й не вузько спрямовані системи.

Метою розробки було створення швидкого й нересурсомісткого шаблонізатору з можливістю простого розширення функціоналу, ефективною системою кешування, з можливістю підстановки змінних, інших шаблонів і виклику підпрограм, причому сам шаблон повинен представляти собою HTML код із вставками псевдокоду. У результаті роботи була створена авторська система шаблонізації JTE 2.

Для забезпечення високої швидкості роботи було прийнято кілька рішень:

1. видалити обгортки стандартних функцій і конструкцій PHP (іноді вони можуть бути потрібні, але процес парсингу (розбору) таких обгортки — занадто ресурсомісткий процес);

2. застосування «сніпетів», які повинні замінити логіку в шаблоні. Сніпет представляє собою підпрограму шаблонізатора на PHP, яка виконується із заданими із шаблону параметрами. Сніпет виконується миттєво, відразу після його виклику. Приклад виклику: «`{[Testsniipp:var|name='var|value']}`»;

3. обробка шаблону повинна проводитися за мінімум етапів. Зараз для парсингу шаблону використовується всього чотири регулярні вирази [6] — мовні конструкції, які сильно

навантажують сервер (SMARTY використовує тільки для fetch (типової обробки шаблону), мінімум шість подібних конструкцій);

4. використання кешування, компілюючого режиму роботи, кешування в пам'ять. Компілюючий режим конвертує шаблон JTE в PHP файл завдяки закладеним у шаблонізатор рядковим значенням конструкцій мови PHP. Кешування в пам'ять дозволяє щоразу не звертатися до диску, коли необхідно завантажити шаблон (діє в межах однієї сесії роботи шаблонізатору).

5. MemoryRemapper (MR) — запропонований механізм кешування не тільки шаблону, але й вихідних даних (діє в межах заданого інтервалу й суттєво підвищує продуктивність). За даними утиліти Apache Benchmark (ab), MR підвищує стійкість шаблонізатора до навантажень на 70 — 90 %.

Серед можливостей шаблонізатору можна виділити повністю змінюваний синтаксис, у тому числі його можна змінювати в «гарячому» режимі (прямо під час обробки шаблону), повну зворотню сумісність усіх версій шаблонізатору, можливість взагалі відключити псевдомову JTE і використовувати JTE як нативний шаблонізатор.

Загальна схема роботи JTE 2 наведена на Рис. 1. Уся робота здійснюється через базовий клас JTE, потрібно мінімум операцій, щоб підключити JTE і змусити його працювати: ініціалізація системи керування шаблонами, коли буде завантажений тільки базовий клас JTE, попередня конфігурація деяких параметрів, і підключення, якщо необхідно, JTE 2.

Базовий клас — це кістяк із синтаксисом методів, схожим на SMARTY, підтримка синтаксису псевдомови реалізується в JTE 2 — ядрі шаблонізатору. Для зворотної сумісності можна вказати у якому режимі слід виконуватися JTE, тобто оброблювач псевдокоду може бути повністю відключений.

JTE 2, залежно від шаблону TPL (TPL — абривіатура від «TemPLate», тобто «шаблон», застосовується для позначення файлів шаблону або їх вихідного тексту), може виконувати різні операції: підключати сніпети, завантажувати чанки й, звичайно, обробляти маркери.

Сніпет — це підпрограма шаблонізатору, шій підпрограмі може бути переданий ряд параметрів, після виконання цієї програми. JTE замінить маркер виклику сніпета на значення, що повертає цей сніпет. Поняття «сніпет» не є новим, цей підхід до розробки системи керування шаблонами запозичений з ModX CMS і



Рис. 1 Схеми роботи JTE 2

доброблений для використання у власному продукті. Виклик сніпета здійснюється командою «`{{sSnippetName[:sVar1Name='sVar1Value',sVar2Name='sVar2Value']}}}`». Сніпету може бути передане до 100 параметрів (`sVar1Name` у прикладі), `sSnippetName` повинен бути переданий обов'язково.

Чанк — це статичний HTML код, але він може містити маркери шаблонізатору. Маркер чанку в тексті TPL буде замінений на зміст файла-чанку. Чанки можна завантажувати з коду сніпетів за допомогою API. Чанк не обробляється окремо: спочатку будується TPL шаблон з усіма чанками й тільки потім починається основна обробка. Виклик чанку здійснюється командою «`[sChunkName]`», де `sChunkName` — це ім'я чанку, воно повинне бути передане обов'язково.

Маркер — це елементарна частина шаблону. Його вид може задаватися як на початку ініціалізації JTE, так і в ході його роботи. Маркери JTE обрамляються за допомогою «`{{}}`», «`[[ ]]`» і «`{ }`» для сніпетів, чанків і змінних відповідно.

JTE розбирає («парсит») шаблони нелінійно: спочатку обробляються чанки, далі — усі змінні значення шаблону, потім — сніпети. Така схема роботи має свої переваги й недоліки. Недоліком такої схеми роботи є те, що потрібно трохи більша кількість пам'яті для обробки шаблону, окрім того порушується традиційна схема роботи: наприклад, у JTE, якщо сніпет оголосив чанк, то він не буде оброблений, програміст шаблонів JTE повинен знати про послідовність обробки конструкцій псевдомови.

Перевагою є можливість використання досить складних мовних конструкцій, якщо знати пріоритет операцій. На відміну від SMARTY, JTE

підтримує змінні значення в значеннях змінних, хоча це порушує дану послідовність. За замовчуванням, значення змінних JTE розглядає як повноцінні JTE шаблони, і, перш ніж потрапити в основний шаблон, конструкції псевдомови в їхніх значеннях будуть оброблені теж.

JTE може працювати як в інтерпретуючому, так і в компілюючому режимах. Інтерпретуючий режим кардинально відрізняється від компілюючого, тобто в першому випадку не проводиться EVAL готового скомпільованого коду без збереження. Замість цього використовуються прямі інструкції PHP з попереднім перехопленням виводу (виду «`ob_`»). Це також дозволяє підвищити швидкість роботи шаблонізатору.

JTE не використовує ніяких СУБД. Усі дані повинні бути традиційно передані шаблонізатору з тексту основної програми. Шаблони, чанки, сніпети завантажуються з текстових файлів, з диску. У зв'язку із цим, можуть виникати проблеми при занадто активній роботі з жорстким диском серверу.

JTE — це система керування шаблонами, орієнтована на швидкість обробки шаблонів.

Для рішення проблеми швидкодії програми зроблено кілька спроб позбавити програму від зайвого звернення до диску:

- `TPLBuffer` — метод кешування в пам'ять, коли готовий шаблон записується в пам'ять для наступного виводу;

- рівень абстракції баз даних, який застосовується для роботи із шаблонізатором і містить у собі функцію `Memory Remapper`, дозволяє ефективно кешувати усі дані, що надходять, в пам'ять й зберігати сеанси роботи на жорсткий диск для наступного відновлення роботи, що суттєво знижує частоту звертання до жорсткого диску, а, отже, підвищує продуктивність усієї системи.

- для підвищення продуктивності JTE компілює шаблони у звичайний PHP код, який використовує можливості JTE як можливості нативного шаблонізатору.

Тобто, основний інструмент підвищення швидкодії — це кешування. Воно може бути повністю відключене, що часто необхідно при тестах і розробці, однак небажане при повноцінній роботі.

Шаблонізатори можуть обробляти шаблони різної складності. Найбільш ресурсомісткими завданнями для них, на нашу думку є:

- обробка підпрограм («сніпетів») у лексичній

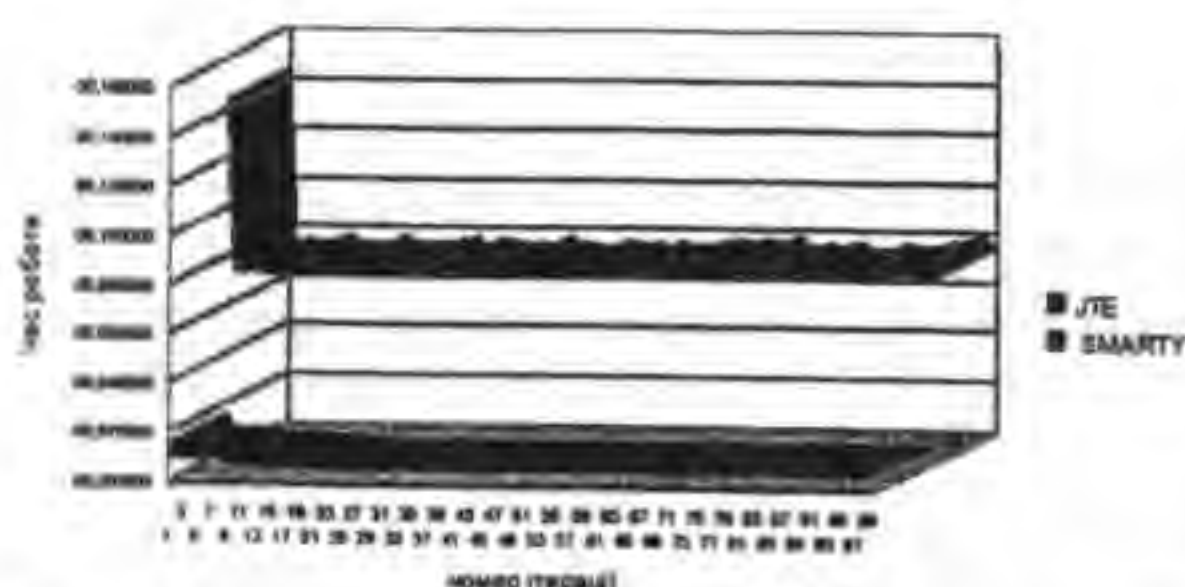


Рис. 2. Порівняльний середніх значень SMARTY та JTE

JTE або «плагинів» у лексиці SMARTY)

- підстановка інших шаблонів (згідно з маркерами шаблонізатору; у лексиці JTE «чанк»);
- обробка складних шаблонів з підстановкою вкладених висловлювань.

Для тестування шаблони були розділені на три рівні складності: прості, середні й складні.

Вони, відповідно, містять у собі просту підстановку змінних, включення підпрограм і інших шаблонів, вкладені конструкції.

Системи керування шаблонами були протестовані на платформі Windows XP SP3, Intel Core 2 Duo 2.1 Ghz, 3.00 ГБ ОЗП. Для тестування обраний популярний набір програмного забезпечення серверів: Apache 2.2.1, PHP 5.2.9.9. Файлова система серверу — NTFS.

Результати тестування наведені на Рис. 2. Коливання результатів на графіку обумовлено технологіями, які використовує Windows, особливостями файлової системи NTFS (наприклад, кешуванням даних, які читалися з диску й т.п.). Кешування файлової системи впливає на результати, однак цими значеннями можна зневажити, оскільки вони порівняно малі.

Список використаних джерел

1. Колисниченко Д. Joomla 1.5. Руководство пользователя / Д.Н. Колисниченко. – М.: Диалектика. – 2009. – 216 с.
2. [електронний ресурс]. – Режим доступу: <http://ru.wikipedia.org/wiki/Шаблонизатор>
3. Лиликович С.А. Разработка Jasenu CMS6 / С.А. Лиликович // Перший крок у дослідженні: Збірник студентських наукових робіт-2009 р., м. Луганськ. – Луганськ: Phoenix, 2009. – С.74-77.
4. Лиликович С.А. Практика создания расширяемой и многофункциональной CMS / С.А. Лиликович // Матеріали Всеукраїнської

Аналіз тестування показує, що середній час роботи JTE менше середнього часу роботи SMARTY у 15 разів, при цьому спостерігається така тенденція: коли шаблонізатори обробляють простий шаблон, різниця в часі обробки більш значна, ніж коли шаблонізатори обробляють складні шаблони. Це досягається за рахунок серйозного спрощення процесу обробки шаблону (зокрема, спрощення розбору шляхом застосування всього чотирьох регулярних виразів), кешування й методу прямого виконання команд без попереднього формування PHP коду.

На підставі проведеного порівняльного тестування JTE і SMARTY, а також на підставі попереднього аналізу ринку систем керування шаблонами можна зробити наступні висновки:

1. Ринок систем керування шаблонами пропонує масу рішень, однак це не задовольняє існуючий попит. При цьому, програмісти й дизайнери потребують не стільки багатофункціональної, скільки швидкої і простої в роботі системи керування шаблонами.

2. Розроблена архітектура для шаблонізатору, яка дозволяє суттєво не втрачаючи швидкодії, розширювати можливості системи керування шаблонами.

3. На підставі досліджень розроблений власний шаблонізатор JTE – Jasenu Template Engine, продукт повністю функціональний, який розповсюджується вільно (доступний з сайту <http://jcms.ho.ua/>).

4. Проведене тестування швидкодії популярного шаблонізатору SMARTY і JTE показує, що шаблонізатор розроблений за запропонованою архітектурою суттєво випереджає SMARTY за швидкістю роботи.

науково-практичної конференції «Перший крок в науку». – Т.8. – Луганск: Полиграфресурс. 2010. – С. 88-91.

5. Лиликович С.А. Анализ систем шаблонизации современных CMS / С.А. Лиликович, А.В. Логинов // Матеріали международной конференции Theoretical and applied aspects of program systems development. TAAPSD-2010. – Киев: Киевский национальный университет имени Тараса Шевченко. – С. 286-292.

6. [електронний ресурс]. – Режим доступу: <http://php.net/manual/en/book.pcre.php>

Надійшла до редколегії 14.12.10

УДК 004.021: 004.312.4: 004.414.2

Лук'янова О.О. к.ф.-м.н., доц.,
Дережа А.В. аспірант

Про автоматичну систему аналізу деяких властивостей алгоритмічних схем

Побудований технологічний процес аналізу властивостей алгоритмічної схеми на прикладі криптографічного протоколу Діффі-Хеллмана. Цей процес може бути застосовано для верифікації властивостей реактивних систем як статичного, так і динамічного характеру. Наведений технологічний процес повністю автоматичний.

Ключові слова: мережі Петрі, діофантові рівняння, скінченний автомат, верифікація.

Lukyanova E.A. PhD,
Dereza A.V. graduate student

About the automatic system of analysis of some properties of algorithmic schemes

Technological process of analysis of properties of the algorithmic scheme on an example of cryptographic protocol of Diffie-Hellman is constructed. This process is applicable to verification of properties of reactive systems as static character so run-time.

Key words: Petri Nets, Diophantine equations, finite state automata, verification.

*E-mail: lukyanovaea@mail.ru, gdevredina@ukr.net.

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Введення

Для перевірки правильності функціонування проектованої системи альтернативою імітаційному моделюванню й тестуванню є методи формальної верифікації, що забезпечують, по-перше, глибокий аналіз можливих варіантів поведінки системи, по-друге, дозволяють знайти й виправити помилки на будь-яких етапах перевірки, а в деяких випадках гарантуючі відсутність певних помилок.

Для цього реальна система зазнає формалізації: виділяються для розгляду певні її властивості, які за допомогою формальної специфікації представляються у вигляді формули формальної логічної мови, яка верифікується, тобто проводиться перевірка того, чи задовольняє модель заданим специфікованим властивостям реальної системи. У даній роботі пропонується технологічний процес, застосування якого ілюструється на прикладі дослідження властивостей криптографічного протоколу Діффі-Хеллмана, що дозволяє аналізувати алгоритмічну схему (програмну систему). При цьому алгоритмічна схема переводиться в Мережу Петрі (МП) [1], за допомогою якої засобами лінійної алгебри автоматично перевіряються необхідні властивості досліджуваної схеми (системи). Так обчислення S - і T -інваріантів [2] розглянутої МП дозволяє верифікувати, наприклад, такі її статичні (структурні) властивості, як досяжність,

обмеженість, безпека, повторюваність, несуперечність та деякі її динамічні властивості (відсутність дедлоків) [1].

1. Постановка задачі

Криптографічний протокол Діффі-Хеллмана являє собою алгоритмічну схему, що дозволяє при незахищеному від прослуховування, але захищеному від модифікацій каналі зв'язку, одержувати двом сторонам загальний секретний ключ K , призначений для обміну конфіденційною інформацією. При цьому один з користувачів створює свій власний секретний ключ x і на його основі формує проміжний ключ $A = g^x \bmod(n)$, де n і g великі прості числа, відомі обом користувачам, і після обміну проміжними ключами, загальний секретний ключ обчислюється формулою $K = B^x \bmod(n)$, (B - проміжний ключ другого користувача).

Для аналізованої алгоритмічної схеми в роботі будується її абстрактна модель у вигляді Мережі Петрі (МП) [1] і формальний аналіз властивостей моделі проводиться за допомогою методів рішення систем однорідних та неоднорідних діофантових рівнянь (СЛОДР, СЛНДР) над множиною натуральних чисел, за допомогою алгоритму TSS [3, 4]. Це дозволяє визначити інваріанти МП та зробити їх аналіз.

2. Моделювання алгоритмічної схеми

Модель схеми являє собою трійку (S, W, M_0) , де S - мережа, $W: F \rightarrow N$ - функція кратності дуг, (F - відношення інцидентності або залежності між місцями P і

переходами T), M_0 – початкова розмітка мережі. Функція $M: P \rightarrow N$ – розмітка мережі S , що описує кількість фішок $M(p)$, що розміщуються цією розміткою в кожне місце $p \in P$. Нехай, $I: (P \times T) \cup (T \times P) \rightarrow N$ – відношення інцидентності:

$$I(x, y) = \begin{cases} n, & \text{якщо } (x, y) \in F \wedge W(x, y) = n; \\ 0, & \text{якщо } (x, y) \in F. \end{cases}$$

Якщо $P = \{p_1, \dots, p_n\}$, то розмітку M мережі можна задати як вектор чисел $M = (m_1, \dots, m_n)$ такий, що $m_i = M(p_i)$ для довільного $i = 1, \dots, n$.

Представимо спочатку роботу схеми Діффі-Хеллмана у вигляді деякої послідовної програми, результатом роботи якої буде стабільне отримання кожним з користувачів загального секретного ключа, для подальшого обміну і шифрування інформації:

```
begin
input(x); input(y); input(n); input(q);
input(A); input(B); input(K1);
input(K2);
A := qx mod(n);
B := qy mod(n);
K1 := Bx mod(n);
K2 := Ay mod(n);
output(K1); output(K2);
end
```

Інтерпретація:

```
input(x); input(K1); input(A);
input(n);
input(y); input(K2); input(B); input(n);
input(q); A := qx mod(n);
input(q); B := qy mod(n);
K1 := Bx mod(n);
K2 := Ay mod(n);
```

t_1
 t_2
 t_3
 t_4
 t_5
 t_6

Мережа Петрі, що моделює цю програму зображена на рисунку 1.

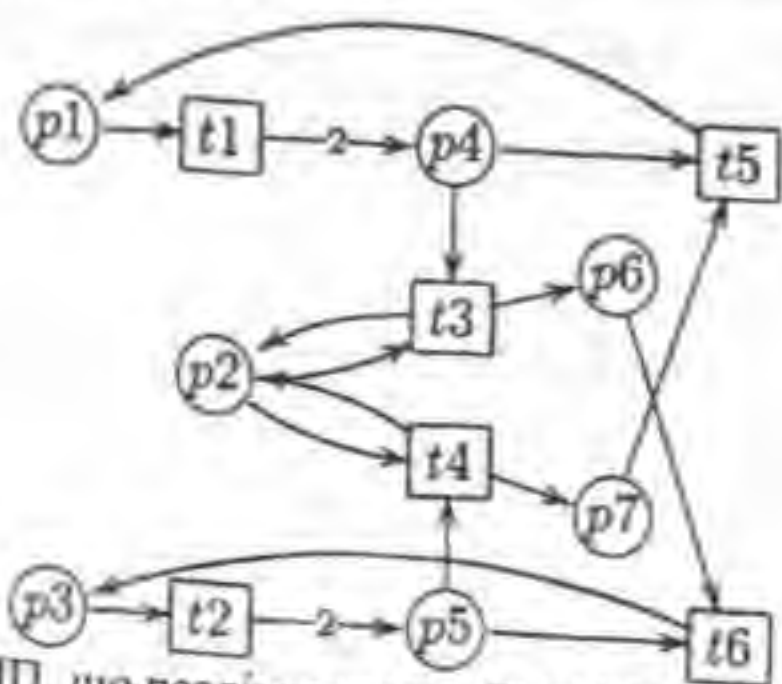


Рис.1 МП, що реалізує схему Діффі-Хеллмана

Переходи t_3, t_4 й t_5, t_6 відповідають процесу обчислення $q^x \bmod(n)$. Розглянемо цей процес як окрему послідовну програму:

```
Begin
input(x); input(n);
input(q); input(r);
while (x > 0) do
begin q := q * q; x-1; end
output(q);
while (q > n) do
begin q := q - n; end
r := n - q;
output(r);
end
```

Інтерпретація:

```
begin
input(x); input(n); input(q);  $t_1$ 
x > 0 ?  $t_2$ 
q := q * q; x := x - 1;  $t_3$ 
input(q);  $t_4$ 
q - n > 0 ?  $t_5$ 
q := q - n;  $t_6$ 
r := n - q;  $t_7$ 
output(r);  $t_8$ 
```

Відповідна мережа Петрі представлена на рисунку 2.

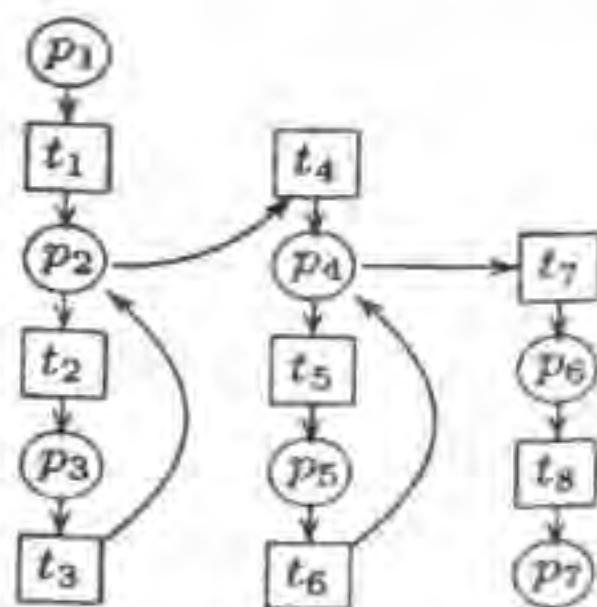


Рис.2 МП, що реалізує процес $q^x \bmod(n)$

Таким чином, отримана вистава вихідної схеми (системи) у вигляді формальної моделі.

3. Аналіз моделі

Для мережі Петрі (S, W, M_0) з n переходами й m місцями будується матриця інцидентності A розміру $n \times m$ із цілими коефіцієнтами

$$a_{ij} = I(t_j, p_i) - I(p_i, t_j),$$

де I – відношення інцидентності даної МП, а коефіцієнти a_{ij} представляють число фішок, які

переміщуються, змінюються й додаються в місце p_i при спрацьовуванні переходу t_j в системі (S, W, M_0) . По матриці інцидентності A будується система рівнянь станів МП (S, W, M_0)

$$A \cdot x = d \quad (1),$$

де $d = M_k - M_0$, (M_0, M_k – відповідно початкова та кінцева розмітки МП),

$$x = \sum_{j=1}^k u_j,$$

де u_k – вектор контролю, що полягає з $n-1$ нульових компонент і однієї j -ої компоненти рівної 1, що сигналізує про спрацьовування на k -му кроці переходу t_j . У випадку досяжності розмітки M_k з розмітки M_0 система (1) завжди має розв'язок [5].

Для МП (S, W, M_0) будується множина T - і S -інваріантів.

Розв'язок x системи лінійних однорідних діофантових рівнянь (СЛОДР) $A \cdot x = 0$, коли в (1) $M_0 = M_k$, називається T -інваріантом МП.

Розв'язок y СЛОДР $A^T \cdot y = 0$, називається S -інваріантом МП.

За допомогою T - і S -інваріантів досліджуються структурні властивості МП.

Означення 1. множина, що породжує, S - (T -) інваріанти називається мінімальною породжуючою множиною, S - (T -) інваріантів, якщо не існує жодної не пустої її підмножини, яка теж є породжуючою множиною.

З теорії діофантових рівнянь [5], випливає, що усічена множина розв'язків СЛОДР, що відповідає даної МП, становить мінімальну породжуючу множину S - (T -) інваріантів, і для побудови цієї множини можна використовувати TSS-алгоритм.

Теорема 1. Якщо всі місця МП покриваються позитивними S -інваріантами, то МП обмежена.

Матриця інцидентності, що відповідає першій МП має вигляд:

-1	0	0	0	1	0
0	0	0	0	0	0
0	-1	0	0	0	1
2	0	-1	0	-1	0
0	2	0	-1	0	-1
0	0	1	0	0	-1
0	0	0	1	-1	0

Оскільки розмітка на k -му кроці $M_k = (1, 1, 1, 0, 0, 0, 0)$ дорівнює початковій розмітці

$M_0 = (1, 1, 1, 0, 0, 0, 0)$ будемо СЛОДР. Її позитивні цілочисленні розв'язки є T -інваріантами МП. Сумісність СЛОДР встановлюємо за допомогою побудови зрізаної множини розв'язків (TSS-алгоритм) – у результаті отримуємо вектор $e = (1, 1, 1, 1, 1, 1, 1)$ – T -інваріант. Для пошуку S -інваріантів (позитивних цілочисельних розв'язків СЛОДР із транспонованою матрицею інцидентності), будемо відповідну СЛОДР. Її сумісність також встановлюється за допомогою TSS-алгоритму, у результаті знаходимо вектори

$$e_1 = (0, 1, 0, 0, 0, 0, 0),$$

$$e_2 = (2, 0, 2, 1, 1, 1, 1) \quad - S\text{-інваріанти.}$$

Матриця інцидентності, відповідна до другої мережі Петри, виглядає таким чином:

-1	0	0	0	0	0	0	0
1	-1	1	-1	0	0	0	0
0	1	-1	0	0	0	0	0
0	0	0	1	-1	1	-1	0
0	0	0	0	1	-1	0	0
0	0	0	0	0	0	1	-1
0	0	0	0	0	0	0	1

За матрицею інцидентності будемо систему лінійних діофантових рівнянь S , яка в даному випадку складається із семи рівнянь. Початкова розмітка для цієї МП і розмітка на k -му кроці наступні:

$$M_0 = (1, 0, 0, 0, 0, 0, 0, 0), \quad M_k = (0, 0, 0, 0, 0, 0, 0, 1).$$

Згідно із критерієм сумісності СЛНДР [3], по отриманій СЛНДР S будемо відповідну їй СЛНДР S' .

Теорема 2. Система S сумісна тоді і тільки тоді, коли сумісна система S' .

Сумісність системи S' встановлює теорема:

Теорема 3. Система S' сумісна тоді і тільки тоді, коли рівняння

$$d_1 u_1 + d_2 u_2 + \dots + d_k u_k - t b_p = 0$$

має хоча б один розв'язок $(\alpha_1, \alpha_2, \dots, \alpha_k, \gamma)$ у множині натуральних чисел N такий, що координати вектора $v = \alpha_1 e'_1 + \alpha_2 e'_2 + \dots + \alpha_k e'_k$

кратні числу γ , де e'_i – TSS-розв'язок системи S' , $i, \gamma \neq 0$, $\alpha_i, \gamma, i \in N$, $i = 1, 2, \dots, k$

Для перших шести рівнянь системи S' множина TSS складається з 6 векторів: $(1, 0, 0, 1, 0, 0, 1, 1)$, $(1, 1, 1, 1, 0, 0, 1, 1)$, $(1, 1, 1, 1, 1, 1, 1, 1)$, $(1, 0, 0, 1, 1, 1, 1, 1)$, $(0, 1, 1, 0, 0, 0, 0, 0)$, $(0, 0, 0, 0, 1, 1, 0, 0)$. Сьомому рівнянню задовольняють тільки 4 перших вектора. Значення на цих векторах для останнього рівняння системи S' такі: 1, 1, 1, 1. Згідно з теоремою 3 отримуємо рівняння:

$$1u_1 + 1u_2 + 1u_3 + 1u_4 - 1t = 0.$$

Один з його коренів $(1, 1, 1, 1, 4)$. Отже, вектор-розв'язок системи S' має вигляд: $(4, 2, 2, 4, 2, 2, 4, 4)$ і система S' сумісна. А з теореми 2 випливає, що спільної буде відповідна до системи S' система S .

Для МП, що моделює схему Діффі-Хеллмана, із множини її S -інваріантів $e_1 = (0, 1, 0, 0, 0, 0, 0, 0)$, $e_2 = (2, 0, 2, 1, 1, 1, 1)$, згідно з теоремою 1 випливає її обмеженість, тобто усі місця МП покриваються ненульовими координатами й можна одержати верхню оцінку для фішок, що додаються в будь-яке місце, наприклад, для

$$p_2: M(p_2) \leq \frac{M_0 e_1^T}{e_1(p)} = 1,$$

а максимальна оцінка числа фішок, наприклад, для

$$p_4: M(p_4) \leq \frac{M_0 e_2^T}{e_2(p)} = 4.$$

Це означає, що отримана МП, будучи обмеженою, не є безпечною (тому що існують позиції мережі, у яких число фішок перевищує значення 1). У досліджуваному випадку можлива примусова безпека – побудувати безпечну мережу, еквівалентну даної. Така мережа в роботі не приводиться у зв'язку з її громіздкістю. Наявність T -інваріантів (усі переходи покриваються ненульовими координатами із множини T -інваріантів) вказує на те, що МП має властивість повторюваності, що інтерпретується в схемі Діффі-Хеллмана як обмін новими ключами нескінченне число раз. Обмеженість досліджуваної МП дозволяє побудувати транзитивну систему (ТС) цієї МП.

4. Транзиційна система моделі і відповідний їй скінченний автомат

Транзиційна система (ТС) являє собою четвірку

$$A = (S = \{a, b, c, d, e, f, g\},$$

$$T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}\}, h, w),$$

де S – множина станів, T – множина переходів, h і w – два відображення з T в S , що ставлять у відповідність кожному переходу t з T два стани $h(t)$ і $w(t)$ – відповідно початок і кінець переходу t . Задамо функції $h(t)$ і $w(t)$ за допомогою наступної таблиці 1:

Таблиця 1

Значення функцій $h(t)$ і $w(t)$ на переходах

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
h	f	g	b	b	a	c	d	e	d	e
w	a	c	a	c	d	e	e	d	f	g

Оскільки множини S і T скінченні, одержимо скінченну ТС, зображену на рисунку 3.

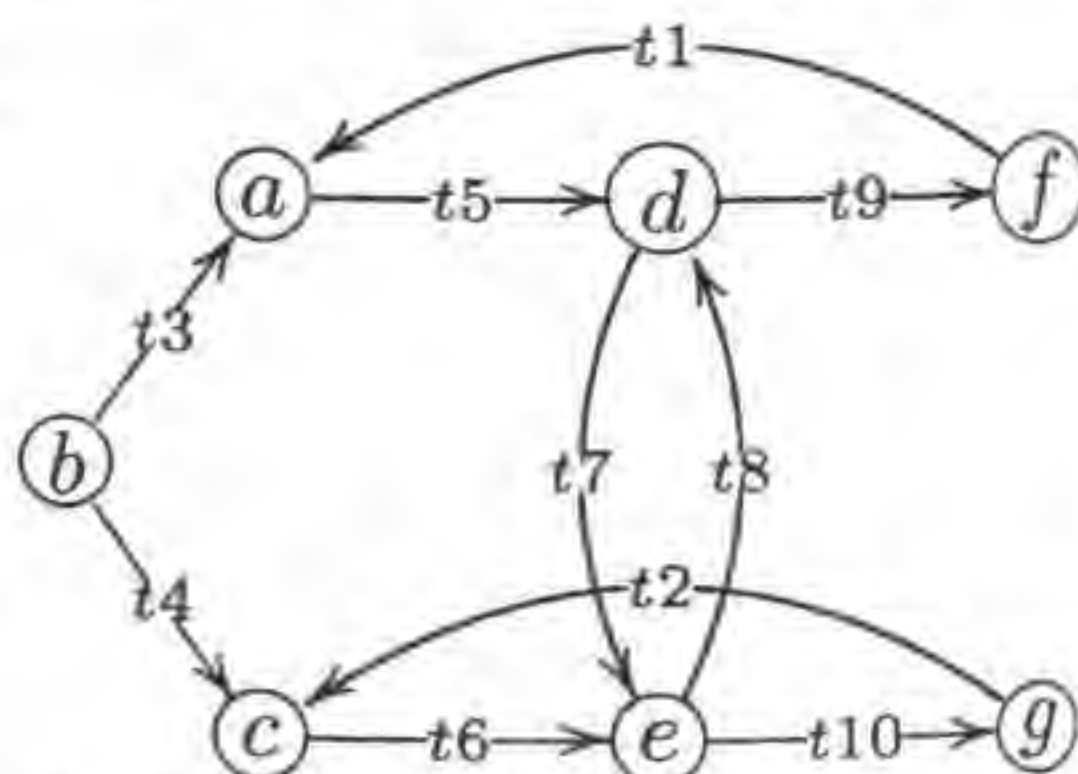


Рис. 3 ТС, моделююча схему Діффі-Хеллмана

Маючи множину станів S ТС, і враховуючи таблицю переходів, одержуємо, що побудована ТС є наступний скінченний автомат (СА):

$$A = (\{a, b, c, d, e, f, g\},$$

$$X = \left\{ \begin{aligned} e_1 &= (1, 0, 0, 0, 1, 0, 0, 0), e_2 = (0, 1, 0, 0, 0, 1, 0, 0), \\ e_3 &= (0, 0, 1, 1, 0, 0, 0, 0), e_4 = (0, 0, 1, 1, 0, 0, 0, 0), \\ e_5 &= (1, 0, 1, 1, 1, 0, 0, 0), e_6 = (0, 1, 1, 1, 0, 1, 0, 0), \\ e_7 &= (0, 0, 0, 0, 1, 0, 0, 0), e_8 = (0, 0, 0, 0, 0, 1, 0, 0), \\ e_9 &= (1, 0, 1, 0, 0, 1, 1, 0), e_{10} = (0, 1, 1, 0, 1, 0, 0, 1) \end{aligned} \right\}$$

$$Y = \{0, 1\}, 0, h, w),$$

де h й w задаються наступними таблицями переходів (таблиця 2) і виходів (таблиця 3):

Таблиця 2

Значення функції переходів

h	a	b	c	d	e	f	g
e_1	a	b	c	d	e	a	g
e_2	a	b	c	d	e	f	c
e_3	a	a	c	d	e	f	g
e_4	a	c	c	d	e	f	g
e_5	d	b	c	d	e	f	g
e_6	a	b	e	d	e	f	g
e_7	a	b	c	e	e	f	g
e_8	a	b	c	d	d	f	g
e_9	a	b	c	f	e	f	g
e_{10}	a	b	c	d	g	f	g

Таблиця 3

Значення функції виходів

h	a	b	c	d	e	f	g
e_1	0	0	0	0	0	1	0
e_2	0	0	0	0	0	0	1
e_3	0	1	0	0	0	0	0
e_4	0	1	0	0	0	0	0
e_5	1	0	0	0	0	0	0
e_6	0	0	1	0	0	0	0
e_7	0	0	0	1	0	0	0
e_8	0	0	0	0	1	0	0
e_9	0	0	0	1	0	0	0
e_{10}	0	0	0	0	1	0	0

Граф переходів і виходів автомата представлений
на рисунку 4.

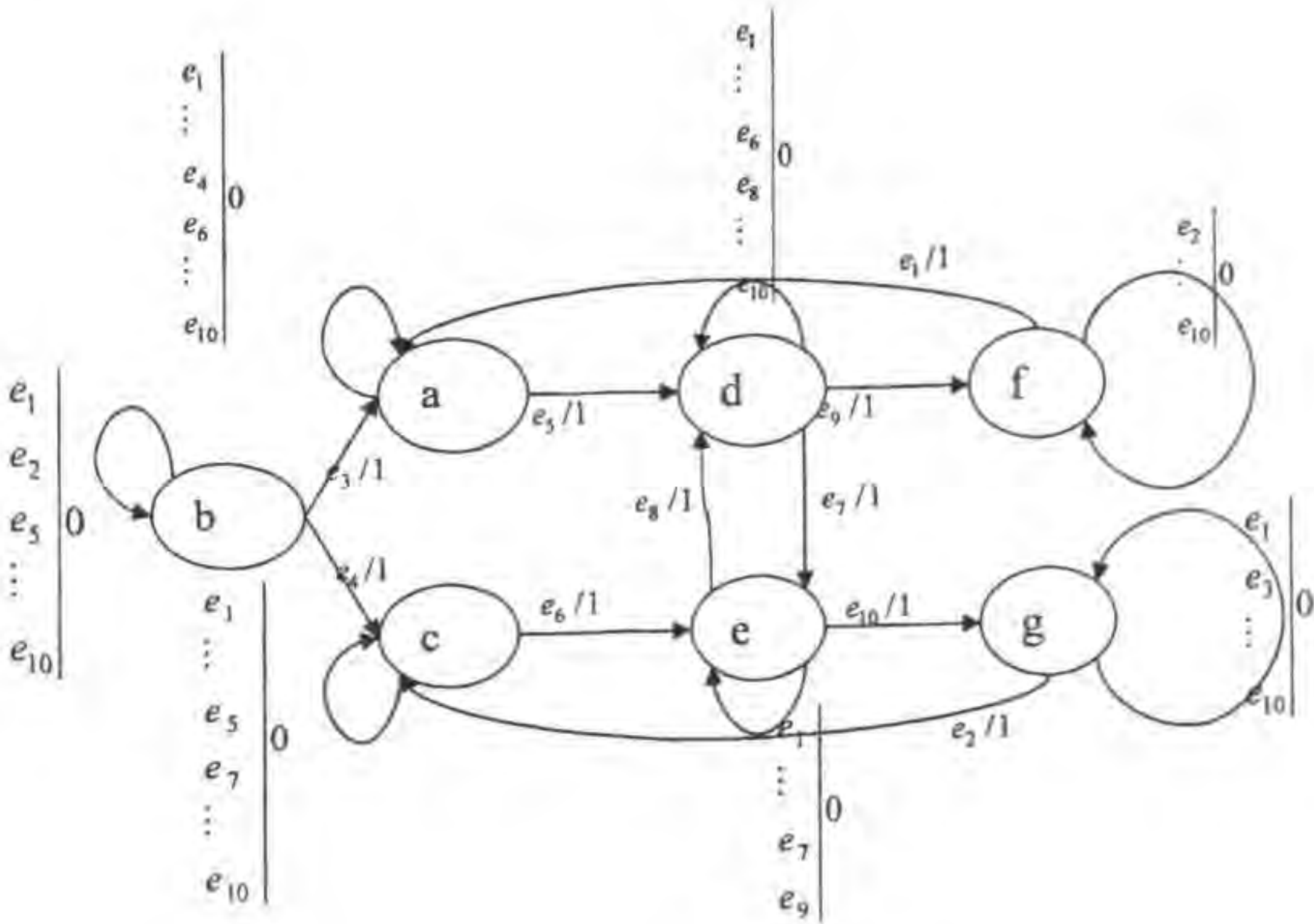


Рис.4 Граф переходів і виходів скінченного автомата, який моделює схему
Діффі-Хеллмана.

Висновок

Для моделювання алгоритмічної схеми (системи) пропонується загальна схема дослідження властивостей реактивної системи. При цьому був використаний математичний апарат мереж Петрі, що дозволило встановити наявність S - (T -) інваріантів, побудованої МП, яка реалізує схему Діффі-Хеллмана. По S - (T -) інваріантам були встановлені такі структурні властивості побудованої МП, як обмеженість і повторюваність. А, оскільки в цієї МП, існує послідовність спрацьовувань переходів, яка запускається принаймні один раз, то розглянута МП має й властивість несуперечності, що для алгоритмічної схеми означає можливість завжди приходити у вихідний стан. При моделюванні схеми Діффі-Хеллмана виявилось необхідним окрема вистава ділянки, що моделює $q^x \bmod(n)$, але на роботі всієї моделі це не відбилося, тому що відповідна до цієї мережі СЛНДР сумісна, а отже в мережі завжди є можливість переходу з одного заданого стану в інший. Аналіз роботи першої МП із урахуванням роботи МП, що моделює обчислення $q^x \bmod(n)$, дає: усі перераховані властивості, побудованої моделі (обмеженість, повторюваність, несуперечність) зберігаються. Крім того, специфіка

досліджуваної схеми така, що ми маємо справу із строго детермінованим процесом, це дозволило провести безпосереднє моделювання відповідного СА. За побудованим СА встановлюються важливі властивості системи: властивість взаємного виключення (mutex) та справедливості (fairness).

Список використаних джерел

1. Котов В.Е. Сети Петри. – М.: Наука, 1984. – 157 с.
2. Murata T. Petri Nets: Properties, Analysis and Applications // Proceedings of the IEEE. – 1989. – Vol.77. – №4. – p.541-580.
3. Крытый С.Л. Критерий срвместности систем линейных диофантовых уравнений над множеством натуральных чисел // Доповіді НАНУ. – 1999. – 5. – С.107-112.
4. Чугаенко А.В. О реализации TSS алгоритма // Управляющие системы и машины. – 2007. – №3. – С.26-32.
5. Крытый С.Л. Алгоритмы решения систем линейных диофантовых уравнений в целочисленных областях // Кибернетика и системный анализ. – 2006. – №2. – С.3-17.

Надійшла до редколегії 15.12.10

УДК 004.519.217

Маєвський Д.А., к.т.н., доц.,
Яремчук С.О., аспірант

Перехідні процеси в лінійних системах та моделювання надійності програмного забезпечення

В статті розглядається підхід до моделювання процесу виявлення помилок в програмних засобах як перехідного процесу в лінійних системах. На підставі цього підходу запропонована та досліджена модель перехідного процесу. Проведено порівняльний аналіз цієї моделі з трьома класичними моделями надійності: Джелінського - Моранди, Шика - Волвертона, Муси та статистичний аналіз результатів розрахунків. Зроблено висновки про ступінь відповідності досліджуваних моделей експериментальним даним.

Ключові слова: модель надійності, програмне забезпечення, програмний комплекс, інформаційні системи.

*E-mail: toe-onpu@ukr.net, svetlana397@yandex.ru

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

У даний час задача прогнозування показників надійності програмних комплексів (ПК) інформаційних систем (ІС) та її кількісна оцінка є дуже актуальною. Це пов'язано з тим, що помилки ПК можуть привести до втрати працездатності ІС, а також викликати серйозні екологічні біди або навіть загибель людей. За останнє десятиріччя масштаби і складність ІС настільки зросли, що процес розробки, відладки і експлуатації їх ПК стає все більш трудомістким і дорогим. За наявними оцінками [1, 2], у зв'язку із зростанням складності ПК, збільшується вірогідність помилок в них, а фінансові витрати на розробку і супровід програмного забезпечення (ПЗ) співвідносяться з витратами на апаратно-технічну частину ІС як 9 до 1.

Надійність ПК, як якісна характеристика, напряду пов'язана з помилками (дефектами, відмовами) програм. Проте, як відзначено в [3], надійне ПЗ зовсім не виключає наявності в ньому помилок. Важливо, щоб ці помилки при практичному використанні ПК в заданих умовах виявлялися достатньо рідко. Причинами, за якими в зданому в експлуатацію ПЗ виявляються помилки, є обмеженість людських, фінансових і часових ресурсів, що виділяються для тестування

Maevsky D.A.,
Yaremchuk S.A. *

Transients in the linear systems and modeling of software reliability

This article describes the model of software reliability proposed by D.A. Mayevski - a model of the transition process, and also comparative analysis of this model with the three classic models of reliability: Jelinski-Moranda, Schick-Wolverton, Musa. We made the calculations identifying bugs in time for this model. A statistical analysis of the results of calculations was performed. The conclusion about the extent to which the investigated models with experimental data correspond.

Key words: reliability model, software, software package, information systems.

програм [5]. Окрім цього, самі методи тестування недостатньо розроблені. В результаті деяка частина дефектів залишається не виявленою на етапі тестування, і спричиняє фінансові та інші ризики на етапі експлуатації ІС.

Тому вивчення, аналіз і розвиток математичних моделей надійності мають велике значення при проектуванні і супроводі складних ІС. В цій статті робиться спроба застосування до моделювання процесу виявлення помилок у програмному забезпеченні ІС добре вивчених методів перехідних процесів у лінійних системах.

Огляд літератури

Для оцінки надійності ПЗ в цьому випадку використовуються спеціальні моделі, які дозволяють більш менш точно спрогнозувати виявлення помилок в процесі експлуатації. Вхідною інформацією для цих моделей служить статистика виявлення помилок і дані про аналізоване ПЗ. На основі цих даних моделі дозволяють спрогнозувати появу помилок в процесі експлуатації. На даний час розроблено близько тридцяти моделей надійності програмного забезпечення (МНПЗ), які відрізняються одна від іншої своїми припущеннями, складом аналізованих даних, і прогнозованими параметрами надійності.

Дослідженням і порівняльному аналізу таких моделей присвячена велика кількість робіт різних авторів, що досліджували дану проблему.

У роботі [1] аналізуються програмні засоби великомасштабних управлінських систем (об'ємом близько 115 тисяч рядків програмного коду) і невеликих програмних проектів (об'ємом до 10 тисяч рядків). Авторами зроблені висновки про те, що для невеликих програмних проектів або коротких періодів відладки модель Джелінського-Моранди [6] і модель Шика-Волвертона [7] на експоненціальному законі розподілу відмов, дають достатньо точний прогноз числа залишкових помилок. Для великомасштабних розробок або проектів з тривалим періодом відладки якнайкращий прогноз дає модель Шика-Волвертона або видозмінений варіант моделі Джелінського-Моранди. Порівняльний аналіз геометричної моделі Моранди і моделі Шика-Волвертона, виконаний в роботі [8] показує, що обидві ці моделі дають стійко завищені оцінки числа залишкових помилок, і є оцінками консервативними або песимістичними. В роботі [9] проведений докладний порівняльний аналіз різних моделей надійності на основі реальних експериментальних даних. Для цих даних найбільшу відповідність показала модель Шика-Волвертона.

У роботі [10] представлена модель, яка розглядає процес виявлення помилок як перехідний процес в програмній системі. Підставами такого підходу є моделі та методи розрахунку перехідних процесів в загальному випадку лінійних систем, що описані в [4].

Позначимо як R_0 невідому поки що кількість помилок на початку експлуатації, $r(t)$ - поточну кількість помилок, що залишилися в ПЗ на момент часу t , а C - загальну кількість місць в ПЗ, в яких можлива помилка. Цей показник будемо називатимемо дефектоємністю системи.

Швидкість виявлення помилок $\frac{dr}{dt}$ буде прямо пропорційна числу наявних в ПЗ на момент часу t помилок r з коефіцієнтом пропорційності k та обернено пропорційною дефектоємності. На підставі цього отримуємо рівняння, що описує процес виявлення помилок у часі:

$$C \cdot \frac{dr}{dt} + k \cdot r = 0 \quad (1)$$

Згідно з [4] це рівняння є подібним до тих, що описують перехідні процеси у лінійних системах. Його розв'язанням експоненціальна залежність

$$r(t) = R_0 \cdot e^{-\frac{k}{C}t} \quad (2)$$

Залежність (2) є рівнянням перехідного процесу в лінійній системі першого порядку. На підставі цього моделювання процесу виявлення помилок в ПЗ можна розглядати, як моделювання добре вивчених в техніці перехідних процесів.

Запропонована модель перехідного процесу дозволяє на підставі даних про кількість виявлених помилок за невеликий інтервал часу експлуатації прогнозувати наступні показники: кількість помилок, що були до ПО на момент початку експлуатації; кількість помилок, що залишилися до ПО на даний момент часу; інтенсивність виявлення помилок в процесі подальшої експлуатації; час повного виявлення наявних помилок. Ці показники дають можливість зробити висновки про якість тестування ПЗ і кваліфікацію розробника. Модель заснована тільки на даних про розподіл вже виявлених помилок в часі, і не вимагає знання інших характеристик ПЗ, що є її безперечною перевагою.

Мета роботи і постановка задачі

Метою даної роботи є порівняльний аналіз моделі перехідного процесу і найбільш часто вживаних МНПЗ для визначення ступеня їх відповідності реальним експериментальним даним.

Для досягнення поставленої мети вирішені наступні задачі:

1. Виконано розрахунок інтенсивності виявлення помилок з використанням запропонованої в [10] моделі перехідного процесу, а також моделей Джелінського-Моранди, Шика-Волвертона, Муси [11].

2. Проведено статистичний аналіз одержаних результатів, і зроблено висновок про ступінь відповідності моделей, що досліджувались реальним даним.

Результати дослідження

В якості експериментальних даних узяті результати виявлення помилок в програмному забезпеченні інформаційної системи з обліку платежів населення за природний газ, що експлуатується в малому державному комунальному підприємстві «Лайф» м. Ізмаїл з 1999 року по теперішній час. Система виконує функції обліку і аналізу платежів, роботи з боржниками, електронного обміну з ПЗ

Таблиця 1

Експериментальні дані про відмови ПО

Місяці роботи ІС	Час роботи в годинах	Кількість помилок за місяць	Кількість помилок з початку експлуатації
1	960	13	13
2	1 920	11	24
3	2 880	8	32
4	3 840	7	39
5	4 800	6	45
6	5 760	5	50
7	6 720	4	54
8	7 680	6	60
9	8 640	3	63
10	9 600	2	65
11	10 560	1	66
12	11 520	1	67

державних структурах. ІС працює в мережевому режимі (6 робочих місць), загальна кількість звернень до системи оцінюється як 12000 в місяць. Розмір системи складає близько 10 тисяч рядків програмного коду. В таблиці 1 приведені експериментальні дані про відмови системи за перші 12 місяців її експлуатації. Результати моделювання залежності кількості виявлених помилок в часі згідно різних моделей наведено в таблиці 2. Для кількісної оцінки ступеню відповідності результатів кожної з моделей даним експерименту використовувалися критерій χ^2 і критерій мінімального сумарного квадратичного відхилення. Розрахункова формула для обчислення критерію χ^2 має вигляд:

$$\chi^2 = n \sum_{i=1}^k \frac{(p_i^* - p_i)^2}{p_i} \quad (4)$$

де k - кількість інтервалів роботи ПЗ (в нашому випадку $k=12$); n - кількість звернень до ПЗ за весь час дослідження (в нашому випадку - 144000 годин), p_i^* - експериментальне значення

Таблиця 2

Результати моделювання

Експериментальні дані		Розрахункова кількість дефектів згідно формул обраних МНПЗ			
Місяці	Помилки (факт)	Перехідного процесу	Джелінського - Моранди	Шика - Волвертона	Муси
1	13	14,1	15,4	17,2	15,0
2	11	11,3	12,0	13,3	11,9
3	8	9,1	9,3	10,2	9,4
4	7	7,3	7,2	7,9	7,4
5	6	5,9	5,6	6,1	5,9
6	5	4,7	4,4	4,7	4,7
7	4	3,8	3,4	3,6	3,7
8	6	3,1	2,6	2,8	2,9
9	3	2,5	2,0	2,1	2,3
10	2	2,0	1,6	1,6	1,8
11	1	1,6	1,2	1,3	1,4
12	1	1,3	1,0	1,0	1,1
Разом	67	66,7	65,7	71,8	67,5

частоти появи помилок на i -тому інтервалі

($p_i^* = \frac{m_i^*}{n}$, m_i^* - кількість помилок, що виявлено експериментально на i -тому інтервалі), p_i - та ж частота, але отримана на підставі моделювання.

Критерій мінімального сумарного квадратичного відхилення (МСКВ) має наступний вигляд:

$$K = \sum_{i=1}^k (m_i^* - m_i)^2 \quad (5)$$

де m_i^* і m_i - експериментальна і розрахункова кількість помилок на i -тому інтервалі.

Результати розрахунків критеріїв відповідності наведені в таблиці 3.

Таблиця 3.

Результати розрахунків критеріїв відповідності

Критерій	Модель перехідного процесу		Модель Муси		Модель Джелінського - Моранди		Модель Шика - Волвертона	
	Значення	Ранг	Значення	Ранг	Значення	Ранг	Значення	Ранг
χ^2	6,31	1	10,08	2	13,45	3	23,45	4
МСКВ	11,98	1	17,58	2	21,97	3	40,48	4

На Рис.1 представлено графік залежності кількості виявлених помилок ПЗ в часі, побудований на підставі даних таблиці 2.

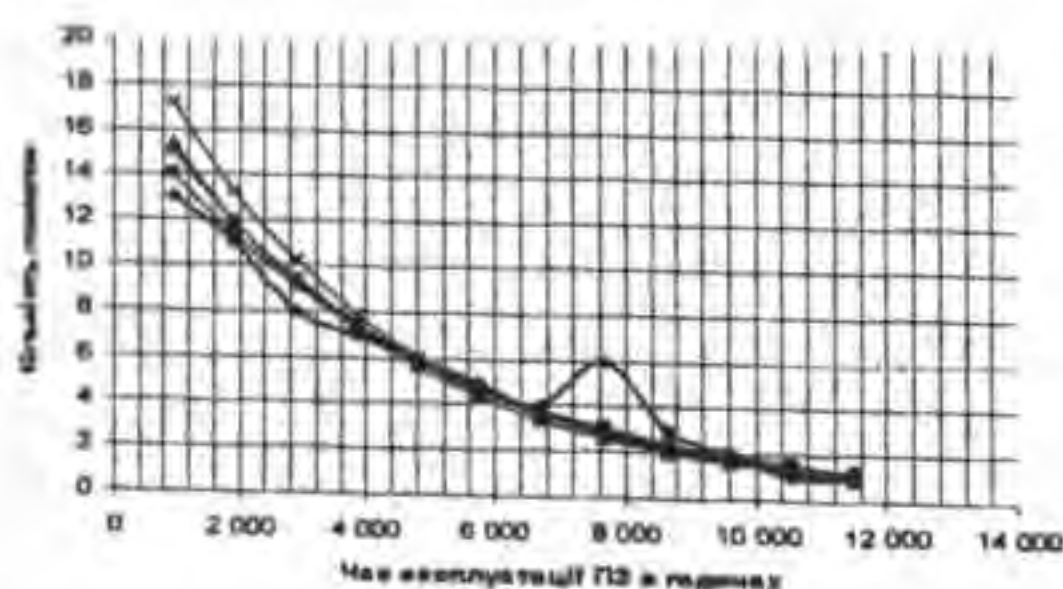


Рис. 1 Результати моделювання кількості виявлених помилок ПЗ в часі

—●— Експеримент
—●— Модель перехідного процесу
—▲— Модель Джелінського-Моранди
—×— Модель Шика-Волвертона
—■— Модель Муси

Список використаних джерел

1. Діллон Б. Інженерні методи забезпечення надійності систем / Б. Діллон, Ч. Сингх // М.: Мир, 1984. – 117 с.
2. Черкесов Г.Н. Надійність апаратно - програмних комплексів / Г.Н.Черкесов/ СПб.: Пітер, 2005. - 6 с.
3. Ямпурин Н.П. Основи надійності електронних засобів / Н.П. Ямпурин, А.В. Баранова // Москва: Вид. центр «Академія», 2010. - 114 с.
4. Гарднер М.Ф. Переходные процессы в линейных системах. / М.Ф. Гарднер, Дж.Л. Бэрнс // М.: Физматгиз, 1961. - 551с.
5. Промислов В.Г. Оцінка надійності програмного забезпечення на різних етапах життєвого циклу складних програм / В.Г. Промислов, А.В. Антонов, С.І. Масолкин, А.С. Степанянц // Праці V Міжн. конф. «Ідентифікація систем і задачі управління», Москва, 2 лютого 2006 р.

Висновки

Ранжування моделей надійності по сумі вибраних критеріїв відповідності показало, що для програмного забезпечення, що було досліджене, найбільш адекватною є модель перехідного процесу. Модель Муси займає друге місце в рейтингу моделей, модель Джелінського-Моранди на третьому місці, модель Шика-Волвертона - на четвертому. Цей порівняльний аналіз доповнює висновки авторів, наведені в огляді літератури.

Актуальними напрямками подальших досліджень уявляються наступні: аналіз відповідності експериментальних і розрахункових даних для інших програмних продуктів, і інших, не розглянутих в цій роботі моделей та залучення поняття ентропії системи для характеристики надійності програмних засобів.

6. Moranda P.B., Jelinski J. Final Report Software Reliability Study. - McDonnell Douglas Astronautic Company. MDC Report № 63921. Dec. 1972.
7. Wolverton R.W., Shick J. Assessment Software Reliability TRW-SS-73-04, 1972.
8. Sukert C.A. An investigation software reliability models / C.A. Sukert // Proc. Annual Reliability and Maintainability Symp. - 1977. - 478-484 p.
9. Харченко В.С. Оцінка і забезпечення якості програмних засобів космічних систем / В.С. Харченко, В.В.Скляр та ін. // Харків: ХАІ, 2007. - 244 с.
10. Антошук С.Г. Прогнозування кількості помилок на етапі експлуатації облікових інформаційних систем, що адаптуються. / С.Г. Антошук, Д.А.Маєвський, С.О.Яремчук // Радіоелектронні і комп'ютерні системи, НТЖ, Харків, ХАІ - 2010 - № 6(47) - 204-209 с.
11. Musa J.D. Validity Execution time theory software reliability // IEEE Trans. on reliability. - 1979. - № 3. - 199-205 p.

Надійшла до редколегії 10.12.2010

УДК 519.816

Міроненко Д.С., аспірант

Вибір варіанту змінно-добового завдання для машинобудівного виробництва з використанням нечіткої моделі прийняття плану

У статті показана актуальність розробки автоматизованої системи планування виробництва. Встановлено, що методами вирішення поставленої задачі є: методи інтелектуального планування та імітаційне моделювання.

Наведена структура гібридної моделі ситуаційного планування та моделювання виробництва. Розглядається нечітка модель прийняття плану.

Ключові слова: планування, виробництво, нечітка модель, лінгвістична змінна.

*E-mail: mironenko_ds@ukr.net

Статтю представив с.п.с., д.ф.-м.н. Буй Д.Б.

Вступ

На підприємствах важкого машинобудування сьогодні використовують ERP-системи для реалізації стандартного циклу планування. Так, наприклад, на Маріупольському заводі важкого машинобудування (ВАТ «МЗТМ») упроваджена і функціонує система SAP R/3. Проте її базова функціональність не забезпечує в належній мірі завдання планування виробництва на цеховому рівні (складання виробничих розкладів) [1]. Функціональність даної платформи ERP покриває істотний об'єм завдань виробничого планування. Виняток становлять рівні оперативного планування виробництва (рис. 1) і деякі завдання управління мережами постачань (SCM), автоматизація яких, як правило, здійснюється за допомогою спеціалізованих прикладень [2]. Що ще раз підкреслює актуальність розробки даних програм.

Питання, пов'язані з побудовою оптимальних планів виробництва, розробкою математичних методів здобуття рішень з використанням відповідних моделей, вивчається в рамках теорії розкладів [3]. Вивчення даних методів показало, що теорія розкладів розглядає прості випадки ухвалення планових рішень в умовах повної визначеності, з чітким знанням цілей і обмежень, з фіксованими відомими значеннями

D. S. Mironenko, PhD student

Selecting the option variable-daily task for machine production using the fuzzy model of plan

The article shows the relevance of automated system for production planning. Established that the methods to solve the problem is: the predictive planning and simulation.

The structure of a hybrid model of contingency planning and simulation production. A fuzzy model for the plan.

Key Words: planning, production, fuzzy model, linguistic variable.

неконтрольованих чинників. Що говорить, про не застосовність їх до вирішення завдань оперативного виробничого планування.

Традиційними методами для завдань планування є мережеві методи [4], імітаційне моделювання [5], комбінаторні процедури і евристичні підходи [6, 7], аналіз обмежень [8, 9].

Встановлено, що найбільш доцільними методами рішення задачі в складній, динамічній, частково спостережуваній проблемній області, якій, по суті, є виробничий процес, залишається комбінація методів імітаційного моделювання і сучасних методів штучного інтелекту (інтелектуального планування, систем нечіткого виводу) [10].

Постановка задачі

Розробити інтелектуальну систему планування виробництва, побудовану на основі гібридної моделі нечіткого ситуаційного планування і моделювання виробництва. Дана система повинна формувати змінно-добові завдання в заготівельному цеху по операції різання металу.

Дана система має функціонувати на цеховому рівні обліку і планування і грати роль проміжної ланки між цеховим рівнем і рівнем управління підприємством KIC SAP R/3.



Рис. 1. Етапи планування виробництва

Структура гібридної моделі ситуаційного планування та моделювання виробництва

Гібридна модель нечіткого ситуаційного планування та моделювання виробництва включає в себе чотири складові і представлена на рис. 2.

Перша модель працює на підставі відомостей, отриманих з SAP R/3 (картки замовлень, маршрут виготовлення) та інформаційної системи Центру АСУ (вибіркова специфікація по виробках). Встановлюється, яким обладнанням буде проведено різання металу кожної позиції замовлення.

В результаті кожній картці замовлення відповідає один план графік різання металу. Обладнання для різання металу може розташовуватися на декількох ділянках. Вибір ділянки здійснюється в залежності від величини поточної завантаженості ділянки. На виході з даної моделі ми маємо множину план графіків, розподілених по ділянках.

У другій моделі на основі чинників (категорія матеріалу, забезпеченість матеріалом ділянки, виробнича напруженість, трудомісткість виготовлення), формуються варіанти змінно-добових завдань по ділянках цеху. Всього складеється п'ять варіантів плану.

Імітаційна модель (третя частина) дозволяє проводити експериментування з різними варіантами плану (змінно-добовими завданнями). Вихідними параметрами є критерії оптимальності (середня тривалість простоїв обладнання, середня тривалість виготовлення деталі, обсяг незавершеного виробництва, виробнича напруженість), які передаються в нечітку модель прийняття плану.

Нечітка модель прийняття плану (четверта частина) використовує систему нечіткого виводу з алгоритмом Мамдані [11] для вибору варіанта плану. У ній критерії оптимальності, виражені у вигляді лінгвістичних змінних, проходять такі стадії: агрегування умов в нечітких правилах продукцій, активізацію висновків в нечітких правилах продукцій, акумулювання висновків нечітких правил продукцій, подання вихідної змінної чітким значенням. Вихідною змінною є ступінь прийняття варіанту плану.



Рис. 2. Гібридна модель нечіткого ситуаційного планування та моделювання виробництва

Змінно-добове завдання з максимальним ступенем прийняття плану направляється у виробництво.

Нечітка модель прийняття плану

В [6] викладено численні підходи до розробки моделей прийняття рішень вибору варіанта з безлічі альтернатив. Ці підходи можна розділити на наступні основні групи: статистичні методи, нейронні мережі, експертні методи, нечітко-множинні описи.

Вибір підходу залежить від багатьох чинників, до яких можна віднести суб'єктивні переваги розробника, наявність і якість вихідних даних, цілі та завдання побудови моделі, складність взаємозв'язків між факторами і критеріями ефективності плану.

В умовах предметної області найбільш адекватним інструментом формалізації задачі вибору альтернативи плану виробництва є теорія нечітких множин.

У контексті вирішення даної задачі визначальними є наступні сильні сторони нечітко-множинного підходу [12]: опис умов і рішення задач на мові, близькій до природної мови; рішення задач з неповними і ненадійними вихідними даними; використання досвіду та інтуїції експерта; пояснення, яким чином отримана класифікація; якісна і кількісна оцінка ступеня прийняття плану. Останнє твердження дуже важливо, оскільки особі, що приймає рішення, крім кількісних значень, необхідно знати, прийнятні чи отримані значення, чи хороші вони і в якій мірі.



Рис. 3. Архітектура нечіткої моделі прийняття плану

Нечітка модель прийняття плану представляє собою систему нечіткого виводу і призначена для перетворення значень вхідних даних (критеріїв плану) у вихідні (ступінь прийняття плану) на основі використання нечітких правил [11]. Архітектура нечіткої моделі прийняття плану представлена на рис. 3.

Наведемо множину вхідних лінгвістичних даних $V = \{\beta_1, \beta_2, \beta_3, \beta_4\}$ моделі прийняття плану.

У якості першої вхідної змінної використовується виробнича напруженість в цеху, яка оцінює стадію виконання плану виробництва з точки зору повного часу, необхідного для його виконання, і наявного запасу робочого часу.

Чим менше ця величина, тим більш ефективним є варіант плану. Низьке значення виробничої напруженості говорить про те, що в ситуації, що склалася на виробництві відсутні такі поняття як: підвищене навантаження на обладнання і відповідно на працівників цеху, затримки у виробництві і відставання від плану.

Лінгвістична змінна «виробнича напруженість в цеху» визначається традиційно [11] як математична структура $\langle \beta_1, T, X \rangle$, де β_1 – «виробнича напруженість в цеху» – назва лінгвістичної змінної; $T = \{\text{«низька»}, \text{«середня»}, \text{«висока»}\}$ – множина її термів; $X = [0, 10]$ – універсум нечітких змінних; PS – «низька», PM – «середня», PB – «висока».

Функції приналежності термів задаються однією з нижче наступних функцій (1), (2), (3).

$$f_z(x; a, b) = \begin{cases} 1, & x < a \\ \frac{1}{2} + \frac{1}{2} \cos\left(\frac{x-a}{b-a} \pi\right), & a \leq x \leq b \\ 0, & x > b \end{cases} \quad (1)$$

$$f_s(x; a, b) = \begin{cases} 1, & x < a \\ \frac{1}{2} + \frac{1}{2} \cos\left(\frac{x-b}{b-a} \pi\right), & a \leq x \leq b \\ 0, & x > b \end{cases} \quad (2)$$

$$f_n(x; a, b, c, d) = f_s(x; a, b) \cdot f_z(x; c, d) \quad (3)$$

Параметри термів лінгвістичної змінної «виробнича напруженість в цеху» зведені до табл. 1.

Таблиця 1

Параметри термів лінгвістичної змінної «виробнича напруженість в цеху»

Ім'я терма	Ім'я функції	Параметри				Діапазон універсуму $X = [X_1, X_2]$	
		a	b	c	d	X_1	X_2
PS	$f_z(x; a, b)$	1	4	–	–	0	10
PM	$f_n(x; a, b, c, d)$	1	4	6	9	0	10
PB	$f_s(x; c, d)$	–	–	6	9	0	10

В якості другої вхідної змінної використовується середня тривалість простоїв обладнання. Ця величина характеризує виробничий процес і чим вище її значення, тим менш ефективний план виробництва.

За аналогією визначена лінгвістична змінна «середня тривалість простоїв обладнання»: $\langle \beta_2, T, X \rangle$, где β_2 = «середня тривалість простоїв обладнання»; $T = \{\text{«мала»}, \text{«середня»}, \text{«велика»}\}$; $X = [0, 10]$; S – «мала», M – «середня», B – «велика».

Параметри її термів зведені до табл. 2.

Таблиця 2

Параметри термів лінгвістичної змінної «середня тривалість простоїв устаткування»

Ім'я терма	Ім'я функції	Параметри				Діапазон універсуму $X = [X_1, X_2]$	
		a	b	c	d	X_1	X_2
S	$f_2(x; a, b)$	1	3	–	–	0	10
M	$f_n(x; a, b, c, d)$	1	4,5	5,5	9	0	10
B	$f_3(x; c, d)$	–	–	7	9	0	10

В якості третьої вхідної змінної використовується оцінка середньої тривалості виготовлення деталі. Ця величина залежить від тривалості очікування напівфабрикату до часу початку обробки, тривалості обробки напівфабрикату та простоїв обладнання, пов'язаних з відмовами. Чим менше ця величина, тим менше відмов та простоїв було в ході виробничого процесу, а відповідно і наладок, налаштувань і ремонтів обладнання.

Визначається лінгвістична змінна «середня тривалість виготовлення деталі» як математична структура: $\langle \beta_3, T, X \rangle$, де β_3 = «середня тривалість виготовлення деталі»; $T = \{\text{«мала»}, \text{«середня»}, \text{«велика»}\}$; $X = [0, 10]$; S – «мала», M – «середня», B – «велика».

Параметри термів лінгвістичної змінної «середня тривалість виготовлення деталі» зведені до табл. 3.

В якості четвертої вхідної змінної використовується обсяг незавершеного виробництва. Незавершене виробництво має місце при будь-якому вигляді виробничої діяльності, але чим менше ця величина на кінець планованого періоду, тим ефективніше проводилося планування.

Визначається лінгвістична змінна «обсяг

незавершеного виробництва» як математична структура: $\langle \beta_4, T, X \rangle$, де β_4 = «обсяг незавершеного виробництва»; $T = \{\text{«малий»}, \text{«середній»}, \text{«великий»}\}$; $X = [0, 10]$; S – «малий», M – «середній», B – «великий».

Параметри термів лінгвістичної змінної «обсяг незавершеного виробництва» зведені до табл. 4.

Таблиця 3

Параметри термів лінгвістичної змінної «середня тривалість виготовлення деталі»

Ім'я терма	Ім'я функції	Параметри				Діапазон універсуму $X = [X_1, X_2]$	
		a	b	c	d	X_1	X_2
S	$f_2(x; a, b)$	0,4	3,6	–	–	0	10
M	$f_n(x; a, b, c, d)$	1,4	4,6	5,4	8,6	0	10
B	$f_3(x; c, d)$	–	–	6,4	9,6	0	10

Таблиця 4

Параметри термів лінгвістичної змінної «Обсяг незавершеного виробництва»

Ім'я терма	Ім'я функції	Параметри				Діапазон універсуму $X = [X_1, X_2]$	
		a	b	c	d	X_1	X_2
S	$f_2(x; a, b)$	2	3	–	–	0	10
M	$f_n(x; a, b, c, d)$	2	4	6	8	0	10
B	$f_3(x; c, d)$	–	–	7	8	0	10

Множина вихідних лінгвістичних змінних: $W = \{\varpi_1, \varpi_2, \dots, \varpi_S\}$ представлено однією змінною «ступінь прийняття плану», яка оцінює ефективність плану виробництва (змінно-добового завдання) і служить основою для прийняття рішення про запуск у виробництво відповідного варіанта плану. Рішення про запуск даного плану у виробництво приймається у разі високої оцінки цієї вихідної змінної.

Лінгвістична змінна «ступінь прийняття плану» як математична структура виглядає наступним чином: $\langle \varpi_1, T, X \rangle$, де ϖ_1 = «ступінь прийняття плану»; $T = \{\text{«дуже низька»}, \text{«низька»}, \text{«середня»}, \text{«висока»}, \text{«дуже висока»}\}$; $X = [0, 10]$; NB – «дуже низька», NS – «низька», Z – «середня», PS – «висока», PB – «дуже висока».

Параметри термів лінгвістичної змінної

зведені до табл. 5.

Таблиця 5
Параметри термів лінгвістичної змінної «ступінь прийняття плану»

Ім'я терма	Ім'я функції	Параметри				Діапазон універсуму $X = [X_1, X_2]$	
		a	b	c	d	X_1	X_2
N B	$f_z(x; a, b)$	1	2	—	—	0	10
N S	$f_n(x; a, b, c, d)$	1	2	3	4	0	10
Z	$f_n(x; a, b, c, d)$	3	4,5	5,5	7	0	10
P S	$f_n(x; a, b, c, d)$	6	7	8	9	0	10
P B	$f_s(x; a, b)$	—	—	8	9	0	10

У нечіткій моделі прийняття плану використовується вісімдесят одне правило нечітких продукцій: $P = \{R_1, R_2, \dots, R_{81}\}$ у вигляді (4).

ПРАВИЛО <#>:

$$\begin{aligned}
 &\text{ЯКЩО } \beta_1 \in \alpha_1'' \\
 &\quad \text{ТА } \beta_2 \in \alpha_2'' \\
 &\quad \text{ТА } \beta_3 \in \alpha_3'' \\
 &\quad \text{ТА } \beta_4 \in \alpha_4'' \\
 &\text{ТО } \omega_1 \in y_1''
 \end{aligned} \quad (4)$$

Наведемо приклад правила на природній мові, воно сформоване зі слів експерта в предметній області:

ПРАВИЛО <1>: ЯКЩО «величина виробничої напруженості низька»

ТА «середня тривалість простоїв устаткування мала»

ТА «середня тривалість виготовлення деталі мала»

ТА «обсяг незавершеного виробництва малий»

ТО «ступінь прийняття плану дуже висока»

У базі знань дане правило записано у вигляді умовних позначень (табл. 6).

В якості схеми нечіткого виводу використовується метод Мамдані [11], метод активації – міп-активація, яка обчислюється за формулою (5), у всіх правилах у якості логічної зв'язки для умов застосовується нечітка

кон'юнкція, тому в якості методу агрегування використовується міп-кон'юнкція (6). Для акумуляції висновків правил використовується метод мах-диз'юнкції (7). В якості методу виключення нечіткості використовується метод центру ваги (8).

$$\mu'(y) = \min\{c_i, \mu(y)\} \quad (5)$$

$$T(A \cap B) = \min\{T(A), T(B)\} \quad (6)$$

$$\mu_D(x) = \max\{\mu_A(x), \mu_B(x)\} \quad \forall x \in X \quad (7)$$

$$y = \frac{\sum_{i=1}^n c_i \cdot w_i}{\sum_{i=1}^n c_i} \quad (8)$$

Таблиця 6
Приклад правила нечітких продукцій для нечіткої моделі прийняття плану

№ правила	Вхідні змінні				Вихідна змінна ω_1
	β_1	β_2	β_3	β_4	
1	PS	S	S	S	PB

Етап прийняття рішень щодо вибору варіанта плану вважається закінченим, коли вихідний лінгвістичної змінної визначається підсумкове кількісне значення у формі деякого дійсного числа, і план з максимальним ступенем прийняття плану запускається у виробництво.

Адекватність нечіткої моделі прийняття плану

Для оцінки адекватності нечіткої моделі прийняття плану оцінювалася повнота і несуперечність нечітких продукційних правил, обчислювалися ймовірності помилок першого і другого роду для отриманих результатів.

Показник повноти переліку продукційних правил змістовно означає, що для кожного поточного набору вхідних змінних існує хоча б одне нечітке правило, відправлення яких має ненульову ступінь приналежності.

Несуперечність системи нечітких продукційних правил трактується як відсутність правил, що мають схожі посилення і різні або взаємовиключні сліdstва.

База знань нечітких продукцій володіє повнотою і несуперечністю.

Керуючись апаратом математичної статистики [9], введено поняття помилок першого і другого роду для оцінки вихідної

моделі.

Ряд типових виробничих ситуацій (варіантів змінно-добового плану і факторів які їх характеризують) були запропоновані до розгляду фахівцям планово-диспетчерського бюро цеху, які справили експертну оцінку даних планів. Таким чином, дані плани були класифіковані на «прийняті» і «не прийняті» у виробництво.

Цей же ряд варіантів плану був оцінений нечіткою моделлю прийняття плану, отримані результати показали, що відносна похибка становить менше 1,2%, в середньому - 0,37%, що дозволяє говорити про досить високий ступінь адекватності побудованої нечіткої моделі прийняття плану.

Висновки

У статті обґрунтована актуальність розробки системи автоматизації управлінської діяльності

Список використаних джерел

1. Каргин А.А. Опыт автоматизации производственного планирования на ОАО «МЗТМ» / А.А. Каргин, Д.С. Мироненко // Вестник Донецкого национального университета, выпуск 2/2008, ч.2. - С. 517-521.
2. Громов С.А. Возможности использования ERP-системы для поддержки оперативного планирования производства: [Електронний ресурс] / С.А. Громов // Chief Information Officer. - 2006. - №9. - С. 18 - 24. Режим доступа до журн.: <http://offline.cio-world.ru/2006/52/286446/>
3. Танаев В.С. Введение в теорию расписаний / В.С. Танаев, В.В. Шкурба. - М.: Наука, 1975. - 256 с.
4. Модер Дж. Метод сетевого планирования в организации работ / Пер. с англ., Дж. Модер, С. Филлипс. - М. - Л.: Энергия, 1966. - 303 с.
5. Саломатин Н.А. Имитационное моделирование в оперативном управлении производством / Н.А. Саломатин, Г.В. Беляев, В.Ф. Петроченко. - М.: Машиностроение, 1984. - 208 с.
6. Jonsson A., Morris P., Muscettola N., Rajan K., Smith B. (2000) Planning in interplanetary space: Theory and practice. In Proceedings of the 5th International Conference on Artificial Intelligence Planning Systems (AIPS-00), P.177-186, Breckenridge, Colorado. AAAI Press.

БАТ «МЗВМ», функціонує на цеховому рівні планування і грає роль проміжної ланки між цеховим рівнем і рівнем управління підприємством SAP R / 3.

Розглянуто різні підходи щодо вирішення задачі оперативного планування. Встановлено, що планування у виробничому середовищі демонструє недетермінізму, виробниче середовище носить характер складної, динамічної проблемної області та найбільш прийнятним рішенням є використання гібридної моделі нечіткого ситуаційного планування та моделювання виробництва.

Наведена нечітка модель прийняття плану, оцінена повнота і несуперечність правил управління, розглянута ступінь адекватності моделі.

7. Fukunaga A.S., Rabideau G., Chien S., Yan D. (1997) ASPEN: A framework for automated planning and scheduling of spacecraft control and operations. In Proceedings of the International Symposium on AI, Robotics and Automation in Space, p.181-187, Tokyo.
8. Bensana, E.W. et al., 1986. An expert-system approach to production scheduling. In, Proc. IEEE Robotics and Automation, pp.1645-1650.
9. Erschler, J. and Esquirol, P., 1986. Decision-aid in job shop scheduling: A knowledge based approach. In, Proc. IEEE Robotics and Automation, pp. 1651-1656.
10. С. Рассел. Искусственный интеллект: современный подход / С. Рассел, П.Норвиг 2-е изд.,: Пер. с англ. - М.: Издательский дом «Вильямс», 2007. - 1408 с.
11. Леоненков А.В. Нечеткое моделирование в среде MathLab и FuzzyTech. / А.В. Леоненков - СПб: БХВ - Петербург, 2005. - 736 с.: ил.
12. Заде Л.А. Роль мягких вычислений и нечеткой логики в понимании, конструировании и развитии интеллектуальных систем / Л.А. Заде // Новости Искусственного Интеллекта. 2001. - №2-3. - С. 7-11.
13. Калинина В.Н., Математическая статистика: учебник / В.Н. Калинина, В.Ф. Панкин. - 3-е изд. - М.: Высш. шк., 2001. - 336 с.

Надійшла до редколегії 15.12.10

УДК 004.4'244

Парасюк І.М., чл.-кор. НАНУ, проф.,
Єршов С.В., к.ф.-м.н., с.н.с.

Мультиагентні системи на основі нечітких моделей: архітектура, координація та трансформації

Розглядаються нечіткі архітектурні моделі мультиагентних систем для розподіленого середовища, що мають властивості реактивності і мотивованості. Запропоновані засоби опису поведінки нечітких агентів на основі трансформацій нечітких графів, способи їх координації і адаптації з врахуванням нечітких значень корисності. Розроблені основні метамоделі та трансформації нечітких мультиагентних систем на основі моделі-орієнтованої архітектури, що дозволяють згенерувати нечіткі платформно-незалежні моделі внутріагентного управління.

Ключові слова: мультиагентні системи, MDE, нечіткі моделі.

*E-mail: ivparl@i.com.ua, sershv@ukr.net

Статтю представив доктор фіз.-мат. наук Буй Д.Б.

Програмні моделі гуманістичних систем – систем, в основу яких покладений принцип гранулювання інформації, а процеси їх створення відображають свідомість людини [1], являються важливими об'єктами вивчення теорії нечітких систем. До таких сучасних і перспективних програмних моделей відносяться і нечіткі мультиагентні системи, для успішного становлення і розвитку яких потрібний високий рівень абстрактності цільових платформ розробки, створення конструктивних архітектурних моделей та засобів породження відповідних цільових агентів.

Зростання інтересу до можливостей мультиагентних інтелектуальних систем в контексті розробки програмного забезпечення, що спостерігається останнім часом, ставить завдання створення методів і засобів моделі-орієнтованої трансформаційної розробки (Model-Driven Engineering, MDE) таких систем. Незважаючи на те, що деякі відомі методології розробки мультиагентних систем, такі як Tropos [2] і INGENIAS [3] пропонують методи та інструменти для автоматизації моделі трансформацій, вони призначені для

I. M. Parasyuk, prof.,
S. V. Yershov, Ph.D.

Multiagent systems based on fuzzy models: architecture, coordination and transformations

Fuzzy architectural models of multiagent systems for distributed environment, that possess properties of reactivity and motivation, are investigated. Means of description of fuzzy agents behavior on the basis of transformations of fuzzy graphs, methods of their coordination and adaptation taking into account fuzzy values of utility are offered. Basic metamodels and transformations of the fuzzy multiagent systems on the basis of model-oriented architecture that allow to generate fuzzy platform-independent models for intra-agent control are developed.

Key Words: multiagent systems, MDE, fuzzy models.

використання тільки на деяких етапах розробки і не можуть розглядатися як систематичні.

Окремим агентам таких розподілених систем відкритий доступ тільки до їхніх власних станів, а не до глобального стану всієї системи, стосовно якого агент може робити тільки певні припущення. До того ж, будь-середовище передачі асинхронних повідомлень між агентами не є абсолютно надійним і спричиняє запізнювання повідомлень, яке значно збільшує ступінь невизначеності станів агента. Тому роботою таких агентів можна управляти тільки з урахуванням ступеня досяжності таких "розмитих" станів, для чого обґрунтованим видається використання нечіткої математики.

В світлі відзначеної проблематики в даній роботі розглядаються питання формалізації архітектури мультиагентних систем на основі нечітких моделей та питання розробки відповідних процедур, метамоделей і трансформацій, які їх породжують, для моделі-орієнтованої розробки з використанням підходу MDE. Перелічені проблеми тематично вписуються в подальший розвиток досліджень у напрямі становлення моделі-орієнтованих

архітектур програмних систем в нечіткому представленні [4-6].

Архітектура інтелектуальних агентів: загальні положення

Під терміном інтелектуальний агент розуміють програми, які отримують інформацію із оточуючого середовища та виконують над ним відповідні операції (дії), а їх поведінка – раціональна [7]. Архітектурно такі агенти поділяється на чотири загальні класи: дедуктивні агенти, реактивні агенти (агенти, що ґрунтуються на поведінці), деліберативні (“розумні”) агенти і агенти, які базуються на спонуканнях [8-12].

При дедуктивному підході агент маніпулює символічним представленням середовища, використовуючи операції логічного виведення [8-10]. Основні недоліки даного підходу: проблема трансдукції, необхідність забезпечення несуперечливих, надійних і “чітких” знань, неможливість маніпулювати описом динамічного і стохастичного середовища, строга ієрархічність, необхідність побудови завершеного (повного) агента для проведення випробувань, що не дозволяє проводити ітераційні покрокові процеси розробки.

Парадигма Переконавання-Бажання-Намір (Belief-Desire-Intention) втілює один з основних видів деліберативних (“розумних” агентів) і містить явно представлену структуру даних, відповідає цим трьом вказаним властивостям міркувань, що вживаються агентом при вирішенні задач [8-10].

Бажання, яке BDI агент [8,9] прагне досягти, прочитується із стека намірів. Цей стек містить всі бажання (тобто цілі), які агент зобов’язався виконати. Агент відшукує плани, які вказують на бажання, що витягнуто із стека, у якості постумов. Із цих планів, лише ті, які задовольняють власній передумові (на основі його поточних переконань), стають можливими варіантами або бажаннями для агента.

Деліберативний агент “починає роздумувати”, використовуючи два функціональні компоненти. Перший компонент це – функція вибору $options: \wp(Bel) \times \wp(Int) \rightarrow \wp(Des)$, яка для поточної множини переконань Bel і намірів Int , виробляє шукану множину можливих бажань Des . Другий компонент це – функція фільтрації $filter: \wp(Bel) \times \wp(Des) \times \wp(Int) \rightarrow \wp(Bel)$, використовується в процесах вибору між конкуруючими варіантами. В процесі розмірковувань подальші “бажання”

поміщаються в стек намірів, що у свою чергу запускає пошук більшої кількості планів, аби досягти вказаної мети, і так далі. Оскільки розмір структури намірів стає дуже великим, обмеження, що накладаються на обчислювальні ресурси для планування, уточнення і оцінки цих намірів, викликають “пізнавальне перевантаження” агента BDI. На жаль, механізми фільтрації, не допомагають значно скоротити “перевантаження” оскільки, з часом генерується ряд дійсних альтернатив, які всі сумісні з поточними намірами агента.

Іншим можливим рішенням є використання реактивних (поведінкових) агентів [8-11]. У таких агентах не розглядаються дії з планування, що ґрунтуються на складних внутрішніх представленнях середовища через невід’ємні помилки їх чіткого представлення і асоційованих з їх обробкою витрат часу. Реактивний агент Ag_r генерує відповідь з множини можливих відповідей $R = \{r_1, r_2, \dots\}$, на основі поточних стимулів S , що може бути представлене наступним відображенням: $Ag_r: \beta(S) \rightarrow R$. Агент Ag_r вибирає поточну відповідь r з безлічі можливих доступних відповідей R кожного разу коли значення стимулу більше, ніж поріг τ . Реактивні агенти можуть бути розроблені з використанням ітеративного процесу з використанням кооперативних або конкурентних методів композиції функцій поведінки β_i (де $\beta_i \in \beta$), що задаються окремо. Недоліком таких методів композиції є їх складність. Крім того, прагнучи спростити функцію поведінки, такі агенти не враховують можливого стану самого агента і подальшої мотивації.

Як можлива альтернатива розглядаються мотивовані агенти [12]. Різні спонукання (мотивації) – ключові визначальні чинники, за допомогою яких агент може виробляти різноманітну поведінку, що має вищу ступінь вигоди (або корисності).

При такому підході, мотив m – це відображення поточних переконань bel агента про стан його середовища в множину активних спонук $m: \wp(bel) \rightarrow \wp(m)$. Такий агент матиме дві функції: породження мети і активація мети обоє з яких призначені для генерування активних цілей у відповідь на виявлені зміни в його поточних переконаннях. Привабливість цього підходу в порівнянні з реактивними агентами в тому, що призначені для користувача очікування відносно поведінки агента відображаються в профіль спонук, який може потім бути

зіставлений з реальною поведінкою агента і вдосконалений. На відміну від деліберативного методу, такий тип агентів не використовує складну систему активації цілей.

Нечіткі архітектурні моделі мультіагентних систем

Формальні системи трансформацій нечітких графів (СТНГ) можуть використовуватися при проектуванні нечітких агентів з двома цілями: 1) як узагальнення алгоритмів нечіткого логічного виводу, зокрема, таких, як системи Мамдані і Ларсена над нечіткими графами; 2) з метою специфікації можливих перетворень між моделями, що задають різні рівні абстракції нечітких агентів у підході MDA (так звані "вертикальні" трансформації або перетворення "уточнення").

Кожна продукція СТНГ будується на основі нечітких графів [4,5], які являють собою антецедент і граф-результат продукції відповідно. Така продукція, може бути записана як вказано нижче:

ЯКЩО нечіткий граф-антецедент
ТО нечіткий граф-результат.

Можливість вказаного представлення продукцій СТНГ засновано на тому факті, що нечіткий граф-антецедент і нечіткий граф-результат (консеквент) можуть бути записані у виді

$$\tilde{N}_1 \& \tilde{N}_2 \dots \tilde{N}_N \& \tilde{E}_1 \& \tilde{E}_2 \& \dots \tilde{E}_M,$$

де $\{N_1, N_2, \dots, N_N\}$ – множина вершин нечіткого графа, $\{E_1, E_2, \dots, E_M\}$ – множина ребер нечіткого графа, а $\tilde{N}, \tilde{E}$ – нечітка множини вершин і ребер нечіткого графа відповідно. Позначимо належність вершини і ребра графу-антецеденту і графу-консеквенту відповідно верхніми індексами A і C .

Таким чином, проста продукція, яка задає відповідне перетворення, може бути записана у виді

ЯКЩО

$$\tilde{N}_1^A = \tilde{N}_1^A \& \dots \& \tilde{N}_N^A = \tilde{N}_N^A \& \tilde{E}_1^A = \tilde{E}_1^A \& \dots$$

$$\dots \& \tilde{E}_M^A = \tilde{E}_M^A \text{ ТО}$$

$$\tilde{N}_1^C = \tilde{N}_1^C \& \dots \& \tilde{N}_K^C = \tilde{N}_K^C \& \tilde{E}_1^C = \tilde{E}_1^C \& \dots$$

$$\dots \& \tilde{E}_L^C = \tilde{E}_L^C,$$

де $\tilde{N}_i^A, \tilde{E}_j^A$ – значення істинності (нечіткі числа) вершин і ребер нечіткого графа до якого застосовується продукція, що зазвичай відрізняється від значень вказаних в продукції; $\tilde{N}_k^C, \tilde{E}_l^C$ – виправлені значення істинності нечітких множин графа, який отримується в результаті застосування продукції.

Нами реалізовано три типи виведення над нечіткими графами: монотонне, при якому послідовні значення істинності вершин і ребер можуть лише зростати; немонотонне, при якому послідовні значення істинності можуть як зростати, так і зменшуватися; і низхідне монотонне, при якому послідовні значення істинності можуть лише зменшуватися. Адекватність застосування певного типу виведення може відрізнятися для різних продукцій, що відповідають певним етапам рішення задачі.

У реалізації нечіткого агента, заснованого на мотивації, використовуються системи СТНГ для того, щоб безпосередньо відображувати стимули X на відповіді R . Вибір цього способу заснований на простоті наявності єдиної системи нечітких правил, що забезпечують функцію відображення β , оскільки цей вибір не вимагає механізму координації. Відсутність координатора поведінки усередині самого агента спрощує його реалізацію і робить очевиднішим ефект впливу нечітких значень спонукання на результуючу поведінку агента.

В разі чітких значень вхідних стимулів $x = \{x_1, \dots, x_n\}$ фузифікатор наводить їх до відповідних нечітких значень. Блок нечіткого виведення здійснює зіставлення нечітких правил СТНГ не лише з нечіткими стимулами $\tilde{x} = \{\tilde{x}_1, \dots, \tilde{x}_n\}$, але і з нечіткими значеннями мотивації $\tilde{m} = \{\tilde{m}_1, \dots, \tilde{m}_N\}$. Для цього задана проєкція нечіткого графа предметної області (ПО) з набору нечітких правил на нечітку множину спонукань, тобто, відображення, що визначає які з вершини графа ПО можуть розглядатися як мотиви (спонукання). При використанні низхідного нечіткого виведення висота функції належності мотивації в правій частині правила може лише зменшуватися, що приведе до зупинки виконання СТНГ через декілька ітерацій зважаючи на відсутність необхідних спонук (міра нечіткої збіжності менше необхідного порогу). Дефузифікатор j -го агенту відображує множину вершин нечіткого графа ПО, які відповідають нечітким значенням

поведінки на множині одночасних дій $\beta_j = \{\beta_{j1}, \dots, \beta_{ji}\}$.

Моделі координації і адаптації з врахуванням нечітких значень корисності

На рис. 1 показано архітектуру нечіткої мультиагентної системи, що складається з N мотивованих агентів. Всі агенти отримують однакові стимули $\tilde{x} = \{\tilde{x}_1, \dots, \tilde{x}_m\}$. Кожний агент здійснює застосування правил СТНГ і виробляє рішення відносно прийнятної поведінки $\beta_j = \{\beta_{j1}, \dots, \beta_{ji}\}$. Вироблення остаточного рішення здійснюється координатором, завданням якого також є також налаштування нечітких правил. С цією метою, модуль оцінки пропозицій агентів на основі $\beta_1, \beta_2, \dots, \beta_N$, набору відповідних цим пропозиціям вхідних стимулів і набору мотивацій здійснює налаштування правил СТНГ для кожного агента і передає отриманий результат модулю обчислення результату. В результаті зіставлення з використанням індексів нечіткого ранжирування виробляється остаточне вирішення мультиагентної системи, що означає "найкращу" запропоновану поведінку $\beta_i = \{\beta_{i1}, \dots, \beta_{ii}\}$.

У нашій реалізації координатора мультиагентної системи початкові установки спонукання використовуються для того, щоб на основі системи нечітких правил Такагі-Сугено визначити нечітку придатність поведінки агента в різних середовищах. Ці спонуки використовуються як вхідні нечіткі числа для запуску кожного експерименту. Для можливості такого налаштування користувач вибирає значення мотивацій (тобто $m_1, \dots, m_N$) які мають бути використані при нечіткому обчисленні корисності (тобто спонуки, які допомагають вибрати кращого агента в популяції) та одне з допустимих середовищ навчання агента.

Модуль оцінювання пропозицій агентів здійснює наступні кроки: на основі генетичного алгоритму генерує популяцію різних мотивованих агентів; кожен агент популяції у свою чергу виробляє і виконує послідовність дій β_j у середовищі; на основі запропонованої поведінки кожного агента координатор генерує набір N значень корисності; результуюча нечітка корисність F обчислюється на основі мотивів $m_1, \dots, m_N$ і окремих значень корисності $f_1, \dots, f_N$.

Оцінювання закінчується, коли досягнуте число максимальних повторень генетичного алгоритму. Кращий агент в популяції

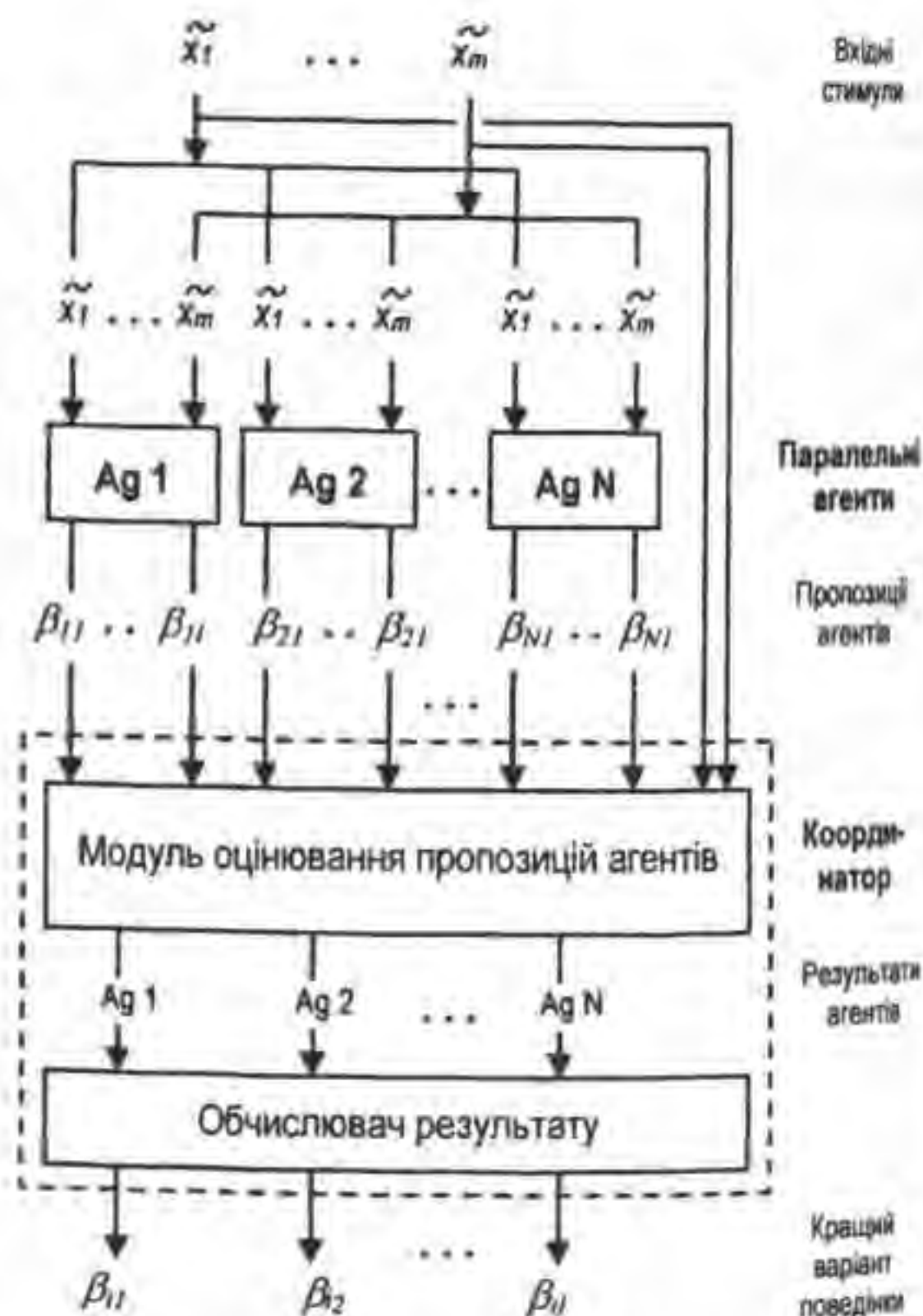


Рис.1 Структура нечіткої мультиагентної системи з координатором

зберігається як завершальний варіант агента. Для визначення нечіткого значення корисності для специфічної поведінки, використовується система Такагі-Сугено, що не вимагає подальшої дефuzифікації. Результатом кожного з правил є чіткі значення, що підсумовуються як зважена середня величина [13].

В процесі навчання, кероване середовище обране, і генетичний алгоритм випадковим чином генерує початкову популяцію агентів. Після цього, кожен агент виконує своє завдання (купівля і продаж) і виробляється набір значень корисності, що відповідають виконаному завданню. Нарешті, на основі значень спонукання (мотивації) і набутих значень корисності за окремими критеріями, координатор обчислює остаточне значення нечіткої корисності.

Основні метамоделі та трансформації

Моделе-орієнтована розробка мультиагентних систем значною мірою залежить від послідовної трансформації їх моделей [14]. При цьому найбільш важливими є наступні способи перетворення моделей:

• Перетворення модель - модель (M2M). Цей вид перетворення використовується для трансформації одного типу графічної моделі (схеми, діаграми) до іншого типу графічної моделі. Перетворення M2M ґрунтується на початковій та цільовій метамоделях і визначає перетворення елементів початкової моделі в елементи цільової моделі.

• Перетворення модель-текст (M2T). Такі перетворення використовуються для трансформації візуального представлення в код програми; синтаксис цільової мови визначається разом із метамоделлю графічної моделі.

В основу моделі-орієнтованої розробки нечітких мультиагентних систем покладено їх метамодель, які створені з використанням формату Ecore [15]. Засоби опису метамodelей Ecore призначені для підтримки часу виконання, такої як повідомлення про зміни, засоби для збереження моделей із вбудованими способами серіалізації на основі XML, і надають ефективний інтерфейс програміста для операцій над об'єктами Ecore. При цьому компонент Ecore Tools забезпечує середовище для операцій над моделями, у тому числі перевірку правильності і створення генераторів.

Розроблені трансформації створені на основі фреймворку Eclipse Modeling Framework [15], що використовує для побудови моделей метамодель Ecore. Вона визначає, що модель складається з екземплярів типу EClass, які можуть володіти атрибутами (екземплярами типу EAttribute) або посиланнями на інші екземпляри EClass (через тип EReference). Нарешті, атрибути можуть бути різними примірниками EDataType (наприклад, цілими числами, рядками, дійсними числами, тощо).

Аналогічна технологія для представлення і обробки метамodelей - Meta-Object Facility (MOF) [16]. Тим не менш, метамодель EMF простіша, ніж метамодель MOF з точки зору понять, властивостей і внутрішньої структури, таким чином, відображення понять EMF в поняття MOF може бути проведено відносно просто.

Вихідною при трансформаційній розробці агентів може служити їх рольова модель (RM). Передбачається, що інтелектуальний агент зобов'язується виконувати одну або декілька ролей у перебігу свого життєвого циклу. Однією з особливостей розробки агентів є те, що види діяльності агента пов'язані з роллю, а не з системою. Крім того, після визначення можливостей агентів і розкладання їх на прості

діяльності, необхідно визначити динамічну композицію цих видів діяльності за кожною роллю таким чином, що агент досягає поставленої мети.

Метамодель рольової моделі визначає поняття "Роль", яке посиляється на поняття:

• Діяльність, що включає два атрибути, ім'я (назва) і функціональність (опис того, що зазначена діяльність виконує),

• Можливість, що відноситься до напрямів діяльності на які вона посиляється, що досягають високорівневої мети,

• Протокол, якому приписується власне ім'я.

Трансформація рольової моделі (RM) у цільову модель - автоматизована задача, яка створює кілька початкових моделей внутрішнього агентного управління (MBAU) на основі раніше створеної RM, по одній для кожної ролі. Для розробки такої трансформації модель-модель (M2M) використано мову Query-View Transformation (QVT).

Певні методології розробки (такі як Tropos або INGENIAS [2,3]) нав'язують специфічні деліберативні моделі агента. На відміну від них, модель MBAU заснована на нечітких схемах станів і не робить ніяких додаткових припущень про такі аспекти як модель комунікації, процес виведення або ментальні стани (наприклад, переконання-бажання-намір) агентів.

Метамодель MBAU, що представлена на рис. 2, визначає поняття "Модель", яке містить нечіткі стани, переходи і значення. Імена моделі мають задовольняти сучасному формату простору імен пакетів Java або C#. Нечіткі стани містять наступні атрибути: назва стану, його тип у схемі станів (як правило, одне із значень I, ABO, ПОЧАТОК, КІНЕЦЬ), мітка стану, і діяльність, що пов'язана з станом.

Стани та переходи також посиляються на нечіткі значення, які в свою чергу включають атрибут "значення" - у діапазоні [0,1]. Наступним поняттям, що визначене у цій метамоделі є "Перехід", який, окрім посилання на нечітке значення, включає також наступні атрибути:

• ім'я, зазвичай у формі <мітка початкового стану> <мітка цільового стану>;

• вираз переходу. Через вирази переходів (ВП) визначається основна інформація стосовно управління в MBAU. Цей вираз містить умови та події, які роблять перехід можливим. Крім того, у виразах може бути зазначено події отримання або передачі повідомлень між агентами (у разі взаємодіючих агентів).

• джерело, початковий стан, і мета, цільовий стан.

Таким чином, кожна модель МВАУ уточнюється шляхом визначення умов та подій прийому/передачі повідомлень, які дозволять переходи від однієї діяльності (задачі) до іншої.

Модель МВАУ задається нечіткою схемою переходів, що може бути представлена нечітким мультиграфом $G = (Q, T, \mu_G, \Sigma)$, де $Q = \{q_1, q_2, \dots, q_n\}$ – множина станів (вершин графа), де q_1 – початковий стан; $T = \{t_1, t_2, \dots, t_n\}$ – множина переходів (ребер) графа, $\mu_G : T \rightarrow [0, 1]$ – функція належності переходів даної схеми, $\Sigma : T \rightarrow A$ – функція розмітки переходів подіями з деякого кінцевого алфавіту подій (повідомлень) A .

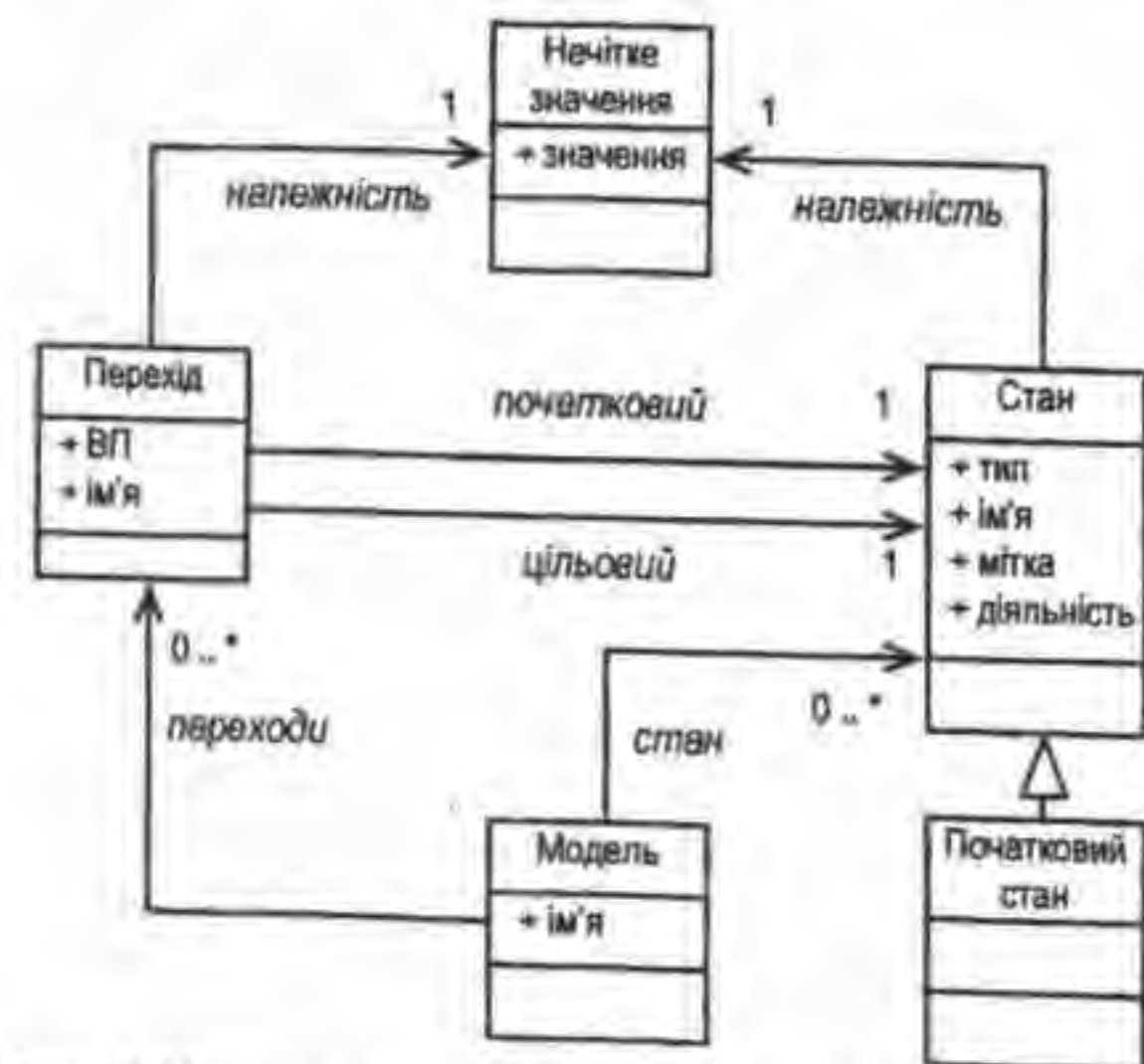


Рис. 2 Метамоделі внутрішнього агентного управління

Спосіб подання належностей переходів фактично визначає структуру нечіткої схеми переходів. Припустимо, що така схема включає n станів, ступінь належності її переходів може бути представлена у вигляді тривимірної матриці

$$P = \begin{bmatrix} P_{11} & P_{12} & \dots & P_{1n} \\ P_{21} & P_{22} & \dots & P_{2n} \\ \dots & \dots & \dots & \dots \\ P_{n1} & P_{n2} & \dots & P_{nn} \end{bmatrix} \quad (1)$$

де $P_{ij} = [\mu_{ij}^1, \mu_{ij}^2, \dots, \mu_{ij}^m]$ – вектор ступеня належності переходів між вершинами i та j , що розмічені відповідно символами a^k , причому $\mu_{ij}^k \in [0, 1]$, а символи в розмітці не

повторюються. Ненульові значення μ_{ij} відповідають переходам станів $q_j = \delta(q_i, a_{ij}^k, \mu_{ij}^k)$, в той час як $\mu_{ij} = 0$ означає, що між двома станами переходу не існує. Правильна схема переходів має задовольняти наступній умові: в одному й тому ж векторі символи подій відрізняються один від одного ($a^k \neq a^l$ для усіх $k \neq l$) та належать кінцевому алфавіту подій A .

Визначимо нечіткий (розмитий) стан як вектор пар

$$Q = [\langle q_1, \mu(q_1) \rangle, \langle q_2, \mu(q_2) \rangle, \dots, \langle q_n, \mu(q_n) \rangle] \quad (2)$$

другий компонент яких $\mu(q_i)$ – ступінь належності агента даному стану. Реакцією (виконанням) схеми переходів у відповідь на настання події a^k є новий стан

$$Q_{Q \otimes P} = [\langle q_1, \mu_{Q \otimes P}(q_1) \rangle, \langle q_2, \mu_{Q \otimes P}(q_2) \rangle, \dots, \langle q_n, \mu_{Q \otimes P}(q_n) \rangle]$$

такий, що

$$\mu(q_i) = \max\{\mu(q_j) * \mu_{ij}^k\} \quad (\forall \langle q_j, \mu(q_j) \rangle \in Q).$$

Послідовне виконання (поведінка), що задається парою подій a^k, a^l і схемою нечітких переходів відповідає $\max$ -* композиції нечітких відношень $P \otimes P$:

$$\mu_{ij} = \max\{\mu_{ir}^k * \mu_{rj}^l\} \quad (\forall \mu_{ir}^k \in P_{ir}, \mu_{rj}^l \in P_{rj}, r = 1, \dots, n).$$

Діяльності, що задаються схемою з n станами можуть бути безпосередньо представлені вектором:

$$D = [d_1, d_2, \dots, d_n],$$

де d_j – діяльність, що виконується в j -му стані. Як правило, виконується тільки діяльність, ступінь належності стану якої – найбільша серед усіх станів.

Таким чином, будь-яка нечітка схема переходів із n станами може бути представлена матрицею належності $n \times n$ (1) і вектором стану (2).

Однією з цікавих можливостей налаштування нечіткої схеми переходів є можливість її навчання на основі представлених прикладів і спеціально розробленої еволюційної стратегії.

Основні трансформації моделей описано з використанням СТНГ як платформи-незалежної моделі агента, для чого створено спеціальний графічний редактор із засобами навігації по нечітких графах і композиції правил заміни (видалення і додавання вершин, вибір початкової і цільової метамodelей тощо). Породження мультіагентної системи складається з двох етапів – виконання ряду перетворень модель-модель з метою здобуття платформно-специфічної моделі нечітких агентів та запуску подальших трансформацій “модель-код” для здобуття тексту програми цільового агента.

Для представлення СТНГ розроблена метамодель Ecore, що містить всі необхідні поняття для конструювання нечітких правил. Фрагмент метамodelі, який відноситься до представлення нечітких правил, представлений на рис. 3.

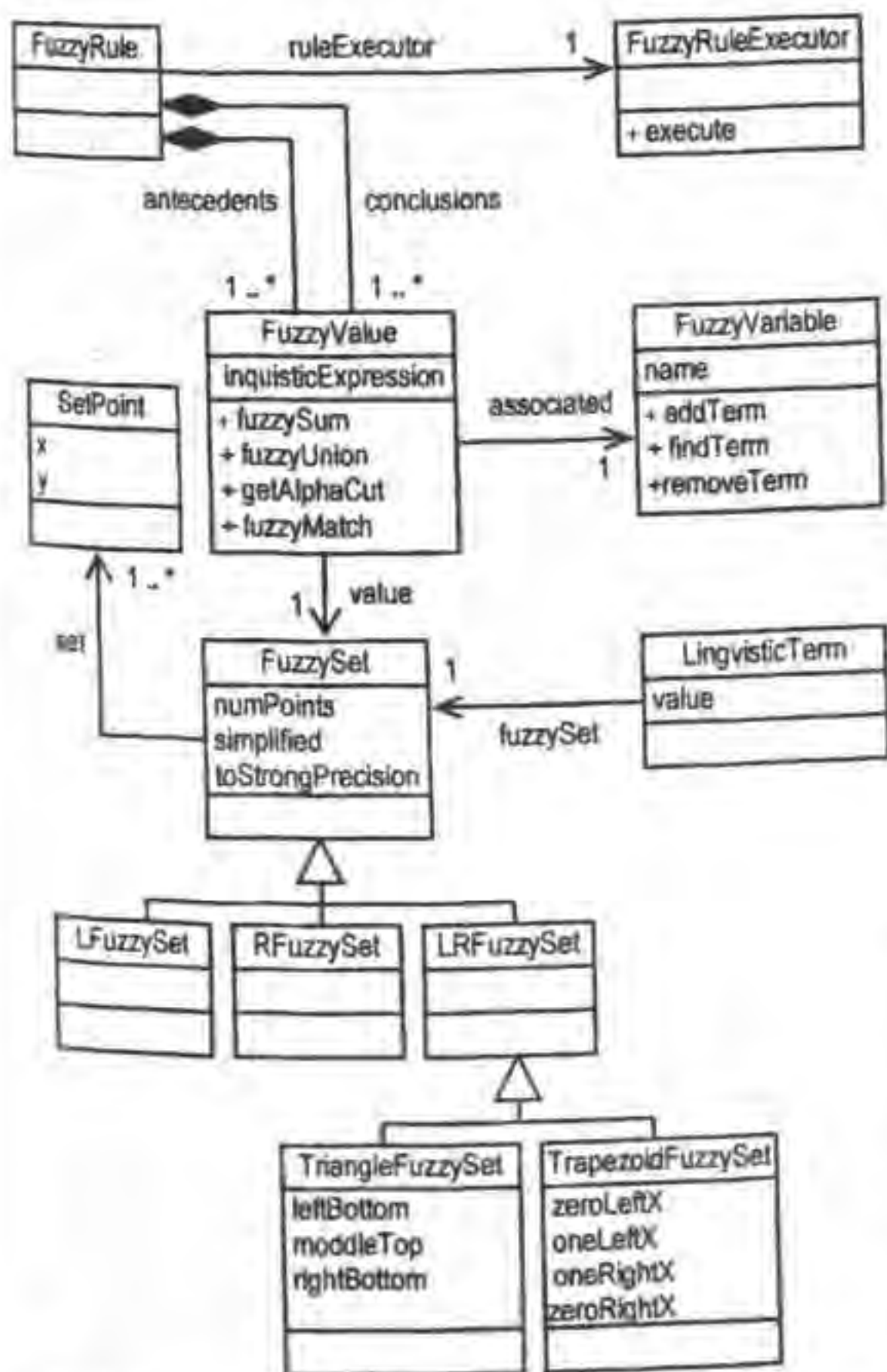


Рис. 3 Фрагмент метамodelі для представлення нечітких правил

Як основні класи (поняття) вибрані FGraph (нечіткий граф), InModel (вхідна модель), OutModel (вихідна модель), FNode та FEdge

(нечіткі вершини і ребра). Кожній вершині або ребру відповідає нечітке значення, що пов'язане у свою чергу з нечіткою змінною, яка описує набір припустимих лінгвістичних значень і їх функцій належності. За допомогою спеціальної трансформації СТНГ перетворюється на модель нечітких правил, що називаються також “нечіткими системами” і знаходяться ближче за рівнем до цільової платформи. Різні способи виконання нечітких правил враховуються за допомогою конкретних класів FuzzyRuleExecutor, наприклад, клас LarsenRuleExecutor відповідає системі Ларсена. Це дозволяє розширювати дану метамодель іншими видами нечітких систем, що описані в літературі, не зачіпаючи при цьому рівень СТНГ.

Проте, засоби представленої вище метамodelі знаходяться досить далеко від засобів програмування мультіагентної цільової платформи, таких як JADE [17] або ABLE [18]. Тому для інтеграції нечітких правил в середовище виконання вказаних платформ використовуються перетворення, яка відображають нечіткі правила в модель на основі JEM (Java EMF Model), що дозволяє представити основні конструкції класів мови Java і платформи J2EE.

Для генерації текстових файлів програм-агентів розроблені перетворення “модель-текст” на основі текстових шаблонів платформи XPand, компонентів, що запускаються в контексті EMFT Workflow [19]. В порівнянні з безпосередньою генерацією коду на основі СТНГ, використання проміжних моделей значно спрощує створення текстових шаблонів.

Висновки

Таким чином, нами запропоновані нечіткі архітектурні моделі мультіагентних систем, які призначені для вирішення задач у розподіленому середовищі. Такі моделі мають властивості як реактивних, так і мотивованих деліберативних агентів, що дозволяє використовувати переваги обох підходів. Формалізовані засоби опису поведінки нечітких агентів на основі трансформацій нечітких графів, способи координації нечіткої мультіагентної системи і її адаптації з врахуванням нечітких значень корисності.

У рамках моделі-орієнтованого підходу розроблені основні перетворення, що дозволяють згенерувати нечіткі платформно-незалежні моделі внутрішнього агентного управління. Запропонований трансформаційний підхід

охоплює всі основні фази розробки мультиагентних систем (від вимог до реалізації), перехід однієї фази до іншої здійснюється за допомогою трансформацій моделей, основу яких складають нечіткі схеми. У результаті, моделі

кожної з фаз збагачують інформацією, що поступово приводить до конкретної реалізації адаптивних інтелектуальних агентів.

Список використаних джерел

1. Заде Л.А. Роль мягких вычислений и нечеткой логики в понимании, конструировании и развитии информационных/интеллектуальных систем. – Новости Искусственного Интеллекта, №2-3, 2001, с. 7 – 11.
2. Perini A., Susi A. Automating model transformations in agent-oriented modeling // Agent-oriented software engineering VI. Lecture notes in computer science. – Springer, 2006. – P.167-178.
3. Pavon J., Gomez-Sanz J. Agent-oriented software engineering with INGENIAS // Multi-agent system and applications III. Lecture notes in computer science. – Springer, 2003. – P.8-38.
4. Парасюк І.М., Єршов С.В. Методи аналізу програмних архітектур, представлених нечіткими графовими моделями // Проблеми програмування. – 2006. – № 1-2. – С. 101-110.
5. Єршов С.В. Нечеткие графы функциональных зависимостей как основа метамоделирования программных систем // Компьютерная математика. – 2005. – № 3. – С.139-149.
6. Єршов С.В. К проблеме формализации объектно-ориентированных методов разработки программного обеспечения на основе нечеткой логики // Там же. – 2003. – № 2. – С. 62-77.
7. Рассел С., Норвиг П. Искусственный интеллект. Современный подход: Вильямс, 2007. – 1408 с.
8. Wooldridge M.J. An Introduction to Multiagent Systems. – Cambridge: MIT Press, 2002. – 366 p.
9. Bussmann S., Jennings N., Jennings N.R., Wooldridge M.J. Multiagent systems for manufacturing control: a design methodology. – 2004. – 288 p.
10. Luck M.M., Ashri R., D'Inverno M. Agent-based software development, 2004. – 208 p.
11. Anumba C.J., Ugwu O. O., Ren Z. Agents and multi-agent systems in construction. – 2005 – 329 p.
12. Plekhanova V. Intelligent agent software engineering. – 2003. – 240 p.
13. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. – М.: Горячая линия – Телеком, 2004. – 452 с.
14. Sendall S., Kozaczynski W. Model transformation: the heart and soul of model-driven software development // IEEE Software. – 2003. – Vol.20, № 5. – P.42-45.
15. Eclipse Modeling Framework. – <http://wiki.eclipse.org/EMF/>
16. OMG MetaObject Facility. – <http://www.omg.org/mof/>
17. Bellifemine F., Caire G., Greenwood D. Developing multi-agent systems with JADE. – New York: John Wiley & Sons, 2007. – 286 p.
18. Bigus J. P., Schlosnagle D. A., Pilgrim J. R. ABLE: A toolkit for building multiagent autonomic systems // IBM Systems Journal. – Vol. 41, № 3. – 2002. – P.350-371.
19. Eclipse Modeling Framework Technology (EMFT). – <http://www.eclipse.org/modeling/emft/project=mwe>.

Надійшла до редколегії 08.12.10

УДК 004.3

Погорілий С.Д., д. т. н., проф.

До методології проектування алгоритмічного забезпечення новітніх паралельних архітектур обчислювальних систем

Виконано аналіз SMP- та MPP-архітектур суперкомп'ютерних систем та виділено підклас систем із гнучкою архітектурою за рахунок їх динамічної реконфігурації. Запропоновано методи створення гнучких архітектур на основі новітніх технологій. Проаналізовано можливість застосування платформи CUDA та новітньої архітектури Fermi для суперкомп'ютерних обчислень. На основі проведеного аналізу обґрунтовано доцільність формалізованого проектування паралельних застосувань для різних архітектур суперкомп'ютерних систем та парадигм паралельного програмування на основі математичного апарату модифікованих систем алгоритмічних алгебр.

Ключові слова: суперкомп'ютерні системи, SMP- та MPP-архітектури, платформа CUDA, системи алгоритмічних алгебр.

*E-mail: sdp@rpd.univ.kiev.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Висновки експертів у галузі комп'ютерних технологій збігаються в тому, що ресурс екстенсивного зростання продуктивності комп'ютерних систем внаслідок підвищення складності та тактової частоти мікропроцесорів себе вичерпав. В той же час існує велика різноманітність архітектур, що можуть забезпечити суттєве підвищення швидкодії алгоритмів, і важливим класом серед них є паралельні. В подальших дослідженнях та розробках акцент перенесено на створення новітніх паралельних архітектур, а саме: SMP-, MPP-, CUDA-архітектур (в тому числі Fermi) тощо, що обумовлює створення паралельних алгоритмів. В роботі висвітлено певні напрями проектування алгоритмічного і відповідного програмного забезпечення стосовно новітніх паралельних архітектур комп'ютерних систем.

MPP-архітектура

Кластерні технології стали логічним продовженням розвитку ідей, закладених в MPP-

S. D. Pogorilyy, Dr.Sci., prof.

Algorithm software design methodology for modern parallel architectures of computer systems.

SMP-MPP-architectures and supercomputer systems were analyzed and subclass of systems with flexible architecture due to their dynamic reconfiguration was selected. Methods to create flexible architectures based on advanced technologies have been proposed. The possibility of application CUDA platform and Fermi architecture for HPC was analyzed. Based on the analysis of the problem expediency of formal planning of applications for various parallel architectures of supercomputer systems and parallel programming paradigms based on mathematical tools of modified systems of algorithmic algebras have been substantiated.

Key Words: HPC, SMP-MPP-architectures, CUDA platform, modified systems of algorithmic algebras.

архітектурах. Зараз розповсюдженою практикою стає створення кластерних систем (систем зі слабим зв'язком), що об'єднують обчислювальні потужності комп'ютерів окремих лабораторій або комп'ютерних класів. На сьогоднішній день активно формуються наукові методи, спрямовані на створення гнучкого механізму керування потужним комп'ютерним ресурсом, який дозволяє реалізувати його багатофункціональне застосування: кластерні обчислення (кластерні системи із гнучкою архітектурою) або розподілені обчислення у клієнт-серверних архітектурах. Таке дуальне застосування комп'ютерного ресурсу дозволяє суттєво підвищити його ефективність та підняти обчислювальний потенціал установи.

Для створення кластерної системи із гнучкою архітектурою розглянемо найбільш доцільні сучасні апаратні та програмні платформи. Це в першу чергу платформи Windows HPC Server 2008 (WHPC) та попередня версія – Windows Compute Cluster Server (WCCS). В ролі комунікаційного середовища, як правило,

обирають технологію Ethernet.

Для формування гнучких програмованих кластерних архітектур [1] доцільно використання технологій: PXE, віртуалізації від компанії Intel (Intel Virtualization) та Intel vPro в поєднанні з протоком iSCSI. Зв'язок методів формування кластерних систем з гнучкою архітектурою в

залежності від обраної технології ілюструється рис. 1.

Використання технології PXE передбачає створення додаткового програмного забезпечення, організаційну структуру якого наведено на рис. 2.



Рис. 1. Кластерні системи з гнучкою архітектурою.

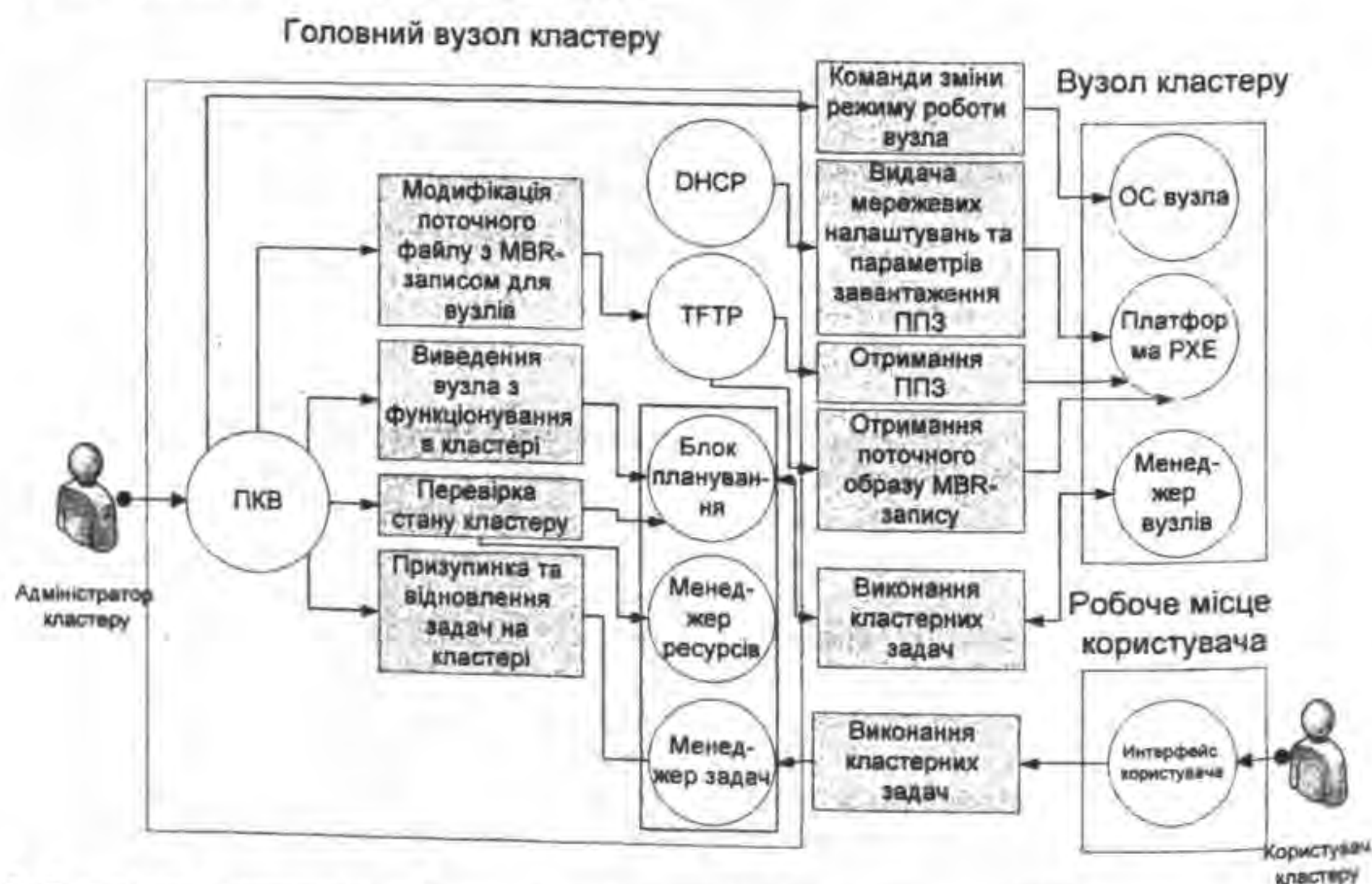


Рис. 2. Структура програмного забезпечення кластеру на базі технології PXE.

Кластер з гнучкою архітектурою створено та впроваджено в комп'ютерному класі академії Майкрософт інформаційно-обчислювального центру Київського національного університету імені Тараса Шевченка.

Реалізація кластеру з гнучкою архітектурою з використанням технології віртуалізації від компанії Intel дозволяє одночасно реалізувати режими проведення кластерних та розподілених

обчислень у клієнт-серверних архітектурах. Одночасна робота цих режимів можлива за рахунок паралельного функціонування двох операційних систем на одній апаратній платформі, рис. 3. Використання такого підходу доцільне при наявності комп'ютерного ресурсу на основі багатоядерних мікропроцесорів (SMP-архітектур). Реалізація запропонованої архітектури також була здійснена в Київському

національному університеті імені Тараса Шевченка дозволяють реалізувати кластерні системи із двома рівнями паралелізму: на рівнях вузлів та кластера в цілому.

Комп'ютерні системи MPP-архітектури з окремими вузлами у вигляді SMP-архітектур



Рис. 3. Архітектура кластеру при використанні технології віртуалізації.

CUDA-архітектура

Новітня програмно-апаратна платформа CUDA [2] надає можливість використання значних ресурсів відеоадаптера в цілях, не пов'язаних з обробкою зображення, наприклад, для виконання математичних розрахунків. Для отримання значного збільшення швидкодії програми на GPU (Graphics Processing Unit – графічний процесор, відеоадаптер) в порівнянні з центральним процесором програміст має досконально знати та використовувати архітектуру графічного процесора, раціонально розраховувати розміщення алгоритму в пам'яті відеоадаптера, чого вимагає специфіка архітектури графічного процесора: неправильне використання пам'яті викличе численні цикли простою GPU. Продуктивність GPU значно підвищує продуктивність комп'ютера за рахунок великої кількості потокових процесорів, які надають можливість виконувати одночасно значну кількість потоків однієї програми.

Обчислювальна архітектура CUDA заснована на концепції паралелізму за даними: одна команда - багато даних (SIMD - Single Instruction Multiple Data) і понятті мультипроцесора. Мультипроцесор - це багатоядерний SIMD-процесор, що дозволяє в кожний певний момент часу виконувати на всіх ядрах тільки одну інструкцію. Кожне ядро мультипроцесора скалярне, тобто воно не підтримує векторні операції в чистому вигляді. Системи MPP-архітектури з окремими вузлами у вигляді SMP-архітектур та відеоадаптерами CUDA дозволяють реалізувати кластерні системи із

трьома рівнями паралелізму: на рівнях відеоадаптерів, вузлів та кластера в цілому.

Архітектура CUDA має розвинену ієрархію системи пам'яті: регістрову, локальну, глобальну, колективну, константну та текстурну.

Регістрова пам'ять (для одного мультипроцесора існує 8192 32-розрядних регістра);

Локальна пам'ять (у випадках, коли локальні дані процедур займають занадто великий розмір, або компілятор не може обчислити для них певний постійний крок при зверненні, він може помістити їх у локальну пам'ять);

Глобальна пам'ять (існує можливість довільної адресації глобальної пам'яті. Можна читати та писати в будь-яку комірку пам'яті (в загальному випадку на GPU це не так. За універсальність доводиться розплачуватися швидкістю. Глобальна пам'ять не керується і працює дуже повільно, кількість звернень до неї слід в будь-якому випадку мінімізувати. Її доцільно використовувати, в основному, для збереження результатів роботи програми перед відправленням їх до центрального процесора);

Колективна пам'ять (це швидка пам'ять, її рекомендується використовувати як керовану кеш-пам'ять);

Константна пам'ять (кешується, причому кеш існує в єдиному примірнику для одного мультипроцесора. З пристроїв константна пам'ять доступна тільки для читання);

Текстурна пам'ять (керується, для кожного мультипроцесора є тільки один кеш, а значить, цей кеш загальний для всіх задач всередині

блоку).

Архітектура CUDA нового покоління з назвою *Fermi-архітектура* - це найбільш просунута архітектура в історії обчислень на GPU, яка пропонує низку абсолютно нових можливостей. 512 ядер CUDA дозволяють новітній архітектурі Fermi забезпечувати суперобчислення і високу продуктивність із мінімальними витратами в порівнянні з традиційними GPU-серверами.

Архітектура Fermi забезпечує більш широке використання гетерогенних обчислень на GPU і центральних процесорах, підтримуючи повний спектр обчислювальних застосувань: вбудовані можливості роботи з C++ і сумісність із середовищем розробки Visual Studio роблять паралельне програмування для архітектури Fermi більш простим і дозволяють досягати неймовірно високої продуктивності роботи в великій кількості застосувань, включаючи значне прискорення високоточних наукових обчислень, роботи з розрідженими матрицями в лінійній алгебрі, сортування та пошукових алгоритмів.

Парадигми паралельного програмування

Наявність різноманітності архітектур паралельних комп'ютерних систем неминує призвести до формування значної кількості парадигм паралельного програмування. На сьогоднішній день найбільш поширені парадигми паралельного програмування включають:

процес-орієнтовану (паралельне застосування формується у вигляді композиції процесів як сукупності відокремлених ресурсів [3]);

потік-орієнтовану (паралельне застосування формується у вигляді сукупності потоків як окремих складових процесів [4]);

MPI (Message Passing Interface, інтерфейс передачі повідомлень. Фактично це специфікація API, що ґрунтується на сутності процесу і дозволяє паралельно працюючим комп'ютерам обмінюватись повідомленнями [5]);

Java-технології (майже нічим не відрізняються від потік-орієнтованої парадигми, але принципова відмінність полягає у тому, що застосування створюється мовою Java і працює на віртуальній Java-машині [6]);

технології платформи .NET (застосування створюється будь-якою мовою цієї платформи і працює на віртуальній машині .NET [7]) тощо.

Зазначені парадигми паралельного

програмування дозволяють у більшості випадків здійснювати інтуїтивне інженерне проектування паралельних застосувань.

Формалізоване проектування паралельних алгоритмів

Складність архітектур паралельних обчислювальних систем, різноманітність парадигм паралельного програмування суттєво ускладнюють, а інколи унеможливають інженерне проектування паралельних алгоритмів складних проектів і висувають актуальну задачу формалізованого проектування таких алгоритмів і асоційованих із ними програм.

Одним з підходів до такого проектування є орієнтація на застосування математичного апарату модифікованих систем алгоритмічних алгебр (САА-М), що є різновидом систем алгоритмічних алгебр (САА) [8,9]. Доцільність цього підходу обґрунтовується такими міркуваннями:

математичний апарат САА-М у порівнянні з класичними алгоритмічними системами орієнтований на проектування паралельних алгоритмів (та асоційованих з ними програм);

будь-який алгоритм, записаний в САА-М, являє собою алгебраїчну формулу, що дозволяє застосовувати до цього алгоритму систему достатньо розвинених в САА-М формальних еквівалентних трансформацій з метою оптимізації і подальшого автоматизованого синтезу програмного забезпечення для різних архітектур паралельних комп'ютерних систем;

САА-М відповідають концепції структурного програмування;

наявністю позитивного досвіду наукового колективу під керівництвом автора формування паралельних схем алгоритмів широкого кола застосувань [3-7].

У процесі розвитку досліджень особлива увага приділялась аналізу запису алгоритмів за допомогою математичних співвідношень та програм, записаних послідовними мовами. Підхід пояснюється тим фактом, що саме в цьому вигляді записано найбільшу кількість світового багажу алгоритмічних знань.

Розглянемо процес формалізованого проектування з використанням САА-М на прикладі формування паралельного алгоритму Флойда_Уоршала (дозволяє знаходити всі шляхи на графі між парами вершин) [9].

Послідовна схема алгоритму має вигляд:

$$(k=1) * \underbrace{\{ \text{InitTemp}^*(i=1) * \}_{\alpha_i}}_{\alpha_k} \underbrace{\{ \}_{\alpha_j}}_{\beta_{(k-1)ik} \wedge \beta_{(k-1)kj}} \left(\underbrace{[(A_D \vee B_D) * (A_P \vee B_P)] \vee [A_D * A_P]}_{\gamma_+} \right) * \underbrace{\{ \}_{\alpha_j}}_{\gamma_-} * \text{incJ} * \text{incI} * \text{incK},$$

де i, j, k – допоміжні змінні;

InitTemp – оператор ініціалізації;

$\text{incl}, \text{incJ}, \text{incK}$ – оператори інкременту;

$$A_D \equiv D_{i,j}^k = D_{i,j}^{k-1},$$

$$A_P \equiv P_{i,j}^k = P_{i,j}^{k-1},$$

$$B_D \equiv D_{i,j}^k = D_{i,j}^{k-1} + D_{k,j}^{k-1},$$

$$B_P \equiv P_{i,j}^k = P_{k,j}^{k-1};$$

– основні оператори

схеми алгоритму;

α_i – альфа-умова;

β_k – умова перевірки елементів матриці;

γ_+ – умова вибору найкоротшого шляху;

$D_{i,j}^k, P_{i,j}^k$ – матриці вихідних даних та передування відповідно.

З метою отримання параметричних паралельних схем алгоритму в [9] виконано низку еквівалентних трансформацій наведеної схеми, в результаті чого отримано паралельну схему алгоритму:

$$\text{InitDE} * (k=1) * \underbrace{\{ \text{InitTemp}^* [\bigvee_{y=1}^p [\bigvee_{x=1}^q (\text{Sync}_{y,x}^i(\vec{\alpha}) * (i_{y,x} = b_{y-1,x}) * \text{InitDE}_{x,y} * \text{DE}_{x,y} * \underbrace{\{ (j_{x,y} = a_{x-1,y}) * \}_{\alpha_{jyx}}} \underbrace{\{ X_{qxy} \}}_{\alpha_{jxy}} * \text{incl}_{y,x} \} * \text{DelDE}_{x,y}]] * \text{incK} \}}_{\alpha_k} * \text{DelDE}, \quad (1)$$

де $\text{InitDE}, \text{DelDE}$ – оператори створення та знищення каналу обміну даними відповідно;

$\text{Sync}_x(\vec{\alpha})$ – оператор синхронізації;

$\vec{\alpha}$ – вектор умов синхронізації;

X – складений оператор, $X = (B \vee A) * \text{incJ}$.

Наступним кроком до пришвидшення схеми алгоритму (1) є внесення оператора ініціалізації тимчасових даних і альфа-ітерації по k до

асинхронних диз'юнкцій. Останній крок до створення остаточної параметричної регулярної схеми алгоритму є застосування дистрибутивної властивості:

$(A_1 \vee A_2) \vee (A_3 \vee A_4) = A_1 \vee A_2 \vee A_3 \vee A_4$. Тоді отримуємо остаточної варіант схеми паралельного алгоритму:

$$\text{InitDE} * \underbrace{\{ \bigvee_{y=1}^p [\bigvee_{x=1}^q ((k=1) * \underbrace{\{ \text{InitTemp}^* \text{Sync}_{y,x}^i(\vec{\alpha}) * (i_{y,x} = b_{y-1,x}) * \text{InitDE}_{x,y} * \text{DE}_{x,y} * \underbrace{\{ (j_{x,y} = a_{x-1,y}) * \}_{\alpha_{jyx}}} \underbrace{\{ X_{qxy} \}}_{\alpha_{jxy}} * \text{incl}_{y,x} \} * \text{DelDE}_{x,y} * \text{incK} \}}_{\alpha_k}]] * \text{DelDE}. \quad (2)$$

Для систем із розподіленою пам'яттю $\text{InitDE}, \text{DelDE}$ можуть бути "одиничними" операторами (наприклад, для MPI-парадигми). Для систем із спільною пам'яттю ці оператори створюють необхідну область спільного користування. Важливо відзначити, що схема (2) є параметризованою як для SMP-, так і для MPP-архітектур. Конкретна версія визначається реалізацією операторів синхронізації.

Висновки

Складність та розвиненість новітніх архітектур комп'ютерних систем із масовим паралелізмом, створення складних ієрархічних систем пам'яті вимагають для досягнення

максимальної продуктивності роботи застосувань раціонального проектування паралельних алгоритмів та асоційованих із ними програм стосовно до певних архітектур.

Різноманітність новітніх архітектур із багатьма рівнями паралелізму та парадигм паралельного програмування суттєво ускладнюють, а інколи унеможливають інженерне проектування паралельних алгоритмів складних систем.

Існуючий дуалізм у галузі проектування паралельних алгоритмів може бути ефективно розв'язаний тільки за рахунок вирішення наукової проблеми формалізованого проектування алгоритмів і асоційованих із ними програм.

Список використаних джерел

1. Погорілий С. Д., Бойко Ю. В., Грязнов Д. Б. та ін. Концепція створення гнучких гомогенних архітектур кластерних систем. Матеріали шостої міжнародної науково-практичної конференції з програмування УкрПРОГ'2008. Проблеми програмування, 2008, №2-3, с. 84-90.
2. Погорелый С.Д., Бойко Ю.В., Трибрат М.И. и др. Анализ методов повышения производительности компьютеров с использованием графических процессоров и программно-аппаратной платформы CUDA. Математичні машини і системи, 2010, №1, с. 40 – 54.
3. Pogorilyy S.D. Y, Gusarov A.D. Paralleling Of Edmonds-Karp Net Flow Algorithm. Applied and Computational Mathematics, 5 (2006), no.2, pp.121-130.
4. Погорілий С.Д., Гусаров А.Д. Паралельна реалізація алгоритму Едмондса-Карпа. Доповіді Національної академії наук України, 2008, № 10, с. 46-51.
5. Pogorilyy S. D., Boyko Yu. V., Gusarov A. D. An approach to the parallel solution of a high-dimensional basic flow problem. Cybernetics and Systems Analysis, Volume 45, Issue 2 (March 2009) pages: 291 – 296. Springer Science and Business Media Inc. ISSN:1060-0396.
6. Бойко Ю.В., Погорілий С.Д., Шкуліна І.Ю. Дослідження паралельних схем алгоритму Прима. Математичні машини і системи, 2007, №2, с. 77 – 89.
7. Погорілий С.Д., Равський О.С. Дослідження віртуальної паралельної машини .NET Framework на прикладі алгоритму Габова. Наукові праці ДНТУ, серія Інформатика, кібернетика і обчислювальна техніка, вип. 9(132), Донецьк, 2008, с. 21-26.
8. Ющенко Е.Л., Цейтлин Г.Е., Грицай В.П., Терзян Т.К. "Многоуровневое структурное проектирование программ" М., "Финансы и статистика", 1989г. 192с.
9. Погорілий С.Д., Мар'яновський В.А., Вітель Д.Ю. та ін. Формування узагальнених паралельних схем алгоритму Флойда-Уоршала. Системні дослідження та інформаційні технології, 2010, №1, с. 52 – 68.

Надійшла до редколегії 07.12.10

УДК 519.6

Пряничникова О.О.*

Алгебра мов, що можуть бути представлені в помічених графах

В останні роки в комп'ютерних науках інтенсивно використовуються скінченні орієнтовані графи з поміченими вершинами. В даній статті ми вводимо алгебру мов, що можуть бути представлені в таких графах, і досліджуємо її властивості. На відміну від алгебри регулярних мов, яка здебільшого використовується для графів з поміченими дугами (скінченних автоматів), запропонована алгебра є зручним засобом для дослідження властивостей мов, що можуть бути представлені в графах з поміченими вершинами. В статті для графів з поміченими вершинами доведено теореми, що є аналогами теореми Кліні і теореми Майхіла-Нерода для скінченних автоматів. Розроблено методи аналізу та синтезу мов, що можуть бути представлені в графах з поміченими вершинами, способи детермінізації та мінімізації таких графів. Для введеної алгебри запропонована скінченна система аксіом, доведена її повнота.

Ключові слова: граф, аналіз, синтез, детермінізація, мінімізація.

*E-mail: pois@suiai.edu.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

В останні роки в комп'ютерних науках інтенсивно використовуються не лише графи з поміченими дугами (автомати), але і графи з поміченими вершинами [1]. В теоретичному програмуванні такими графами є блок-схеми програм. У робототехніці графи з поміченими вершинами використовуються для представлення операційного середовища в задачах навігації мобільних роботів: задачі самокалібрації, тобто відрізнення деякої вершини відомого графа від усіх інших його вершин, і задачі контролю карти, тобто відрізнення графа-еталону від заданого класу графів [2]. Подібні графи під назвою «модель Кріпке» використовуються в дослідженнях з надійності програмних систем [3, 4].

E.A. Pryanichnikova*

An Algebra of Languages Representable in Vertex-Labelled Graphs

Recently directed finite vertex-labelled graphs have been successfully applied to the diverse areas of computer science, robotics, etc. In this paper we introduce and study an algebra of languages representable by vertex-labelled graphs. In contrast to Kleene algebra of regular languages, which is mainly used for edge-labelled graphs, it can adequately represent many properties of languages generated by vertex-labelled graphs. We prove an analog of Kleene's theorem establishing equivalence of regular expressions in this algebra and vertex-labelled graphs and an analog of the Myhill-Nerode theorem giving the necessary and sufficient conditions for the languages to be representable by a class of directed vertex-labelled graphs. We give several basic constructions, including methods for obtaining of a regular expression in this algebra from vertex-labelled graph and vice versa, determinization and state minimization of vertex-labelled graphs. A finitary axiomatization for considered algebra is developed and its completeness is proved.

Key Words: graph, analysis, synthesis, determinization, minimization.

У зв'язку з широким використанням цього типу графових моделей, є актуальною задача вивчення їх властивостей, і, зокрема, властивостей мов, що можуть бути в них представлені. В даній роботі мови, що можуть бути представлені в графах з поміченими вершинами, вивчаються шляхом дослідження нової алгебри, подібної до алгебри регулярних мов, але з особливостями (наприклад, у властивостях операції ітерації), які дозволяють, на наш погляд, більш адекватно описувати властивості мов, що можуть бути представлені в графах з поміченими вершинами. Розробка даної алгебри викликана, зокрема, тим, що завдання мов, що можуть бути представлені в графах з поміченими вершинами, у вигляді регулярних виразів алгебри Кліні неадекватно відображає структуру цих мов і викликає труднощі переходу від графів до формул і навпаки.

Робота таку. В першому розділі вводяться основні визначення. В другому розділі розглянуто основні властивості алгебри мов, що можуть бути представлені в графах з поміченими вершинами. Третій розділ присвячено задачам аналізу і синтезу графів з поміченими вершинами. В четвертому розділі розглянуто питання детермінізації та мінімізації графів з поміченими вершинами.

Основні визначення

Скінченим орієнтованим графом з поміченими вершинами назвемо четвірку $G = (Q, E, X, \mu)$, де Q – скінченна множина вершин, $|Q| = n$, $E \subseteq Q \times Q$ – множина дуг, X – скінчений алфавіт поміток вершин, $\mu: Q \rightarrow X$ – функція розмітки. Нехай $I \subseteq Q$ – множина початкових, а $F \subseteq Q$ – множина фінальних вершин графа.

Вершини q_1 і q_2 називатимемо суміжними, якщо $(q_1, q_2) \in E$. Шляхом у графі G називатимемо скінченну послідовність суміжних вершин $l = q_1 q_2 \dots q_k$, де $(q_i, q_{i+1}) \in E$, q_1 – початкова вершина шляху, q_k – заключна вершина. Число $k-1$ називатимемо довжиною шляху. Нехай X^* – множина всіх непустих слів скінченної довжини в алфавіті X .

Елемент $x = \mu(q_1) \mu(q_2) \dots \mu(q_k) = x_1 x_2 \dots x_k \in X^*$ називатимемо поміткою шляху l в графі G . Для кожної вершини q визначимо шлях нульової довжини, що розпочинається та закінчується в цій вершині. Поміткою нульового шляху вважатимемо $x \in X$, де $\mu(q) = x$.

Множина поміток всіх шляхів, початковою вершиною яких є вершина q_1 , а заключна вершина належить до множини фінальних вершин, назвемо мовою, породженою вершиною q_1 .

Мовою, породженою графом G , назвемо помітки всіх шляхів в графі G , початкові вершини яких належать до множини I , а заключні – до множини F . Мову, породжену графом G , позначимо $L(G)$.

Якщо в графі G є шляхи $l = q_1 q_2 \dots q_k$ та $m = q_k q_{k+1} \dots q_m$, у яких заключна вершина першого шляху співпадає з початковою вершиною другого шляху, то результатом поєднання таких шляхів буде шлях $q_1 q_2 \dots q_k q_{k+1} q_{k+2} \dots q_m$. До такої операції поєднання

шляхів відповідає часткова бінарна операція склеювання слів (поміток шляхів), що визначається на множині X^* наступним чином: для всіх $w_1, w_2 \in X^*$ і всіх $x, y \in X$

$$w_1 x \circ y w_2 = \begin{cases} w_1 x w_2, & \text{якщо } x = y; \\ \text{не визначено,} & \text{інакше.} \end{cases}$$

Введемо на мовах $L, R \in 2^{X^*}$ операції:

- 1) об'єднання $L \cup R = \{w | w \in L \text{ або } w \in R\}$;
- 2) конкатенація $L \cdot R = \{w_1 w_2 | w_1 \in L \text{ і } w_2 \in R\}$;
- 3) склеювання $L \circ R = \{w_1 \circ w_2 | w_1 \in L \text{ і } w_2 \in R\}$;

- 4) ітерація $L^* = \bigcup_{i=0}^{\infty} L^i$, де $L^0 = X$;

$$L^{n+1} = L^n \circ L; n \geq 0;$$

- 5) позитивна ітерація $L^+ = \bigcup_{i=1}^{\infty} L^i$, де $L^0 = X$;

$$L^{n+1} = L^n \circ L; n \geq 0;$$

- 6) $L^\circ = \bigcup_{i=0}^{\infty} L^i$, де $L^0 = L_{beg} \circ L_{end}$; $L^1 = L$;

$$L^{n+1} = L^n \circ L; n \geq 1$$

$$L_{beg} = \{x | xw \in L, x \in X, w \in X^*\}$$

$$L_{end} = \{x | wx \in L, x \in X, w \in X^*\}.$$

Розглянемо алгебру $\langle 2^{X^*}, \circ, \cup, \otimes, \emptyset, X \rangle$.

Регулярні вирази в цій алгебрі визначимо індуктивно наступним чином:

1) $\emptyset$ є регулярним виразом, що представляє мову $\emptyset$;

2) x і xu є регулярними виразами та представляють мови $L(x) = \{x\}$ і $L(xu) = \{xu\}$ для всіх $x, u \in X$;

3) Якщо p і q – регулярні вирази, що представляють мови $L(p)$ і $L(q)$ відповідно, то вирази $(p \circ q)$, $(p \cup q)$, (p°) також є регулярними, і $L(p \circ q) = L(p) \circ L(q)$, $L(p \cup q) = L(p) \cup L(q)$, $L(p^\circ) = (L(p))^\circ$.

Таким чином, регулярними виразами алгебри $\langle 2^{X^*}, \circ, \cup, \otimes, \emptyset, X \rangle$ є всі терми сигнатури $\sigma = (\circ, \cup, \otimes, \emptyset, X)$, що породжені алфавітом $X \cup X^2$, де $X^2 = \{xu | x \in X, u \in X\}$.

Властивості алгебри мов, що можуть бути представлені в графах з поміченими вершинами

Розглянемо основні властивості операцій
алгебри $\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$.

Операція $\cup$ комутативна і асоціативна,
оскільки за визначенням співпадає з теоретико-
множинною операцією об'єднання;
 $R \cup \emptyset = \emptyset \cup R = R$ та $R \cup R = R$ для будь-якого
 $R \in 2^{X^+}$, тому відносно операції $\cup$ множина
 2^{X^+} являє собою комутативний ідемпотентний
моноїд, в якому роль нейтрального елементу
виконує пуста множина $\emptyset$.

Відносно операції $\circ$ множина 2^{X^+} є моноїд,
нейтральним елементом якого є мова X .

Доведемо, що операція $\circ$ асоціативна,
тобто $P \circ (R \circ Q) = (P \circ R) \circ Q$ для будь-яких мов
 $P, Q, R \in 2^{X^+}$.

Нехай $w \in P \circ (R \circ Q)$. За визначенням
операції $\circ$ слово w має вигляд $w_1 x w_2 y w_3$, де
 $w_1 \in P$, $w_2 \in R$, $w_3 \in Q$, $x \in X$, $y \in X$. Але це
означає, що слово w належить до мови
 $(P \circ R) \circ Q$.

Таким чином, $P \circ (R \circ Q) \subset (P \circ R) \circ Q$.
Аналогічно доводиться $(P \circ R) \circ Q \subset P \circ (R \circ Q)$.

Оскільки $P \circ (R \circ Q) \subset (P \circ R) \circ Q$ і
 $(P \circ R) \circ Q \subset P \circ (R \circ Q)$ для будь-якого $w \in X^+$,
то $P \circ (R \circ Q) = (P \circ R) \circ Q$. За визначенням
операції $\circ$ для будь-якого $R \in 2^{X^+}$ виконується
 $R \circ X = X \circ R = R$, тому алгебра $\langle 2^{X^+}, \circ, X \rangle$ є
моноїдом.

Операціям $\circ$ і $\cup$ задовольняють закони
дистрибутивності:

$$R \circ (Q \cup P) = R \circ Q \cup R \circ P;$$

$$(R \cup Q) \circ P = R \circ P \cup Q \circ P.$$

Доведемо, що $R \circ (Q \cup P) \subset R \circ Q \cup R \circ P$.
Нехай $w \in R \circ (Q \cup P)$. За визначенням
 $w = w_1 \circ w_2$, де $w_1 \in R$, $w_2 \in Q \cup P$. Якщо
 $w_2 \in Q \cup P$, то або $w_2 \in Q$, або $w_2 \in P$.

В тому випадку, коли $w_2 \in Q$, з того, що
 $w = w_1 \circ w_2$ і $w_1 \in R$, $w_2 \in Q$ випливає, що
 $w \in R \circ Q$.

У випадку коли $w_2 \in P$ з того, що
 $w = w_1 \circ w_2$ і $w_1 \in R$, $w_2 \in P$ випливає, що

$w \in R \circ P$. Тоді $w \in R \circ Q \cup R \circ P$, звідки
 $R \circ (Q \cup P) \subset R \circ Q \cup R \circ P$.

Аналогічно доводиться, що
 $R \circ Q \cup R \circ P \subset R \circ (Q \cup P)$.

Наслідком двох включень є
 $R \circ (Q \cup P) = R \circ Q \cup R \circ P$.

Тотожність $(R \cup Q) \circ P = R \circ P \cup Q \circ P$
доводиться аналогічно.

Із розглянутих властивостей операцій
випливає що алгебра $\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$ є
ідемпотентне напівкільце. Нехай $\leq$ –
відношення природного часткового порядку
напівкільця, тобто $P \leq R$ тоді і тільки тоді, коли
 $P \cup R = R$ для будь-яких $P, R \in 2^{X^+}$.

Замкнутість напівкільця $\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$
випливає із існування об'єднання будь-якої сім'ї
множин, що являє собою супремум цієї сім'ї, а
також із неперервності операції $\circ$ [5].

Введена алгебра $\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$, як і
алгебра $\langle 2^{X^+}, \cup, *, \emptyset, \lambda \rangle$ регулярних мов в
алфавіті X , являє собою замкнуте ідемпотентне
напівкільце з однією додатковою операцією [5].
На відміну від алгебри регулярних мов, введена
алгебра не є алгебра Кліні: легко перевірити, що
вона не задовольняє її аксіомам [6]. Основні
відмінності алгебри $\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$ від
алгебри $\langle 2^{X^+}, \cup, *, \emptyset, \lambda \rangle$ пов'язані з тим, що,
на відміну від конкатенації, операція $\circ$ часткова.

Якщо в алгебрі Кліні $p \cdot q = \emptyset$ тільки тоді,
коли $p = \emptyset$ або $q = \emptyset$, то в алгебрі
 $\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$ можна підібрати приклади,
коли це не вірно: нехай $L(p) = \{ab\}$, $L(q) = \{cd\}$,
тоді $p \circ q = \emptyset$.

Ще одна важлива відмінність полягає в тому,
що в алгебрі Кліні множина R^* завжди включає
пусте слово та є нескінченною для будь-якого R ,
а в алгебрі $\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$ можна навести
приклади множин, для яких результат
застосування операції $\otimes$ є нескінченною,
скінченною або пустою множиною: якщо
 $L(p) = \{aba\}$, то $L(p^{\otimes}) = \{a, aba, ababa, \dots\}$;

якщо $L(p) = \{ab\}$ то $L(p^{\circ}) = \{ab\}$; якщо $p = \emptyset$, то $p^{\circ} = \emptyset$.

Оскільки алгебра $\langle 2^{X^+}, \circ, \cup, *, \emptyset, X \rangle$ не являє собою алгебру Кліні, виникла необхідність створити нові еквівалентні співвідношення для цієї алгебри, що розкривають властивості її операцій. В результаті отримана скінченна система аксіом (1), що складається з тотожностей і квазітотожностей, яка є суттєво відмінною від аксіоматики алгебр Кліні.

$$\begin{aligned}
 & p \cup q = q \cup p \\
 & p \cup \emptyset = p \\
 & p \cup p = p \\
 & p \circ (q \circ r) = (p \circ q) \circ r \\
 & X \circ p = p \\
 & p \circ X = p \\
 & p \circ (q \cup r) = p \circ q \cup p \circ r \\
 & (p \cup q) \circ r = p \circ r \cup q \circ r \\
 & \emptyset \circ p = \emptyset \\
 & p \circ \emptyset = \emptyset \\
 & p^{\circ} \cup X \geq p^{\circ} \circ p \cup p \cup X \\
 & p^{\circ} \cup X \geq p \circ p^{\circ} \cup p \cup X \\
 & q \cup p \circ y \leq y \Rightarrow p^{\circ} \circ q \cup q \leq y \\
 & q \cup y \circ p \leq y \Rightarrow q \circ p^{\circ} \cup q \leq y \\
 & p \leq X \Rightarrow p \circ p = p
 \end{aligned} \tag{1}$$

Система аксіом (1) повна для алгебри $\langle 2^{X^+}, \circ, \cup, *, \emptyset, X \rangle$, а саме: два регулярних вирази p і q над скінченним алфавітом X позначають одну і ту саму мову, що може бути представлена в графі, тоді і тільки тоді, коли формула $p = q$ є логічним наслідком аксіом (1).

Для доведення того, що у випадку, коли із аксіом логічно випливає, що $p = q$, для регулярних виразів p і q виконується $L(p) = L(q)$, достатньо показати, що всі аксіоми мають таку властивість, що мови, які представляють регулярні вирази у лівих частинах тотожностей, співпадають із мовами їх правих частин.

Для доведення того, що в тому випадку, коли $L(p) = L(q)$, формула $p = q$ являє собою наслідок аксіом (1), можна показати, що для будь-якого терму p вільної алгебри багатовиду, що описується тотожностями і

квазітотожностями (1), можливо знайти відповідну систему рівнянь таким чином, що лінійна комбінація рішень цієї системи дає регулярний вираз p' , який представляє ту саму мову, що і початковий терм, причому із аксіом (1) випливає, що $p = p'$. Шляхом введення порядку на множині всіх слів X^+ можливо добитися того, що для всіх регулярних виразів, що представляють одну і ту саму мову, відповідні коефіцієнти системи рівнянь будуть однакові. Тоді можливо показати, що для будь-яких регулярних виразів p і q , що представляють одну мову, із (1) випливає рівність регулярних виразів, що відповідають рішенням систем для p і q , тобто $p' = q'$, звідки випливає, що $p = q$ також є наслідком (1).

Задачі аналізу і синтезу для графів з поміченими вершинами

Найважливішими задачами, пов'язаними з дослідженням графів з поміченими вершинами, є задача аналізу графа, тобто знаходження регулярного виразу, що представляє мову, породжену цим графом, і задача синтезу, тобто побудова графа, що породжує мову, яка може бути представлена даним регулярним виразом [5].

Рішення цих задач базується на наступній теоремі, що в певному сенсі аналогічна відомій теоремі Кліні, яка стверджує, що клас мов, що розпізнаються скінченними автоматами, співпадає з класом мов, які можна представити регулярними виразами алгебри регулярних мов $\langle 2^{X^+}, \cup, *, \emptyset, \lambda \rangle$.

Теорема 1. Для будь-якої мови $L \subseteq X^+$ еквівалентні твердження:

1) L – мова, що може бути представлена регулярним виразом алгебри $\langle 2^{X^+}, \circ, \cup, *, \emptyset, X \rangle$;

2) L – мова, що може бути представлена графом з поміченими вершинами.

Розв'язання задачі аналізу для графів з поміченими вершинами ґрунтується на тому, що мова, яка може бути представлена графом з поміченими вершинами, також може бути представлена системою лінійних рівнянь. Для таких систем запропоновано метод розв'язання, який базується на використанні відомої теореми про нерухому точку для замкнутих на півкільці

[1]. В результаті застосування цього методу для будь-якого графа G можна отримати регулярний вираз, для якого $L(R) = L(G)$.

Для рішення задачі синтезу виявилось необхідним ввести алгебру матриць над елементами розглянутої алгебри і матричне зображення графів з поміченими вершинами, яке суттєво відрізняється від матричного зображення скінченних автоматів, що розглядається в роботах [6, 7].

Твердження. Клас мов, що можуть бути представлені регулярними виразами алгебри $\langle 2^{X^*}, \cup, *, \emptyset, X \rangle$, співпадає з класом всіх регулярних мов, що не містять в собі пустого слова.

Дане твердження витікає з того, що в роботі [1] було показано, що будь-яка мова, що може бути представлена графом з поміченими вершинами, може бути представлена регулярним виразом алгебри $\langle 2^{X^*}, \cup, *, \emptyset, \lambda \rangle$, а з теореми і випливає, що мова може бути представлена графом з поміченими вершинами тоді і тільки тоді, коли вона може бути представлена регулярним виразом алгебри $\langle 2^{X^*}, \cup, *, \emptyset, X \rangle$.

Детермінізація і мінімізація графів з поміченими вершинами

Граф $G = (Q, E, X, \mu)$ назовемо детермінованим, якщо для будь-якої пари $(q_i, x_i) \in Q \times X$ знайдеться не більше однієї вершини $q_j \in Q$, для якої справедливо, що $(q_i, q_j) \in E$ і $\mu(q_j) = x_i$. Якщо для будь-якої пари $(q_i, x_i) \in Q \times X$ в графі G знайдеться не менше однієї вершини $q_j \in Q$, для якої справедливо, що $(q_i, q_j) \in E$ і $\mu(q_j) = x_i$, то граф G назовемо повним.

Теорема 2. Для будь-якого недетермінованого графа G , що породжує мову $L(G)$, існує такий повний детермінований граф G' , що $L(G) = L(G')$.

Доведення теореми 2 базується на методі, подібному до конструкції підмножин для скінченних автоматів [6].

Слід зазначити, що є суттєва різниця між детермінованими графами з поміченими вершинами і детермінованими скінченними автоматами.

У повного детермінованого автомата є лише один початковий стан і лише один «пустий» стан. У повного детермінованого графа з поміченими вершинами початкових і «пустих вершин» стільки, скільки символів в алфавіті X .

Відрізняються і оцінки кількості вершин після детермінізації. Якщо у вихідного не детермінованого автомата кількість станів дорівнювала n , то після застосування алгоритму конструкції підмножин для його детермінізації у отриманого автомата у найгіршому випадку буде 2^n станів.

При застосуванні алгоритму детермінізації графів з поміченими вершинами, кількість вершин повного детермінованого графа, отриманого в результаті, в найгіршому випадку буде дорівнювати $|Q| = \sum 2^{|Q_i|}$, де через Q_i позначена множина всіх вершин вихідного не детермінованого графа G з поміткою x_i .

В той же час і у повних детермінованих графів, і у повних детермінованих автоматів є важлива спільна властивість, на наявності якої ґрунтуються подальші міркування: для будь-якого слова з X^+ в них завжди є єдиний шлях, що веде із початкової вершини, помітка якого співпадає з цим словом.

Визначимо похідну мови $L \subseteq X^*$, що може бути представлена графом з поміченими вершинами, по слову $w \in X^+$ як $D_w L = \{y | w \circ y \in L\}$.

Для мови $L \subseteq X^*$ визначимо відношення $\rho_L \subseteq X^* \times X^*$ наступним чином: для всіх $u, v \in X^*$ $u \rho_L v$ тоді і тільки тоді, коли $D_u L = D_v L$, і слова u та v закінчуються одним і тим самим символом $x \in X$. Легко перевірити, що це відношення є рефлексивним, симетричним, транзитивним та право-інваріантним.

Теорема 3. Мова $L \subseteq X^*$ може бути представлена графом з поміченими вершинами тоді і тільки тоді, коли відношення ρ_L поділяє X^* на скінченне число класів еквівалентності.

Дана теорема дає необхідну і достатню умову того, що мова може бути представлена в графі з поміченими вершинами (аналогічно теоремі Майхілла-Нерода, яка дає необхідну і достатню умову того, що мова регулярна, тобто розпізнається скінченним автоматом). Наслідком цієї теореми є існування канонічних форм для графів з поміченими вершинами та регулярних виразів, що представляють мови таких графів:

будь-якій мові, що може бути представлена в графі з поміченими вершинами, відповідає єдиний з точністю до ізоморфізму повний детермінований граф з мінімальною кількістю вершин та єдина система ліво-лінійних рівнянь, із рішень якої може бути отриманий регулярний вираз, що представляє дану мову.

Для графів з поміченими вершинами запропоновано алгоритми для вирішення наступних задач: побудови для графа G детермінованого графа, що породжує мову $L(G)$; побудови для будь-якого детермінованого графу з поміченими вершинами графа G повного детермінованого графу з поміченими вершинами, що породжує мову, $L(G)$ з мінімальним числом вершин.

Таким чином, для алгебри мов, що можуть бути представлені в графах з поміченими вершинами, отримані аналоги всіх основних результатів, відомих для алгебри регулярних мов і скінченних автоматів. Це дає змогу використовувати введену алгебру

$\langle 2^{X^+}, \circ, \cup, \otimes, \emptyset, X \rangle$ під час аналізу графів з поміченими вершинами так, як алгебра регулярних мов використовується для скінченних автоматів.

Висновки

В даній роботі досліджені основні властивості алгебри мов, що можуть бути представлені в графах з поміченими вершинами: знайдені алгебраїчна і лінгвістична характеристики мов, що можуть бути представлені регулярними виразами цієї алгебри, розроблені методи аналізу і синтезу розглядуваних мов та способи детермінізації і мінімізації графів з поміченими вершинами.

Список використаних джерел

1. Капитонова Ю.В. Математическая теория проектирования вычислительных систем / Ю.В. Капитонова, А.А. Лещичевский. – Москва: Наука, 1988. – 296 с.
2. Dudek G. Map validation and robot self-location in a graph-like world / G. Dudek, M. Jenkin, E. Milios, D. Wilkes // Robotics and autonomous systems. – 1997. – Vol. 22(2). – P. 159-178.
3. Baier C. Principles of Model Checking / C. Baier, J.-P. Katoen. – Cambridge: MIT Press, 2008. – 975 p.
4. Clarke E. M. Model Checking / E. M. Clarke, O. Grumberg, D. Peled – Cambridge: MIT Press, 1999. – 257 p.
5. Белоусов А.И. Дискретная математика / А.И. Белоусов, С.Б. Ткачев. – Москва: Изд-во МГТУ им. Н.Э. Баумана, 2004. – 744 с.
6. Conway J.H. Regular Algebra and Finite Machines / J.H. Conway – London: Chapman and Hall, 1971. – 147 p.
7. Грунский И.С. Об алгебре языков, представимых графами с отмеченными вершинами / И.С. Грунский, Е.А. Пряничникова // Труды Ин-та прикл. математики и механики НАН Украины. – 2009. – т.18. – С. 37-46.
8. Yu.S. Regular Languages / S. Yu // Handbook of Formal Languages. – Berlin: Springer, 1997. – P. 41-110

Надійшла до редколегії 15.12.10

УДК 007.51 : 681.518.25

Тараненко К.Г., к.т.н., Коломійченко Т.В.,
к.т.н., Бочарников В.П., д.т.н., проф.

Автоматизація підтримки прийняття рішень у реальному часі в доказовій медицині

У статті окреслені проблеми прийняття
рішень у сучасній медицині в реальному часі в
умовах невизначеності характеристик, що
вивчаються, а також обмеженості первинних
даних. Показана перспективність сумісного
застосування Fuzzy-технологій із методами та
засобами поглибленої аналітики (Data Mining)
для розв'язання даних проблем. Запропонована
структурна схема автоматизованої системи
моделювання, прогнозування і підтримки
ухвалення рішень, заснованих на доказах.

Ключові слова: доказова медицина,
підтримка прийняття рішень, поглиблена
аналітика, автоматизація.

*E-mail: k.taranenko@gmail.com

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Основні тенденції розвитку технічних і
біомедичних наук в даний час визначаються
фактором глобалізації і, як наслідок, широким
спектром використання методів і засобів
наукових досліджень. Інформаційні потоки, що
формується в результаті, викликають проблему
отримання науково обґрунтованих висновків при
прийнятті рішень в умовах невизначеності
характеристик, що вивчаються, а також
обмеженості первинних даних. Крім того, існує
необхідність зниження впливу суб'єктивних
чинників на вибір стратегій і тактик медичної
діяльності [1].

Оптимізація ухвалених рішень з погляду
найефективнішої практики пов'язана з
необхідністю автоматизації у реальному часі
процесів пошуку, відбору, інтерпретації і
інтеграції надійної інформації, одержаної від
пацієнтів, спостережуваної клініцистами і
сформованої дослідниками. Це формує потребу в
практичній реалізації системних підходів
відповідно до нового напрямку медичної науки і
практики - доказової медицини (Evidence-Based
Medicine), зокрема, в розв'язанні задач
автоматизації пошуку і аналізу доказової
інформації у реальному часі.

K.G. Taranenko, Ph.D., T.V. Kolomiichenko,
Ph.D., V.P. Bocharnikov., Dr., prof.

Automation of the decision's support in the real time in the Evidence-Based Medicine

In the article there are the outlined problems of
decisions acceptance in modern medicine in the
real time in the conditions of vagueness of studied
descriptions and also narrow-mindedness of basic
data. Shown perspective of compatible application
of Fuzzy-technologies with methods and facilities of
deep analytics (Data Mining) for the decision
support of these problems. Offered flow diagram of
the automated system of design, prognosis and
support of acceptance of the decisions based on
evidence.

Key words: evidence-based medicine, decision
support systems, data mining, automation.

До недавнього часу процес автоматизації,
пов'язаний з необхідністю підтримки в
актуальному стані правил і умов логіки
ухвалення рішень, був достатньо трудомістким.
А через швидкі зміни внутрішнього і
зовнішнього середовища також виникали
численні проблеми, пов'язані з необхідністю
постійного внесення змін в моделі і правила,
розроблені раніше і «жорстко» запрограмовані у
відповідних системах [2].

Завдяки розвитку методів і засобів
поглибленої аналітики (Data Mining) на основі
web-сервісів є можливість забезпечити
приймання від користувачів запитів на обробку, з
одного боку, а з іншого - доставку рішень в точки
взаємодії з ними. При цьому рішення,
сформовані такого роду системами, повинні
ґрунтуватися на спеціально розроблених моделях
і технологіях їх реалізації, що враховують
специфіку предметної області. Прикладом
подібної технології є рішення SAS® Real-Time
Decision Manager (RTDM), що інтегрує на єдиній
платформі різні програмні продукти, які
використовуються для рішення різних задач
планування, реалізації і контролю прийняття
управлінських рішень [3].

Алгоритм підтримки прийняття у реальному часі рішень, заснованих на доказах, включає наступні основні етапи.

1. Вибір предметної області, постановка задачі, визначення проблеми (функція модуля ідентифікації).

2. Пошук необхідної інформації у внутрішніх і зовнішніх джерелах і базах даних з урахуванням понятійного апарату даної предметної області (пошуковий модуль).

3. Оцінка відповідності моделей прогнозування і ухвалення рішень, що використовуються, природі наявних даних, в т.ч. з урахуванням їх достовірності (модуль аналізу і моделювання).

4. Отримання результатів моделювання і визначення їх доказовості з урахуванням користі, збитку і витрат по кожному з методів (модуль реалізації).

5. Оцінка можливості застосування результатів моделювання на практиці (модуль прогнозу).

6. Прогнозування наслідків застосування результатів моделювання і вибір кращої стратегії реалізації з точки зору ефективності методів (модуль оптимізації).

У реальних умовах традиційні підходи до прогнозування зазвичай або виключають невизначеність з моделей, що розробляються (що неправомірно, оскільки невизначеність є невід'ємною характеристикою будь-якого прогнозу), або не враховують можливу різноманітність видів і чинників невизначеностей: невизначеності початкових даних, зовнішнього середовища, невизначеності, пов'язаної з характером, варіантами і особливостями реалізації моделей ухвалення рішень, невизначеністю бажаних і цільових умов прогнозування і т.д.

З іншого боку, абсолютно точні прогнози одержати практично ніколи неможливо, оскільки на будь-який процес, що моделюється і аналізується, постійно впливає безліч чинників, які визначають ризики прогнозування (наприклад, у вигляді загрози ускладнень внаслідок неправильно ухваленого рішення, або пов'язаних з цим додаткових витрат).

Отже, існує нерозривний зв'язок трьох основних категорій, урахування і розгляд яких необхідні при автоматизації підтримки процесів ухвалення рішень, заснованих на доказах, а саме: *прогноз - невизначеність - ризик*.

В умовах, коли первинна інформація є неточною і неповною, а швидкість або час

отримання оцінок дуже критичні (тобто чим більше невизначеність прогнозу, тим більше ризик ухвалення неправильного рішення) – нечітка логіка практично не має альтернатив.

Для підтримки прийняття рішень, заснованих на доказах, в системі крім методів і засобів Data Mining передбачається використовувати також Fuzzy-технології, які дозволяють працювати з історичними, накопиченими в базах даних, і оперативними даними [4]. Зазначені технології є альтернативою (і доповненням) до традиційних методів аналізу, моделювання і прогнозування, оскільки дозволяють природним для користувача системи чином формалізувати початкові дані, що не формалізуються іншими методами; описати невизначеності будь-яких видів; провести прогнозні розрахунки в умовах невизначеності, кількісно оцінити ризики будь-якого роду і управляти ними.

У результаті сумісного використання Data Mining і Fuzzy-технологій істотно знижуються вимоги щодо якості початкових даних, а також їх кількості і достовірності. Це є істотним саме для науково-медичної практики, де, як правило:

- важко одержати потрібний об'єм первинної достовірної інформації;
- присутня переважно експертна вербальна (лінгвістична) якісна інформація;
- є різноманітна за складом і джерелами інформація (якісна і кількісна) від різних експертів;
- бракує або спотворена статистична інформація, або ця інформація не викликає довіри у користувача;
- необхідно враховувати нові істотні чинники, які можуть виникнути в майбутньому.

З урахуванням принципів доказової медицини [1] пропонується структурна схема автоматизованої системи моделювання, прогнозування і підтримки ухвалення рішень, заснованих на доказах (рис. 1).

Система включає наступні основні компоненти:

- підсистема моделювання і прогнозування;
- підсистема вироблення рекомендацій;
- база даних і знань (доказів).

Гарантії того, що тільки уповноважені користувачі можуть впливати на ухвалення та використання рішень, що приймаються, забезпечує вбудована в операційну систему політика безпеки з відповідними правами.

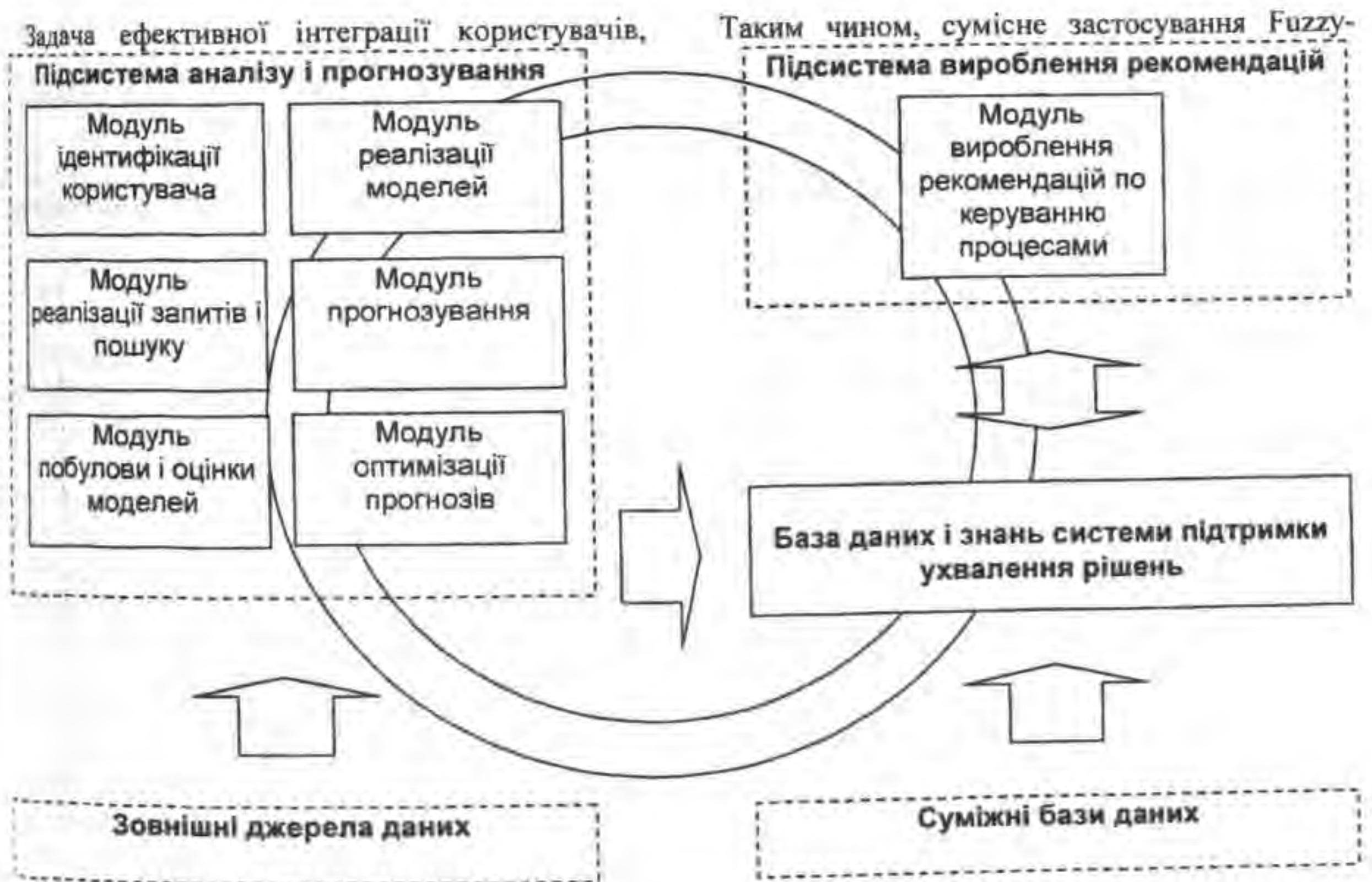


Рис.1 Структурна схема автоматизованої системи моделювання, прогнозування і підтримки прийняття рішень в доказовій медицині.

web-сервісів і первинних інформаційних систем вирішується завдяки використанню сервіс-орієнтованої архітектури (SOA) системи.

В разі побудови єдиного сховища даних (Data Warehouse) можливе створення рішення, яке може використовуватися в різних системах та додатках багато разів, хоча сама система може працювати лише на одному сервері. При цьому можлива також кластеризація серверів для забезпечення їхньої взаємодії в режимі реального часу в інтернет-середовищі.

Список використаних джерел

1. Чубенко А.В. Медицина, основанная на доказательствах, и современные информационные технологии / А.В.Чубенко, П.Н.Бабич, С.Н.Лапач // Український медичний часопис. – 2004. – №2. – С.49-56.
2. Методы и модели анализа данных: OLAP и Data Mining / А.А.Барсегян, М.С.Куприянов, В.В.Степаненко,

технологій разом із методами та засобами поглибленої аналітики (Data Mining) є перспективним з точки зору розв'язання проблеми автоматизації підтримки прийняття рішень в медицині, заснованій на доказах, у реальному часі.

И.И.Холод. – БХВ-Петербург, 2004. – 336 с.

3. Svolba G. Data Preparation for Analytics using SAS® / G. Svolba. – Cary, NC: SAS Institute, 2006. – 408 p.
4. Бочарников В.П. Fuzzy Technology: Основы моделирования и решения экспертно-аналитических задач / В.П.Бочарников, С.В.Свешников – К.: Эльга, Ника-Центр, 2003. – 296 с.

Надійшла до редколегії 13.12.2010

УДК 004.925.8

В.О. Твердохлебов, д.т.н., проф.

Геометричні моделі автоматних відображень та автоматів

Розглядається новий спосіб задання дискретних детермінованих автоматів. Для цього на автоматному відображенні (множині пар) визначаються лінійні порядки для перших та для других елементів пар. Суміщення лінійно впорядкованих множин перших та других елементів пар автоматного відображення набуває форми графіка. Символьні елементи пар замінюються їх номерами по лінійним порядкам. Результат такої заміни ізоморфно вкладається у числову систему координат на площині. Для символного автоматного відображення, представленого числовим графіком, використовується задання точками на геометричних кривих, інтерполяція методами інтерполяції часткових графіків, розпізнавання на основі розв'язання рівнянь та нерівностей для кривих та ін.

Ключові слова: автомат, геометрична крива, розпізнавання, інтерполяція.

*E-mail: tverdokhlebovva@list.ru

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

1. Automaton and its geometrical images.

A discrete deterministic automaton is a mathematical model of a set of discrete processes. Each of these processes in case of fixed initial state is defined by interaction of three processes: the process of feeding sequence of input signals into the input of automaton; the process of transforming automata's states and the process of giving out sequence of output states. In exact mathematical form (and in one of the variants of automata) automaton is formally represented by the 5-tuple $A=(S, X, Y, \delta, \lambda)$, where S, X and Y are sets of states, input and output signals, respectively; transition function δ and output function λ that are defined as $\delta: S \times X \rightarrow S$ and $\lambda: S \times X \rightarrow Y$. Automata's functioning processes are defined in abstract integral positive time $t \in \mathbb{N}^+$ by dynamic equations:

$$s(t+1) = \delta(s(t), x(t)), \quad y(t) = \lambda(s(t), x(t)).$$

If states and sequences of states are considered in time as auxiliary (or as nonmetering) variables, then

V. A. Tverdokhlebov, doctor of technical science, professor

Geometrical models of automaton mappings and automata.

In article considered a new way of definition of laws of functioning of discrete deterministic automata. For that on automaton mapping (set of pairs) define linear orders for first and second elements of pairs. Combination of linearly ordered set of first and second elements of pairs of automaton mapping take a shape of graphic. Symbolic elements of pair replace on their number by linear orders. Result of that replace isomorphically embed in numerical coordinate system on plane. For symbolic automaton mapping, which presented in form of numeric graphic, is used definition by points on geometrical curves, interpolation by method of interpolation of partially specific graphics, recognition on the basis of solving of equality and non inequality.

Key Words: automata, geometrical curve, recognition, interpolation.

the set of possible processes of initial automaton functioning (A, s_0) is represented by automaton reflection

$$\rho_{s_0} = \bigcup_{p \in X^*} \{(p, \lambda(s_0, p))\},$$

where

$$(\forall x \in X)(\forall p \in X^*) \delta(s_0, xp) = \delta(\delta(s_0, xp), p) \& \\ \& \lambda(s_0, xp) = \lambda(s_0, x) \lambda(\delta(s_0, x), p).$$

In what follows we will use the next equal form of automaton reflection:

$$\rho'_{s_0} = \bigcup_{p \in X^*} \{(p, \lambda'(s_0, p))\}, \text{ where}$$

$$(\forall x \in X)(\forall p \in X^*) \lambda'(s_0, px) = \lambda(\delta(s_0, p), x).$$

The set of pairs $(p, \lambda(s_0, p))$ and the set of pairs $(p, \lambda'(s_0, p))$ are converted into character graphics based on the introduction of a linear order ω_1 on X^* and ω_2 on Y^* (ω_2 on Y).

The rules below are used to determine the linear orders ω_1 and ω_2 .

The rules for a linear order ω_1 on X^* :

Rule 1: On the X set, we introduce some linear order (which is denoted as ω_1)

Rule 2: The order of ω_1 on X will be distributed to a linear order on X^* , assuming that:

- for any unequivalent words $p_1, p_2 \in X^*$ for which $|p_1| < |p_2|$, p_1 will be less than p_2 .

- for any unequivalent words $p_1, p_2 \in X^*$ for which $|p_1| = |p_2|$, their position will be defined according the order of their first unequal letters on X .

The rules for a linear order ω_2 on Y^* are defined by the same way as ω_1 on X^* . We introduce a standard representation of the transition function $\delta: S \times X^* \rightarrow S$, where $S = \bigcup_{p \in X^*} S_p$, $x \in X$, $p \in X^*$ using the rule

$\delta(S_p, x) = s_{px}$ (Pic. 1).

Let us introduce some linear order ω_1 into the set X^* and transform ρ_{s_0} into linear ordered set (ρ_{s_0}, ω_1) according to linear order w of the first elements of pairs of automaton reflection ρ_{s_0} . Obviously, if the set X and linear order w on X^* are known, then linear ordered automaton reflection (ρ_{s_0}, ω_1) is definitely defined by sequence of the second elements of their pairs. This consequence is designated as $Pr_2(\rho_{s_0}, \omega_1)$.

The linear order ω_2 is defined on set Y and coordinate system with abscissa axis (X^*, ω_1) and ordinate axis (Y, ω_2) is considered. Automaton mapping ρ_{s_0} as a set of points, which coordinates are linearly ordered by linear orders ω_1 on X^* and ω_2

on Y , is imagined in this coordinate system. The corresponding graph $G = (\rho_{s_0}, \omega_1, \omega_2)$ with symbolic points' coordinates and the group of three $(\rho_{s_0}, \omega_1, \omega_2)$ have a one-one correspondence. The consequence $Pr_2(\rho_{s_0}, \omega_1)$ and the linear order ω_2 on Y also have one-one correspondence with the graph $G = (\rho_{s_0}, \omega_1, \omega_2)$ of automaton mapping ρ_{s_0} (example 2). At greater length see [1-6].

The symbolic graph $G = (\rho_{s_0}, \omega_1, \omega_2)$ is converted in a graph G_N with points that have integer positive coordinates, which are a numbers of symbolic coordinates of points by linear orders ω_1 и ω_2 . The graph G_N can be isomorphically embedded in the first quadrant of a rectangle coordinate system on a plane and systematize interdisposition of the graph's points by putting them on geometrical curved lines and geometrical figures. At first, it lets define discrete deterministic automaton with finite and enumerably infinite sets of states by new tools and research attributes of principles of automaton functioning on a basis of comparison of automaton attributes and geometrical structures. Turn of research of principles of automaton functioning (functions δ and λ in a symbolic form) to a numerical structures (numerical automaton mappings that are put on numerical structures) lets use powerful mathematical idealizations: actual infinite, infinitesimal, continuity, limiting process, summing series, etc.

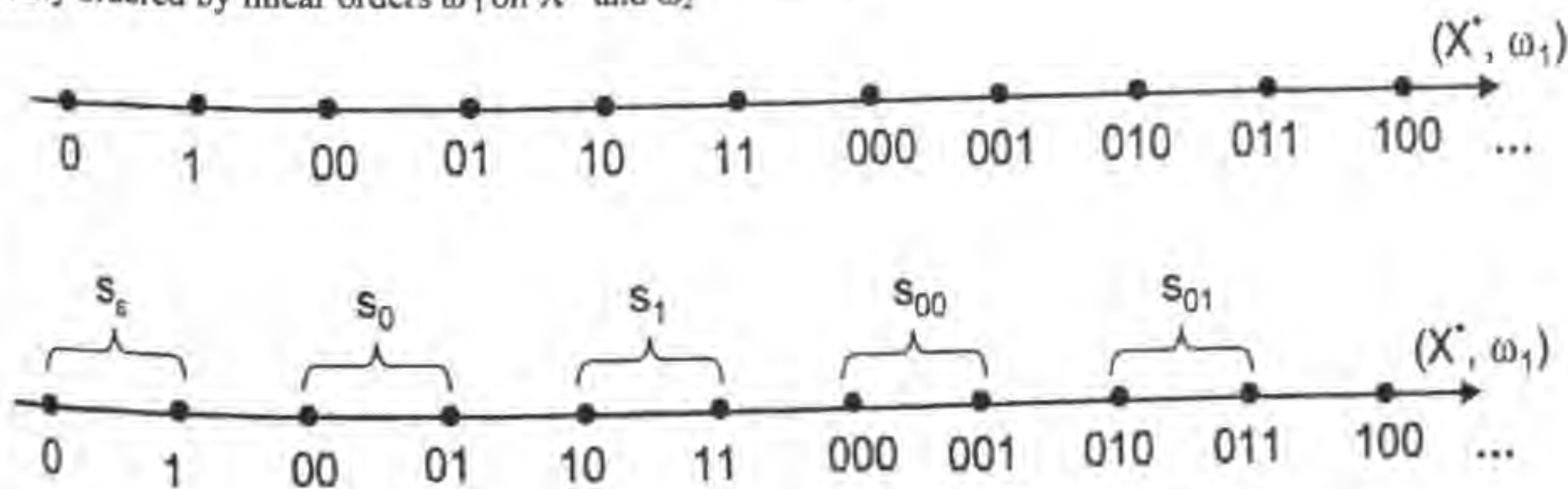


Fig.1. Allocation of sequences of input signals on the horizontal axis (X^*, ω_1) , for $X = \{0, 1\}$ and the standard notation of interrelation between X -axis and the automaton states.

The transformation of character graphics $H_1 = (\rho_{s_0}, (X^*, \omega_1), (Y, \omega_2))$ and $H_2 = (\rho_{s_0}, (X^*, \omega_1), (Y^*, \omega_2))$

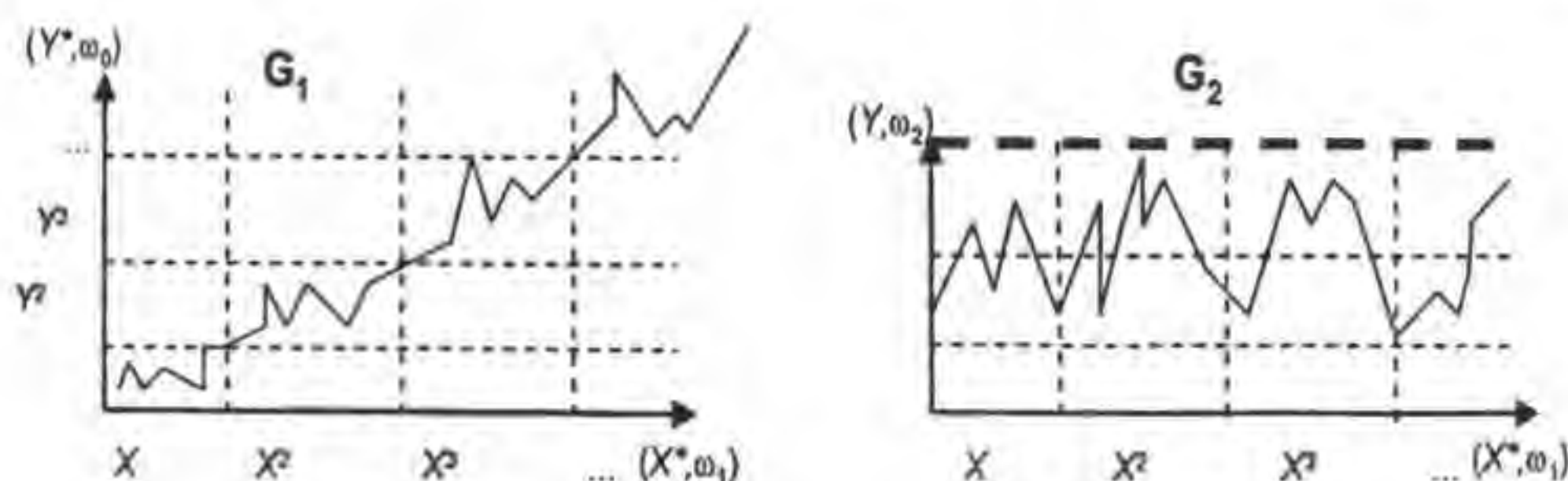


Fig.2. Variants of symbolic graphs of automata mapping.

to their digital analogs is done according to the next rules:

1. Each element $p \in X^*$ is replaced by its number $r_1(p)$ according to the linear order ω_1 , that has its first element according to the construction.
2. Each element $y \in Y$ ($q \in Y^*$) is replaced by its number $r_2(y)$ ($r_2(q)$) according to the linear order ω_2 .
3. The numerical structures that were received after the implementation of rules 1 and 2 are isomorphically embedded into the main quadrant (rectangle) of the Cartesian coordinate system.

So, the discrete positive integer graphics G_1 and G_2 were obtained.

There are some variants of matching the $G_{s_0}^A = G_1$ graphic with the numeric geometrical structure (geometric curve). So these variants are:

- 1) by the combination of graphic-points with some points of geometric lines, given by the equation $y = f(x)$ (with the proof of the possibility of such combination);
- 2) by the constructing a curve using interpolation and extrapolation methods based on using the finite set of

points of the graph (with the proof of the possibility of such combination).

2. Example of interpolation partially specific lows of functioning of automaton.

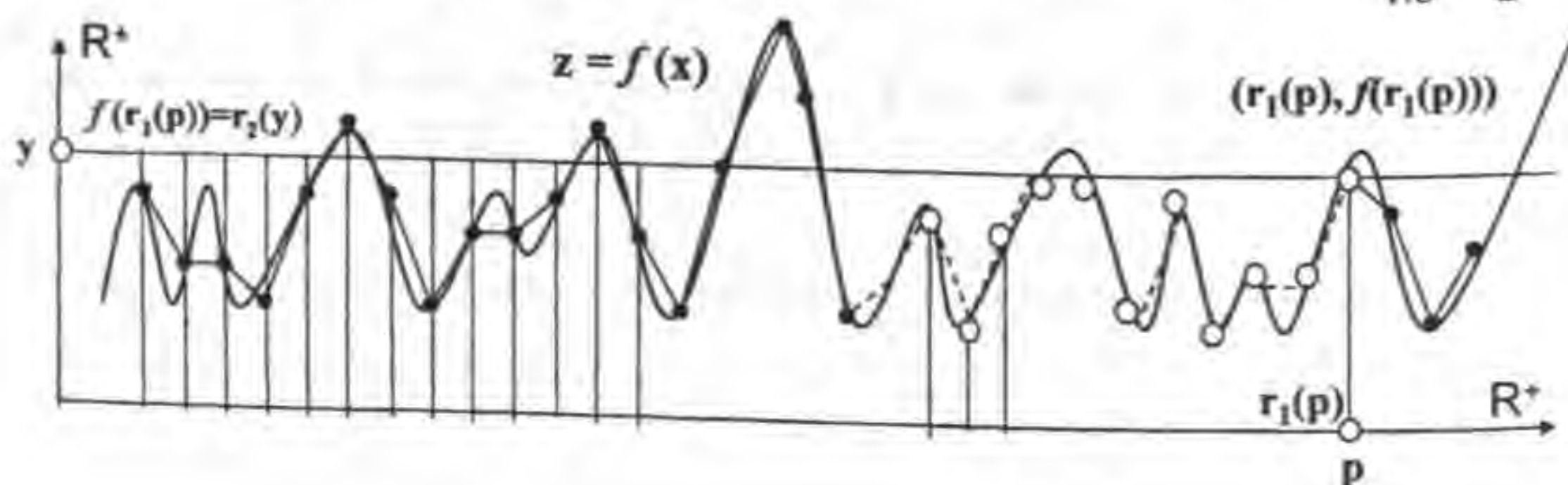
Let geometrical image of initial automaton only partially set. Specified points supposed as points of interpolation. By points of interpolation by selected method of interpolation constructed geometrical curve $z = f(x)$. Accepted hypothesis is that missing

Fig.3. Location on geometrical curve $z = f(x)$ points of interpolation $\bullet$ and missing points $\circ$ of the geometrical image of automaton.

point also located on this geometrical curve $z = f(x)$ (fig. 3). Numerical coordinates of points define input word and output signal.

3. Example of recognition of automata by their geometrical image.

Let geometrical images of initials automata (A_1, s_{01}) , (A_2, s_{02}) , (A_3, s_{03}) locate on analytically define curves $y^2 = 4x$ (parabola), $x^3 + y^3 = 4.5xy$ (folium of Descartes) and $\frac{x^2}{1.8^2} + \frac{y^2}{2^2} = 1$ (ellipse)



(fig.4). Solutions of equalities and inequalities between equation, which determined curves, define intervals on axis of abscises: ineffective , partially effective and effective for recognition automaton in set of automatons $\{(A_1, s_{01}), (A_2, s_{02}), (A_3, s_{03})\}$.

4. The universal solution of a problem of recognition of the automaton in the finite automatons families.

In the articles of Moore[7], Gill[8] were presented the basic provisions, models and methods, which were the base for constructing the methods of testing of the finite deterministic automaton.

$$a = (\bigcup_{i=1} S_i, X, Y, \bigcup_{i=1} \delta_i, \bigcup_{i=1} \lambda_i).$$

Problems of recognition of the automat in the given family of automat are interpreted as a problem of technical diagnosing.

A set of a technical malfunctions of the object represented by an automat family $\alpha = \{A_i\}_{i \in I}$, where $A_i = (S_i, X_i, Y_i, \delta_i, \lambda_i)$, $S_i \cap S_j = \emptyset$ and $i \neq j$. A model of a operable object is an automat

$$A_0 = (S_0, X, Y, \delta_0, \lambda_0), \text{ where } S_0 \cap S_i = \emptyset, i \in I.$$

In the control problem is required to determine what is correct: $z = 0$ or $z \in I$ for the test object with an unknown model $A_z = (S_z, X_z, Y_z, \delta_z, \lambda_z)$.

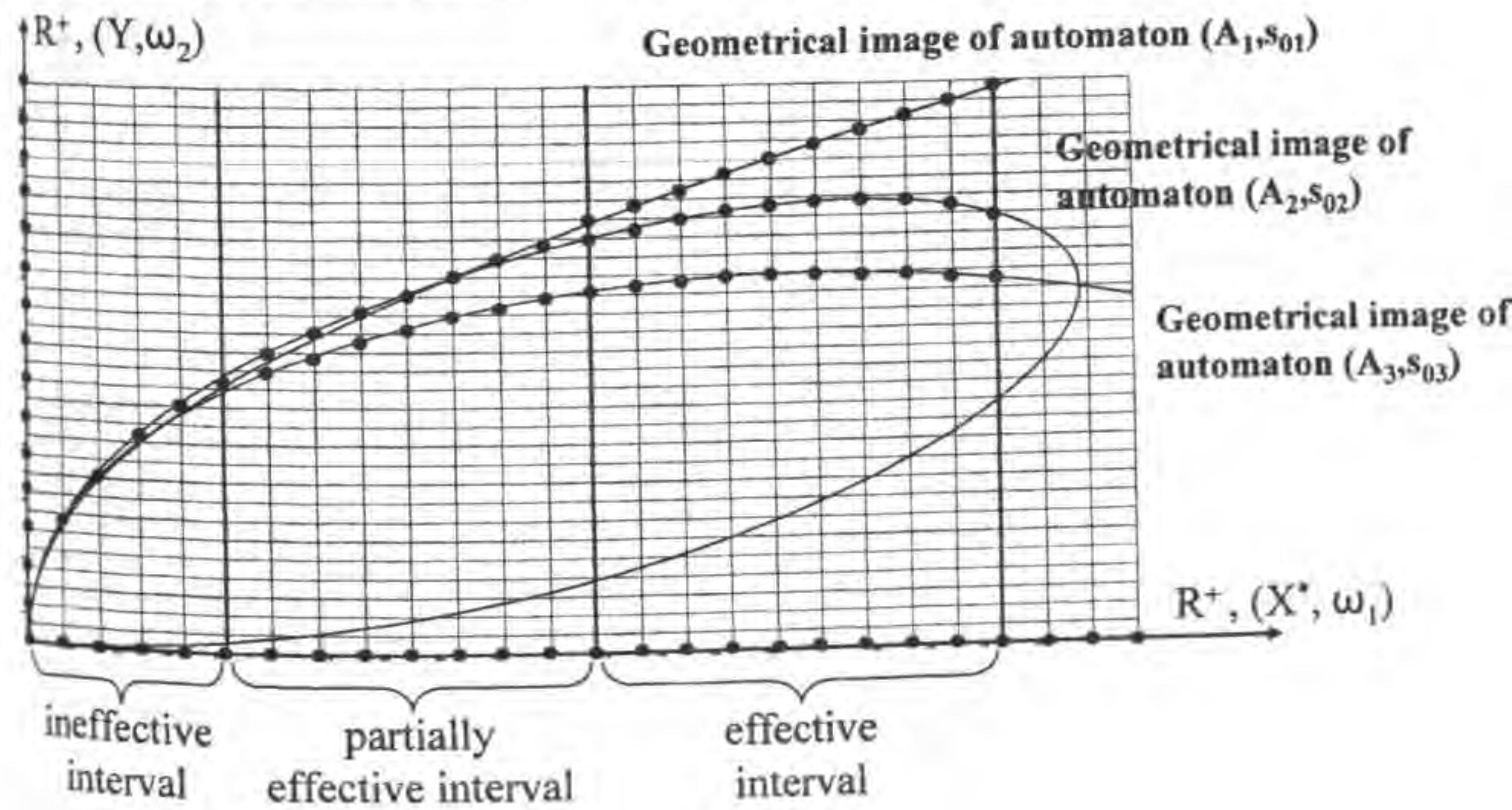


Fig.4. Ineffective , partially effective and effective intervals for recognition automaton in set of automatons $\{(A_1, s_{01}), (A_2, s_{02}), (A_3, s_{03})\}$.

Article[7] contains a statement for an automat adjustment problem, criteria of the solution existence for these problem(automat must be already minimized), the method of solving the problem (construction of a tree of functioning variants) and estimation the length of the input sequence for the p-solution ($p \in X^*, |p| < 0.5 * (|S| * (|S|-1))$), where X – is the set of input signals and S – is the set of the automat states.

The automat adjustment problem in the final comparable family of automat $\alpha = \{A_i\}_{i \in I}$, where $A_i = (S_i, X_i, Y_i, \delta_i, \lambda_i)$, $S_i \cap S_j = \emptyset$ and $i \neq j$, presented in the article[8] was reduced to the adjustment problem for a fissionable automat

In the diagnosis problem is required to determine for the what element $i_0 \in I$, the statement $z = i_0$ is correct.

The Moore-Gill method gives us only a specific solution for a problems of technical diagnostics for a specific families of automat $\alpha = \{A_i\}_{i \in I}$ and $A_0 = (S_0, X, Y, \delta_0, \lambda_0)$ automat.

In the articles [9-10] was found an analytic universal solution for a problems of technical diagnostics for a finite deterministic systems with memory.

The efficiency of the implementation of the universal testing significantly limited due to the large size of the sequence of input signals received as a universal solution.

During the analysis of the set of families of automats in the Gill-Moore method:

$\alpha_1 = \{A_{i_1}\}_{i_1 \in I_1}, \alpha_2 = \{A_{i_2}\}_{i_2 \in I_2}, \dots, \alpha_k = \{A_{i_k}\}_{i_k \in I_k}$, each family is treated separately and the resulting set of tests $p_1, p_2, \dots, p_k$ contains mutually independent tests. If the study set of malfunctions I_j , the test will be constructed from the start.

The test, which is suitable for a fairly large set of families, limited only by common sets of inputs signals X and output signals Y , as well as an overall assessment of the number of states (no more than m states) will be called universal test.

The existence of universal tests for the full set of families of comparable machines, provided that there is no any two equivalent states $s \in S_i$ and $s \in S_j$, and $i \neq j$, was presented in 1980-1982 in the articles [4,5]. In these articles and articles [6,7] was represented a method for constructing the universal test and was reduced the estimation of the test length.

The main idea of the possibility of a universal testing of an automat, constructing method and the estimation for a test length, determined by the following theorems:

the solution of automat recognition problem exists (using simple unconditional experiment) for any final family of a finite deterministic automats, which are an exceptional class of automats and there is the estimation of a length for a sequence of input signals p

$$|p| \leq 0.5 * (|S_{\max}| * (|S_{\max}| - 1))$$

Theorem 1. (Moore [7]) The adjustment problem (finding the final state of an automat) for automat $A = (S, X, Y, \delta, \lambda)$ without equivalent states has the solution $p \in X^*$, where

$$|p| \leq 0.5 * (|S_{\max}| * (|S_{\max}| - 1)).$$

Theorem 2. (Gill [8]) The adjustment problem in the final family of automats $\alpha = \{A_i\}_{i \in I}$, where $A_i = (S_i, X_i, Y_i, \delta_i, \lambda_i)$, $S_i \cap S_j = \emptyset$ if $i \neq j$, reduced to the adjustment problem for automat

$$a = (\bigcup_{i=1} S_i, X, Y, \bigcup_{i=1} \delta_i, \bigcup_{i=1} \lambda_i).$$

Theorem 3. (Tverdohlebov [9-10]) If $p \in X^*$ is the solution for an adjustment problem for the automat $A = (S, X, Y, \delta, \lambda)$, then for any $p_1, p_2 \in X^*$, sequence $p_1 p p_2$ will be the solution of the adjustment problem too. The $u \in X^*$ sequence of the input signals, which contains as subsequences all sequences with the length of $0.5 * n * (n-1)$, is the universal test for the all automaton families (as the

exceptional classes), with X as the set of the input signals and with the number of states no more than n .

For given X and n the length of the universal test will be no more than

$$|X|^{n(2n-1)} + n(2n-1) - 1.$$

Class (2, 2, 2) - automats contains 256 automats, forming machines 32,640 2-element families. According to the Moore's theorem, for each automat family as the exceptional class there is a solution of the adjustment problem the length of 6. So the universal test for all automaton families as the exceptional classes in the (2,2,2) - automats has the length of 69. For example sequence $u = 1^6 0^6 10^4 1^2 0^3 101 0^3 1^3 0^2 10^2 101^2 0^2 1^2 010^2 1^4 0^3 1^5 01^2 01^5$.

Two values are used in assessing the length of the universal test: the number of states of fissionable automat and the number of input signals. In the article [11].

5. Interpretation of arbitrary sequence as representation of definition of lows of functioning of automaton.

Arbitrary sequence $\xi = y_{j_1} y_{j_2} \dots y_{j_k} \dots$ of elements can be interpreted as representation automaton with finite or infinite set of states. For that sequence divide on parts with length m , where m - estimated number of input signals of automaton. Sequence ξ located on points of tree of functioning of automaton (fig.5) from up to down and from left to right. Tree arc are marked by input signals. In result we have representation of definition of lows of functioning of automaton by tree or in form of geometrical image (fig.6).

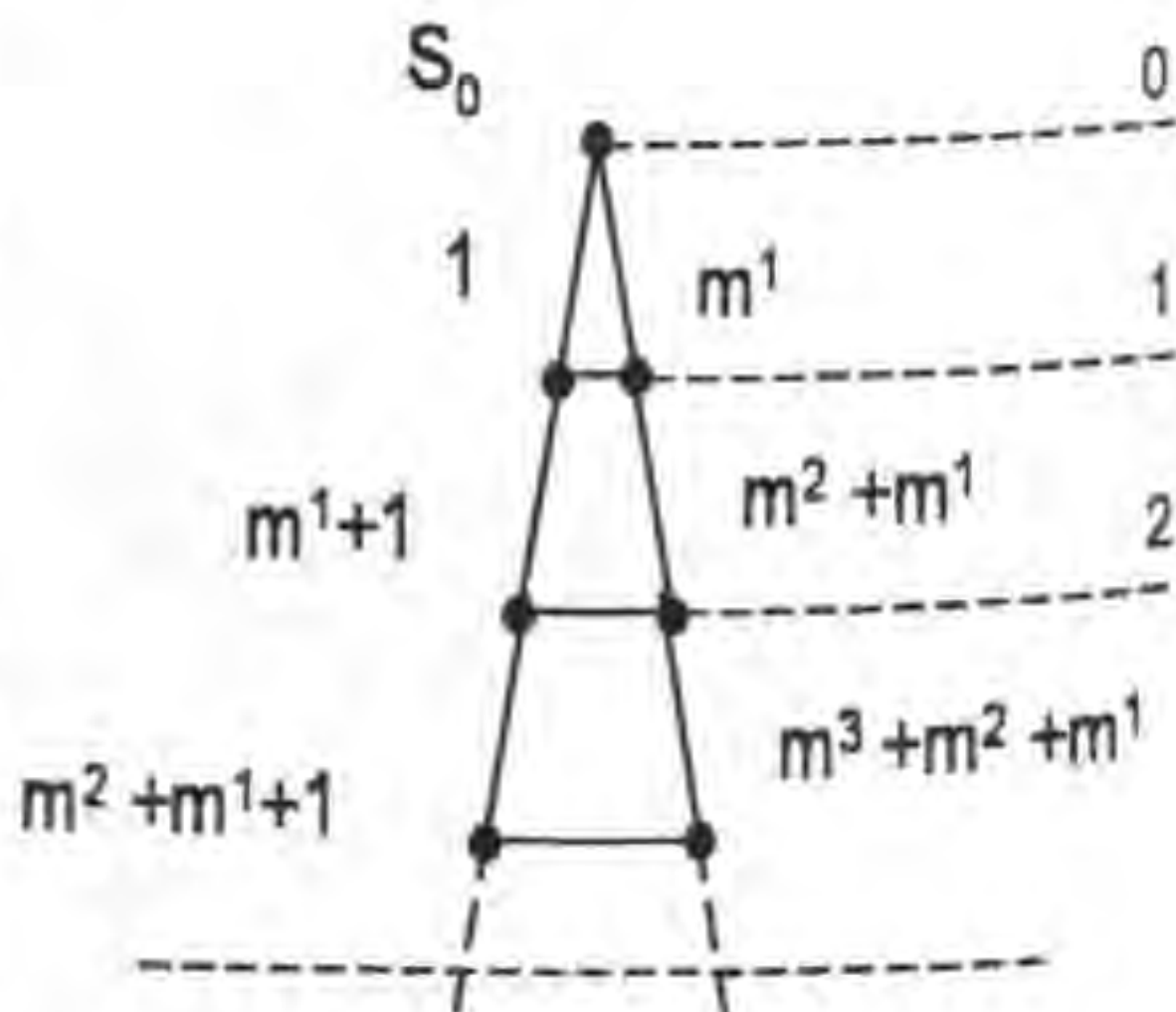


Fig.5. Location of numbers of states of automaton on tree of functioning of automaton.

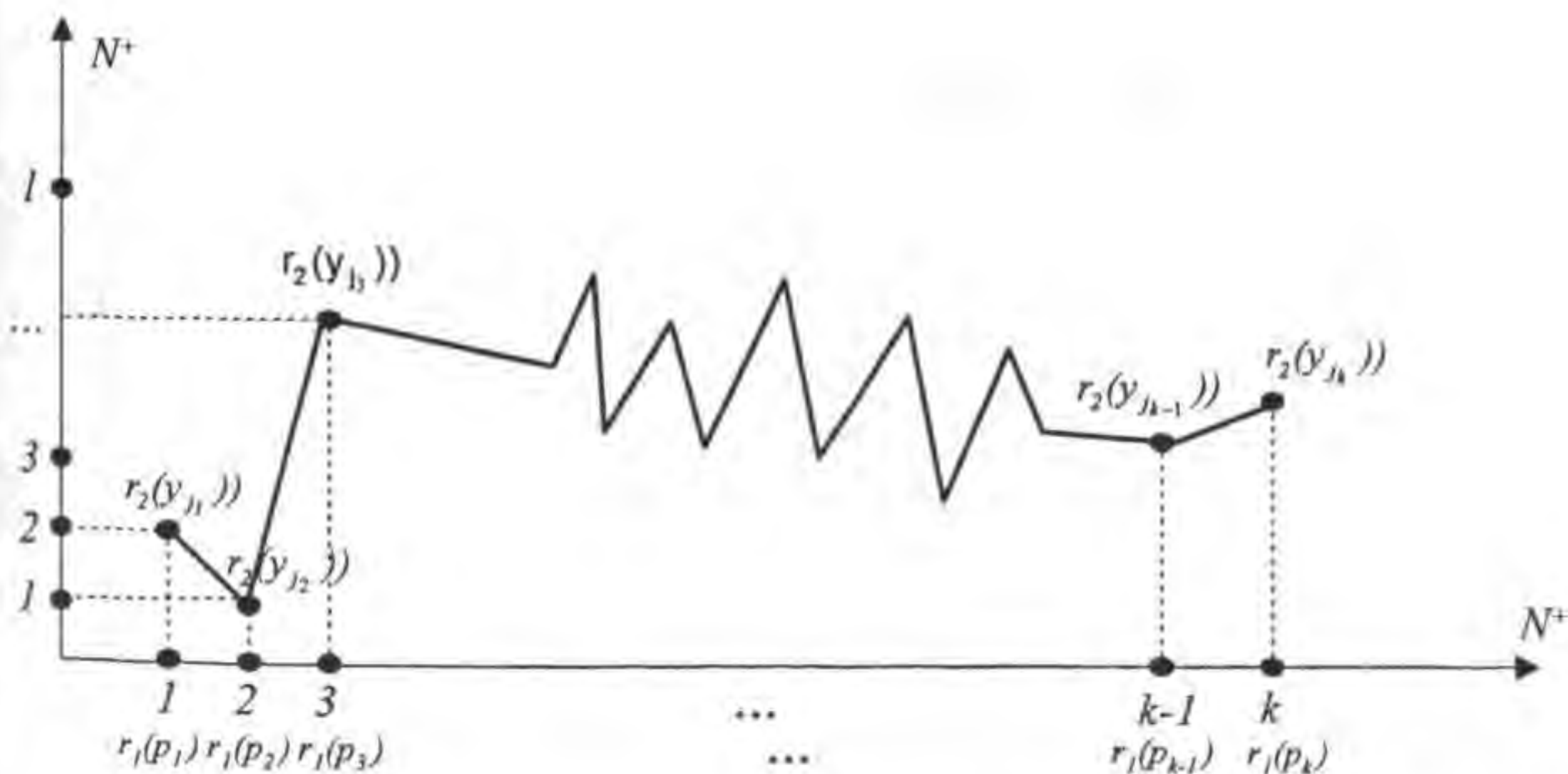


Fig.6. Location of arbitrary sequence $\xi = y_{j1} y_{j2} \dots y_{jk}$ on geometrical image lows of functioning of automaton interpretable as sequence of second coordinates of points of geometrical image.

References.

1. Твердохлебов В. А. Геометрические модели и методы в техническом диагностировании // Информ.-управляющие системы на ж.-д. транспорте. - 1996, - № 3/4, - С. 58. - В номере отражены материалы междунар. науч.-практ. конф. "Перспективные системы управления на железнодорожном, промышленном и городском транспорте", Украина, Алушта, 11-16 сент. 1996г.
2. The general features of geometrical images of finite state machines / V. A. Tverdokhlebov // Proceedings of East-West Design & Test Workshop (EWDTW'04), Ylta, Alushta, Crimea, Ukraine, Septembre 23-26, 2004, - Kharkov : National University of Radioelectronics, 2004. - P. 243-246
3. Рекуррентность геометрических образов / В. А. Твердохлебов // Інформаційно-керуючі системи на залізничному транспорті. - 2004. - № 4/5 (48/49). - С. 88-90.
4. - Твердохлебов В.А. Геометрические образы конечных детерминированных автоматов // Изв. Сарат. ун-та. Новая серия. Сер. Математика, Механика, Информатика. - 2005. - Т. 5, вып. 1. - С. 141-153 : рис. - Библиогр.: с. 152-153 (29 назв.).
5. Геометрические образы поведения дискретных детерминированных систем / В. А. Твердохлебов // Радіоелектронні комп'ютерні системи. - 2006. - № 5 (17). - С. 161-165. - Библиогр.: с. 165 (14 назв.).
6. Твердохлебов В.А. Геометрические образы законов функционирования автоматов. - Саратов: Изд-во «Научная книга», 2008. - 183 с. ISBN 978-5-9758-0924-7
7. Moore, E. Speculative experiments with sequential machines, "Automata, IL, 1956.
8. Gill, Introduction to the theory of finite automata. «Nauka», Moscow, 1966.
9. Tverdokhlebov V.A. Universal Generators tests and diagnosing system / / Technical diagnosis: [Intercollege. Sat] / - Rostov on Don: [Growth, Ing.-constructs. Inst], 1982. - pp. 106-109.
10. Tverdokhlebov V.A. Methods of constructing universal tests for finite automata, / Automation and Remote Control. - 2005. - № 1. - pp. 154-163.
11. B. Lippel and I.J. Epstein, A Method for Obtaining Complete Digital Coding Chains, IRE Trans. EC-6 p. 121, 1957

УДК 004.4

Ткачук І.Ю., асистент

Елементи формалізації Бі-методу

У статті пропонується формалізація одного з ключових інструментів побудови специфікацій в рамках формального методу Бі, так званих узагальнених підстановок (generalized substitutions). Формалізація проводиться на основі принципів композиційно-номінативного підходу, запропонованого В. Н. Редько та М. С. Нікітченко

Ключові слова: Бі-метод, узагальнена підстановка.

*E-mail: tkachuk@unicyb.kiev.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

На сьогоднішній день широкий спектр задач розв'язується в автоматичному режимі, використовуючи відповідне програмне забезпечення. Але проблема полягає в тому, що програмне забезпечення не завжди є цілком відповідним до задач, для яких воно було розроблено, тобто в програмах досить часто зустрічаються помилки. Для зменшення числа таких помилок в сучасній комп'ютерній індустрії поширено використання формальних методів розробки програм.

В статті пропонується формалізація підстановок Бі-методу на основі принципів композиційно-номінативного підходу [1, 2].

Проста (елементарна) підстановка, яка подається в [1] як синтаксична операція заміни імені змінної на її значення, може бути формалізована за допомогою використання операції накладки як

$$ES^{[1]}(p, g)(d) = p(d \nabla [x \mapsto g(d)]).$$

Беручи до уваги ключову роль, яка відводиться цій підстановці в [1] при заданні значення інших підстановок, можна дати таку їх формалізацію:

1) Підстановка порожньої дії формалізується як $Skip(p)(d) = p(id(d))$, де p – предикат, id – тотожна функція, d – множина змінних програми, кожна змінна програми задається у вигляді $v \mapsto e$, де v – ім'я змінної, e – значення цієї змінної;

2) Багаторазова проста підстановка формалізується як $MSS^{[1, \dots, n]}(p, f_1, \dots, f_n)(d) =$

I. Y. Tkachuk, assistant professor

Elements of B-method formalization

Formalization of one important instrument of specification construction with B-Method is demonstrated, namely formalization of generalized substitutions is performed on the basis of composition-nominative approach, originated by Redko V. N., Nikitchenko M. S.

Key Words: B-method, generalized substitution.

$= p(d \nabla [v_1 \mapsto f_1(d), \dots, v_l \mapsto f_l(d)])$, де p – предикат, d – множина змінних, $v_1, \dots, v_l$ – імена змінних, $f_i, i \in \overline{1, l}$ – функції над змінними програми і множиною значень, що є множиною значень змінної з ім'ям v_i ;

3) Підстановка з передумовою (preconditioned substitution) формалізується як $PS(f, p)(d) = \begin{cases} f(d), & \text{якщо } p(d) = \text{true}; \\ \text{undef}, & \text{якщо } p(d) = \text{false}; \end{cases}$

4) Застережена підстановка (guarded substitution) формалізується як $GS(f, p)(d) = \begin{cases} f(d), & \text{якщо } p(d) = \text{true}; \\ id(d), & \text{якщо } p(d) = \text{false}; \end{cases}$

5) Скорочена умовна підстановка формалізується як

$$SCS(f, p)(d) = \begin{cases} f(d), & \text{якщо } p(d) = \text{true}; \\ id(d), & \text{якщо } p(d) = \text{false}; \end{cases}$$

6) Умовна підстановка формалізується як $CS(f, g, p)(d) = \begin{cases} f(d), & \text{якщо } p(d) = \text{true}; \\ g(d), & \text{якщо } p(d) = \text{false}; \end{cases}$

7) Підстановка обмеженого вибору формалізується як $BCS(p, f, g)(d) = p(f(d)) \wedge p(g(d))$;

8) Підстановка необмеженого вибору формалізується як

$$UCS^{[1]}(p, f, E)(d) = \bigwedge_{e \in E} p(f(d \nabla [z \mapsto e])), \text{ де}$$

E – множина значень, яких може набути змінна x в результаті виконання підстановки;

9) Підстановка довільного елемента з множини формалізується як

$$CIS^{[x]}(p, E)(d) = \bigwedge_{e \in U} p(d \nabla [x \mapsto e]), \text{ де } x - \text{ім'я}$$

змінної, по якій відбувається підстановка, E – множина можливих значень змінної x , $U = \{e \mid e \in E \wedge p(d \nabla [x \mapsto e]) = \text{true}\}$, $U \subseteq E$, p – предикат вигляду $p: S \rightarrow \text{Bool}$, де E – множина можливих для проведення підстановки значень змінної x ;

10) Підстановка значення, що задовільняє предикат формалізується як

$$CBP^{[x]}(r, p)(d) = \bigwedge_{e \in E} r(d \nabla [x \mapsto e]), \text{ де } E -$$

множина, можливих для проведенні підстановки значень змінної x ;

11) Послідовна підстановка формалізується як $SS(p, f, g)(d) = p(f(g(d)))$;

12) Підстановка циклу формалізується як $L(f)(d) = \{d, L(f)(f(d)), L(f)(f(f(d))), \dots\}$, де

Список використаних джерел

1. Abrial J.-R. The B-book: assigning programs to meanings / J.-R. Abrial. – New York: Cambridge University Press, 1996. – 779 p.
2. Никитченко Н.С. Композиционно-номинативный подход к уточнению

$d \in \text{State}$, L – формалізація підстановки циклу T^* у вигляді багатозначної функції $d \xrightarrow{L} d^n$, де n – кількість ітерацій застосування функції f

до даних d при використанні підстановки L , $n \in N \cup \{\infty\}$; у випадку коли $n = \infty$ значення $L(f)(d)$ вважається невизначеним, f – функція вигляду $\text{State} \xrightarrow{f} \text{State}$ (формалізація підстановки T з визначення);

13) Підстановка циклу **WHILE** формалізується як

$$WS(p, f)(d) = \begin{cases} WS(p, f)(f(d)), & p(d) = \text{true}; \\ d, & \text{інакше}; \end{cases}$$

де p – предикат вигляду $p: S \rightarrow \text{Bool}$, f – функція вигляду $\text{State} \xrightarrow{f} \text{State}$.

понятия программы / Н.С. Никитченко. – Проблемы программирования, 1999, №1, с.16–31.

3. Редько В.Н. Композиционная структура программологии / В.Н. Редько. – Кибернетика и системный анализ, 1998, № 4, с. 47–66.

Надійшла до редколегії 12.12.10

УДК 004:052:42

Туркін І.Б., д.т.н., проф., Соколова* Є.В.,
Юр'єва*** О.І.

Афінний інтервальний метод автоматичного контролю достовірності телеметричної інформації

Розглядається метод афінно-інтервального аналізу для обробки телеметричної інформації, що надходить у процесі функціонування космічного апарату. Показано, що використання даного методу дозволяє забезпечити достовірність результатів обробки телеметричних даних і дає можливість їх використання в якості вихідної інформації для автоматизованої системи підтримки прийняття рішень.

Ключові слова: афінно-інтервальний аналіз, космічний апарат, обробка телеметричної інформації, інформаційно-вимірювальні системи.

I.B. Turkin*, Sc. D., Professor, Y.V. Sokolova**,
E.I. Yuryeva***.

Affine interval method for telemetry information accuracy automatic control

A method of affine interval analysis for processing telemetry information coming in the operation of spacecraft. There was shown that using this method ensures the accuracy of results of processing telemetry information, and allow their use as background information for automated decision support system.

Key Words: Affine interval analysis, spacecraft, telemetry data processing, an information-measurement system.

*E-mail: energy@d4.khai.edu, nikrutrogi@mail.ru

**E-mail: EVSkhai@gmail.com, Sev_ai@mail.ru

***E-mail: yuryevaelena@gmail.com

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Введення

У процесі функціонування космічного апарату (КА) контролюються різні з фізичної сутності й динаміці процеси, що відбуваються як на самому об'єкті, так і поза ним. Всі контрольовані системи КА пов'язані між собою і представлені сукупністю різнорідних параметрів. Дистанційний контроль здійснюється в режимах безпосередньої передачі телеметричних параметрів в реальному масштабі часу (при проведенні сеансів зв'язку та знаходженні КА в зоні видимості наземних засобів) або запису телеметричної інформації поза зонами видимості з наступним відтворенням інформації. Складність інтерпретації результатів аналізу вимірювальної інформації зростає через можливі стани системи, які викликані несправностями або зовнішніми обурюючими факторами [1].

1. Мета і завдання дослідження

Міжнародне співробітництво у космосі вимагає від учасників великих міжнародних космічних проектів проведення великомасштабної інтеграції космічних засобів, включаючи системи телеметрії і управління. Основою і необхідною умовою для цього є уніфікація і міжнародна стандартизація інформаційного обміну з КА. Така стандартизація широко проводиться в рамках робіт Консультативного комітету з систем пере-

дачі космічних даних (CCSDS), починаючи з 1990-х років, для організації надійних засобів обміну інформацією.

Концепція пакетної телеметрії CCSDS [2] забезпечує три різних категорії якості передачі даних космічним каналом зв'язку за рахунок поєднання процедур виявлення і корекції помилок, повторної передачі даних. Як показує практика обробки ТМІ льотних випробувань, результати первинної обробки вихідних параметрів містять як поодинокі, так і згруповані аномальні похибки. Використання таких результатів в якості вихідної вимірювальної інформації для автоматизованої системи підтримки прийняття рішень у реальному масштабі часу призводить до помилкових висновків про контрольовані події [3].

Сьогодні в метрології інформаційно-вимірювальних систем (ІВС) для оцінювання їх точності застосовуються методи, засновані на положеннях теорії автоматичного регулювання, статистики і т.д. Бортові інформаційно-вимірювальні системи сьогодні - це спеціалізовані обчислювачі, засоби сполучення з об'єктом контролю і управління, обчислювальна мережа, точність і якість яких залежить значною мірою від зовнішніх по відношенню до ІВС сигналів і перешкод, до яких у процесі обробки в каналах системи додаються внутрішні похибки і збої [4].

Розподілена архітектура бортових систем збору інформації вимагає врахування апертурних похибок, тобто похибок обумовлених несинхронністю вимірювань різних параметрів, виконаних автономними вимірниками [5].

Реалізація складних програм обробки ТМІ вимагає спеціальних методів контролю точності і надійності програмної реалізації на основі розрахункової оцінки точності обчислення всіх операторів і використовуваних даних, що складають цю програму [6]. Метрологічна атестація, як «дослідження точносних властивостей алгоритму в рамках конкретної вимірювальної задачі або методики виконання вимірювань з метою оцінювання характеристик складових похибок одержуваних результатів вимірювань» [7], проводиться для найгірших (з точки зору шуканих характеристик алгоритму або програми) варіантів поєднань моделей корисних сигналів і моделей похибок вихідних даних. Використання найбільш несприятливих варіантів дає гарантовану, але завищену оцінку підсумкової похибки, тому створення алгоритмів, самостійно контролюючих в процесі обчислень точність результату є актуальною задачею.

Можливість виконання доказових (достовірних, надійних) обчислень на ЕОМ з гарантованою точністю, коли невизначеність і неоднозначність інформації виникає з моменту її отримання, є однією з переваг методів інтервального аналізу [8]. В цьому випадку результат кожного виміру можна розглядати як інтервальну величину з центром, що відповідає свідченням датчика, і шириною інтервалу, що визначається класом точності датчика [9]. Інтервальна модель даних є основою методів обробки інформації, що забезпечують контроль достовірності інформації. Під достовірністю розумітимемо відповідність інформації про об'єкти предметної області реальної дійсності. Забезпечення достовірності - це підтвердження на основі представлення об'єктивних доказів можливості використання результатів обробки ТМІ.

Метою роботи є обґрунтування можливості і доцільності застосування в завданнях обробки ТМІ методів афінно-інтервального аналізу.

2. Результати досліджень

Визначення. Об'єкт даних (ОД) - це формалізоване представлення вимірної або вчисленої величини з урахуванням погрешностей виміру і перетворень, а також часу, коли ця величина отримана:

$$Data_i = \langle V_i, T, k \rangle,$$

де $V_i = [V_i, \bar{V}_i] = \{V_i \in R | V_i \leq V_i \leq \bar{V}_i\}$ - інтервальне значення i -того об'єкту даних в деякий інтервал часу $T = \{t \in R | t \leq t \leq \bar{t}, t > 0\}$ від моменту формування запиту до моменту отримання відповіді, $k \in R, k \geq 0$ - динамічна характеристика для опису максимальної швидкості зміни об'єкту даних.

У розгляд вводиться інтервал часу, оскільки в результаті пакетного, найбільш часто вживаного запиту на введення інформації повертається пакет даних, що супроводжується відміткою часу усього пакету, а не окремих даних, в нього що входять.

2.1. Правила виконання операцій над ОД

1. Результатом арифметичної операції $\otimes \in \{+, -, \cdot, /\}$ над ОД А і В із співпадаючими інтервалами часу: $T_A = T_B = T_0$ буде також ОД:

$$C = A \otimes B = \langle \left[\min(V_A \otimes V_B, \bar{V}_A \otimes \bar{V}_B, V_A \otimes \bar{V}_B, \bar{V}_A \otimes V_B), \max(V_A \otimes V_B, \bar{V}_A \otimes \bar{V}_B, V_A \otimes \bar{V}_B, \bar{V}_A \otimes V_B) \right], T_0, k_C \rangle$$

$$k_C = \begin{cases} k_A + k_B, & \otimes \in \{+, -\}, \\ k_A \bar{V}_B + k_B \bar{V}_A, & \otimes = \cdot, \\ \frac{k_A \bar{V}_B + k_B \bar{V}_A}{\min(V_B, \bar{V}_B)^2}, & \otimes = / \end{cases} \quad (2)$$

Результат операції ділення, при $V_B \leq 0 \leq \bar{V}_B$ невизначено.

Для автоматичної оцінки впливу несинхронного отримання даних на достовірність результату операції в модель введено три операції приведення до загального часового інтервалу, або моменту часу, які дозволяють розповсюдити арифметичні операції на випадок довільних часових інтервалів усіх ОД, що беруть участь в обчисленні якого-небудь розрахункового параметра.

2. Приведення ОД до моменту часу τ . Результатом приведення ОД $A = \langle V_A, t, k_A \rangle$ до моменту часу τ називатимемо ОД $A_\tau = \langle V_{A_\tau}, \tau, k_A \rangle$, де:

$$\begin{aligned} \underline{\tau} &= \tau = \bar{\tau}, \\ \underline{V}_{A_\tau} &= \underline{V}_A + k_A \cdot \max(\tau - \underline{t}, \bar{t} - \tau) \\ \bar{V}_{A_\tau} &= \bar{V}_A - k_A \cdot \max(\tau - \underline{t}, \bar{t} - \tau). \end{aligned}$$

3. Приведення ОД до розширеного часового інтервалу. Під розширенням часовим інтервалом

для ОД $A = \langle V_A, t, k_A \rangle$ розумітимемо деякий інтервал часу t_e такий, що $\underline{t}_e \leq \underline{t}, \bar{t} \leq \bar{t}_e$. Тоді результатом приведення ОД $A = \langle V_A, t, k_A \rangle$ до часового інтервалу t_e називатимемо ОД $A_e = \langle V_{A_e}, t_e, k_A \rangle$, де

$$\begin{aligned} \overline{V_{A_e}} &= \overline{V_A} + k_A \cdot \max(\bar{t}_e - \underline{t}, \bar{t} - \underline{t}_e), \\ \underline{V_{A_e}} &= \underline{V_A} - k_A \cdot \max(\underline{t}_e - \underline{t}, \bar{t} - \underline{t}_e). \end{aligned}$$

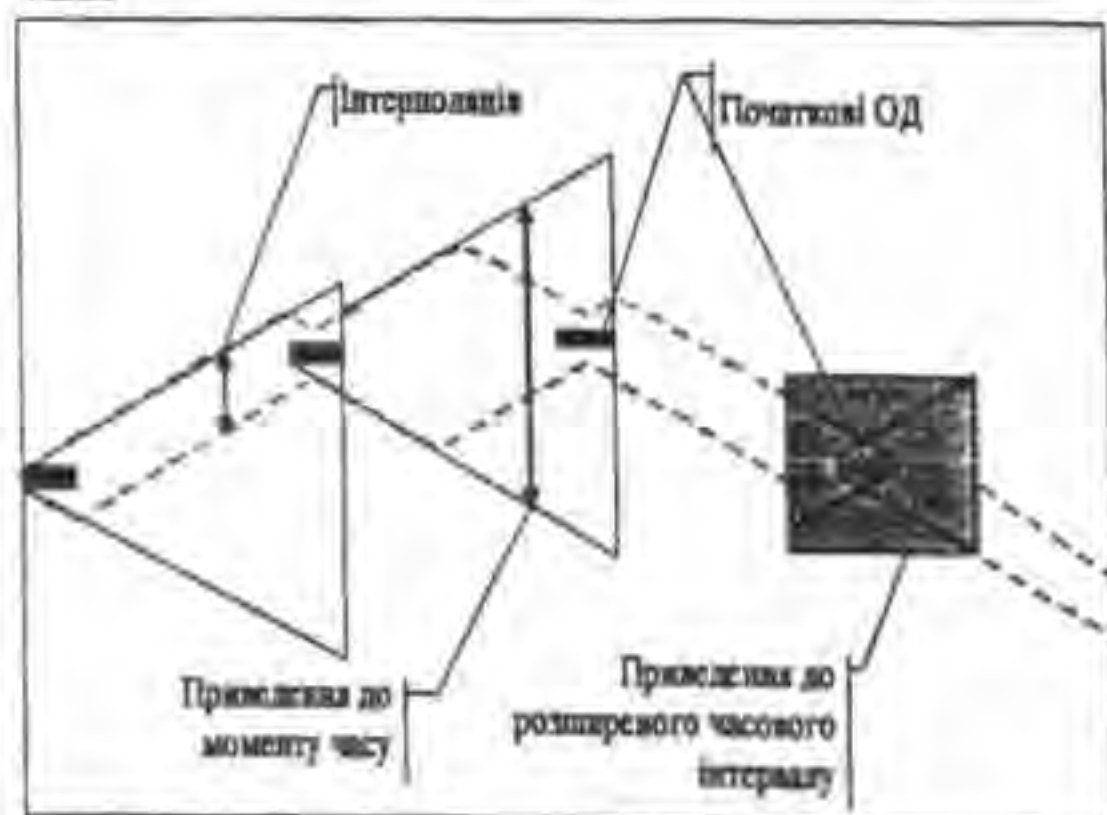
4. Інтерполяція ОД до моменту часу τ . Результатом інтерполяції ОД $A = \langle V_A, t, k_A \rangle$ до моменту часу τ при відомих значеннях A_i, A_{i+1} , таких, що: $\underline{t}_i < \tau < \bar{t}_{i+1}$ називатимемо

ОД $A|_\tau = \langle V_A|_\tau, t, k_A \rangle$, де

$$\begin{aligned} \overline{V_A|_\tau} &= \min(\overline{V_{A_i}} + k_A(\tau - \underline{t}_i), \overline{V_{A_{i+1}}} + k_A(\bar{t}_{i+1} - \tau)), \\ \underline{V_A|_\tau} &= \max(\underline{V_{A_i}} - k_A(\tau - \underline{t}_i), \underline{V_{A_{i+1}}} - k_A(\bar{t}_{i+1} - \tau)). \end{aligned}$$

Операція приведення ОД до загального часового інтервалу дозволяє поширити операції (2) на випадок довільних часових інтервалів ОД. Для цього досить зажадати попереднього приведення до загального часового інтервалу, або моменту часу всі ОД, що беруть участь в обчисленні якого-небудь розрахункового параметра. Приклад виконання операцій над об'єктами даних наведено на рис. 1.

Value



8

Рис. 1. Приклад виконання операцій над ОД

2.2 Облік взаємозалежності ОД

Афінна математика [10] дозволяє отримувати точніші оцінки областей значень, ніж інтервальне числення, на основі обліку взаємної залежності між величинами, що виникає в процесі обчислень в довгих обчислювальних ланцюжках, коли результати одних операцій стають вхідними для

інших.

Інтервальне значення V_i i -того ОД у деякий інтервал часу T представимо афінною формою $\hat{V}_i$, тобто поліномом першого ступеня:

$$\hat{V}_i = V_{0i} + \sum_{k=1}^n V_{ki} \varepsilon_k, \text{ де } V_{ki} \in R, \text{ а } \varepsilon_k \in [-1, 1]. \text{ Число}$$

V_{0i} називають центром афінної форми V_i , коефіцієнти V_{ki} — частковими відхиленнями, а ε_k — символами шуму. Таке представлення неявно виражає залежності між величинами.

Афінна форма є інтервалом, що відбиває значення V , тобто, $V = [V_0 - r, V_0 + r]$, де r сумарне відхилення з $\hat{V}$, $\sum_{k=1}^n |V_{ki}|$. В той же час,

будь-яке інтервальне значення $V = [\underline{V}, \bar{V}]$ можна представити афінною формою $\hat{V} = V_0 + \sum_{k=1}^n V_k \varepsilon_k$,

де $V_0 = (\underline{V} + \bar{V})/2$, $V_k = (\bar{V} - \underline{V})/2$ и ε_k — новий символ шуму, що не зустрічається в будь-якій іншій існуючій афінній формі.

Головна особливість афінної арифметики полягає в тому, що за допомогою одного і того ж ε_k можна відобразити невизначеність декількох вимірів [11]. Наприклад, припустимо, що значення V_1 та V_2 представлені афінними формами

$$\hat{V}_1 = 20 - 4\varepsilon_1 + 2\varepsilon_2 + 3\varepsilon_3,$$

$$\hat{V}_2 = 10 - 2\varepsilon_1 + 1\varepsilon_2 - 1\varepsilon_3.$$

Тоді, $V_1 = [11, 29]$ та $V_2 = [6, 14]$, тобто, пара (V_1, V_2) лежить в прямокутнику (рис. 2). Однак, враховуючи залежність, пара $(\hat{V}_1, \hat{V}_2)$ лежить в опуклому багатокутнику симетричному відносно центру (V_{10}, V_{20}) , отриманому в результаті перебору всіх значень шумів ε_1 та ε_2 з інтервалу $[-1, 1]$.

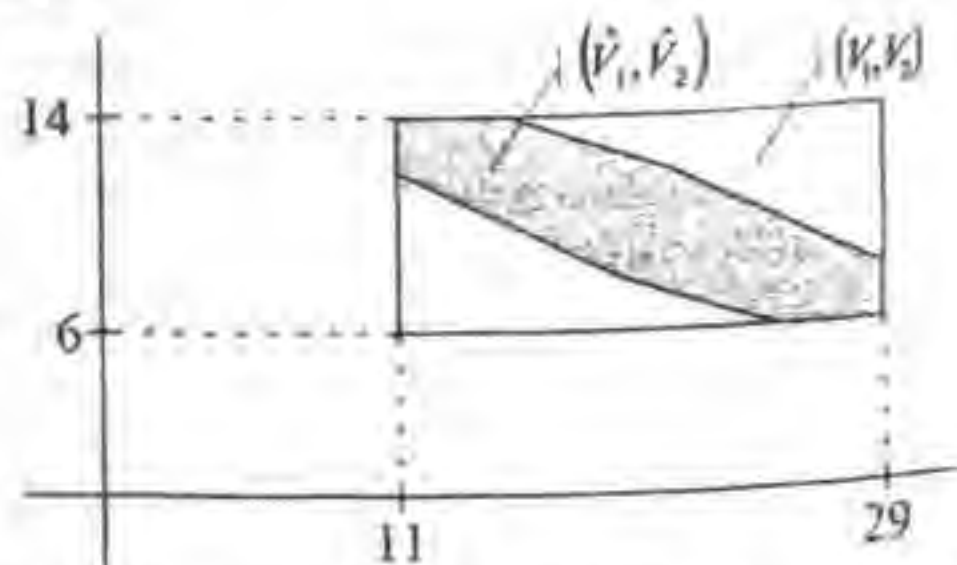


Рис. 2 Приклад об'єднання двох частково залежних афінних форм

При проведенні обчислень з урахуванням взаємозалежності ОД кожна елементарна операція над дійсними значеннями $z \leftarrow f(x, y)$ замінюється відповідною процедурою $\hat{z} \leftarrow f(\hat{x}, \hat{y})$ над афінним формами, результатом якої теж є афінна форма.

Обчислення $\hat{x} - \hat{x}$ в афінній формі дасть точний нуль, тотожності подібні $(\hat{x} + \hat{y}) - \hat{x} = \hat{y}$ не дійсні в інтервальній арифметиці, прашують в афінній (за винятком помилок округлення та переповнення).

Якщо сама операція f є не афінною $z \leftarrow f(x, y)$, то найбільш точний результат буде отримано, якщо уявити z у вигляді суми афінної комбінації шумів і додаткового терма $z_k \varepsilon_k$, який представляє помилку апроксимації:

$$\hat{z} = f^a(\varepsilon_1, \dots, \varepsilon_n) + z_k \varepsilon_k = z_0 + z_1 \varepsilon_1 + \dots + z_n \varepsilon_n + z_k \varepsilon_k.$$

Приклад. Розглянемо множення двох афінних форм, $\hat{z} \leftarrow \hat{x} \hat{y}$, де $\hat{x} = 30 - 4\varepsilon_1 + 2\varepsilon_2$ та $\hat{y} = 20 + 3\varepsilon_1 + 1\varepsilon_3$. Операнди частково корелюють через ε_1 . Згрупуємо терми добутку в афінну частину $A(\varepsilon_1, \dots, \varepsilon_n) = 600 + 10\varepsilon_1 + 40\varepsilon_2 + 30\varepsilon_3$ і квадратичний залишок $Q(\varepsilon_1, \dots, \varepsilon_n) = (-4\varepsilon_1 + 2\varepsilon_2)(3\varepsilon_1 + 1\varepsilon_3)$. Оцінка діапазону Q дасть інтервал $\bar{Q} = [-24, 24]$, отже афінна форма $\hat{z} = 600 + 10\varepsilon_1 + 40\varepsilon_2 + 30\varepsilon_3 + 24\varepsilon_4$ доводить, що z лежить в діапазоні $600 \pm 104 = [496, 704]$. Фактичним діапазоном $\hat{x} \hat{y}$ буде $[528, 675]$, тому діапазон отриманий за допомогою афінної арифметики тільки в 1,42 рази ширше, ніж точний. Для порівняння, діапазон, отриманий за правилами інтервальної арифметики у 3,26 рази ширше. Велика невідповідність виходить з-за негативної кореляції між x і y , зазначеної шумом ε_1 . Кореляційні терми $-80\varepsilon_1$ і $90\varepsilon_1$ майже врівноважуються в афінній арифметиці, але додаються в інтервальній.

2.3 Обробка помилки округлення

Для обліку помилки округлення розширимо результуючу формулу $\hat{z}$ додатковим термом $z_k \varepsilon_k$, де z_k верхня межа суми всіх абсолютних помилок d_i , внесених обчисленням коефіцієнтів z_i та ε_k — шум, що не входить в інші афінні форми.

Приклад. У роботі [12] розглянуто задачу

знаходження послідовності: $I_n = \frac{1}{e} \int_0^1 x^n e^x dx$, для

якої справедливо: $I_0 = 1 - e^{-1}$, а для $n > 0$ доводиться рекурентна рівність: $I_n = 1 - n \cdot I_{n-1}$. Оче-

видно, що $\forall n, 0 < I_n \leq \frac{1}{(n+1)}$. Через накопичення

помилок округлення пряма розрахункова схема є нестійкою, оскільки розрахункова помилка, внесена при визначенні I_0 , зростає пропорційно $n!$. При збільшенні точності розрахунку I_0 зростає максимальна величина n , при якій похибка розрахунку залишається допустимою, але сам процес на певному етапі все одно втрачає стійкість (рис. 3).

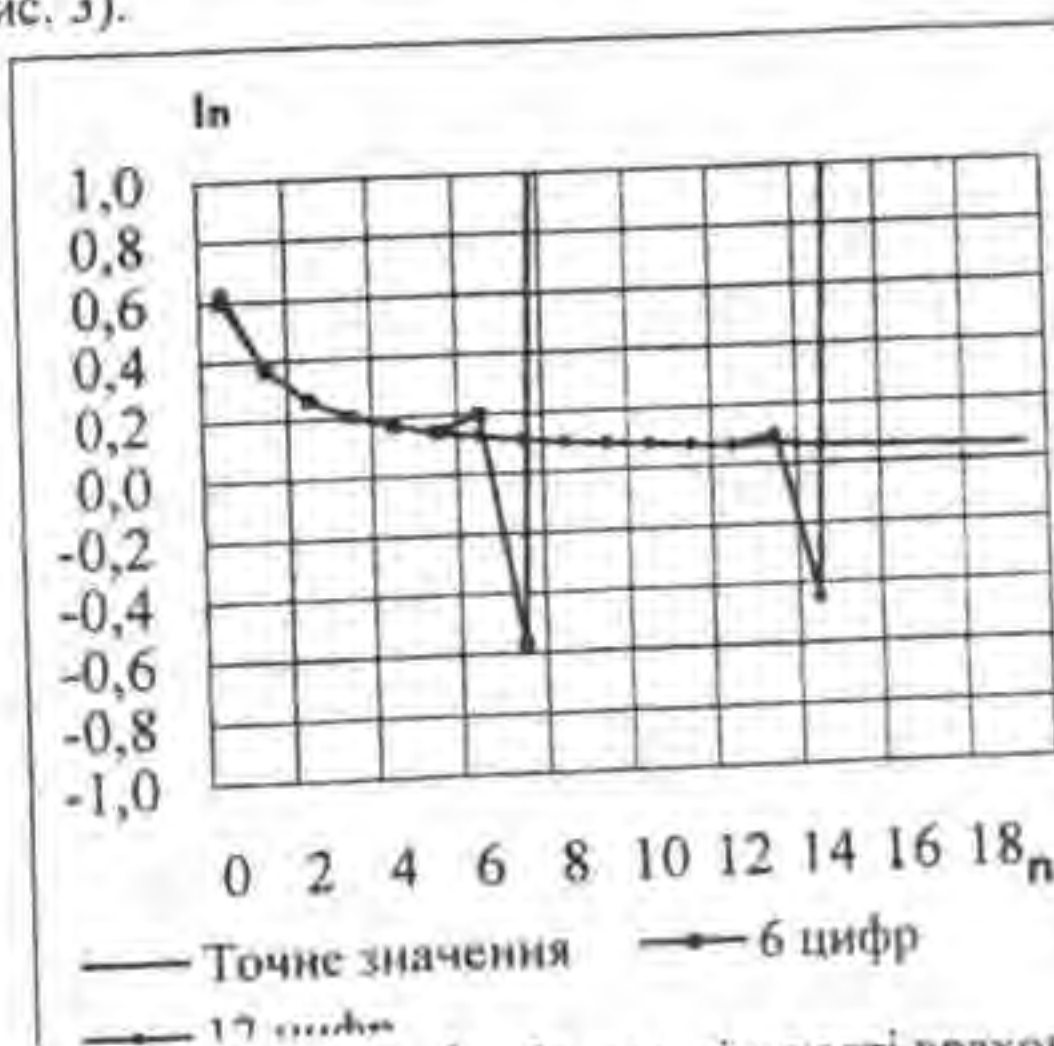


Рис. 3 Залежність I_n від n та кількості врахованих цифр при розрахунку I_0

Розрахункова залежність шуму для прямої і зворотної схеми розрахунку доводить необхідність контролю коректності навіть найпростіших алгоритмів, і можливість виконання такого контролю за допомогою афінних перетворень.

Точне значення на рис. 2 отримано з зворотної рекурентної схеми: $I_{10} = 0, I_{n-1} = \frac{I_n - 1}{n}$. Їх

застосуванням правил афінних операцій, припускаючи, що відбувається округлення до 12 цифр, для обчислень в прямому напрямку отримаємо оцінку шуму ε_n для кожного I_n : $\varepsilon_0 = 0.5 \cdot 10^{-12}$, $\varepsilon_n = n \cdot \varepsilon_{n-1} + \varepsilon_0$, а при обчисленнях зворотному:

$$\varepsilon_{10} = (30 + 1)^{-1}, \varepsilon_{n-1} = \frac{\varepsilon_n}{n} + 0.5 \cdot 10^{-12} \text{ (рис. 4).}$$

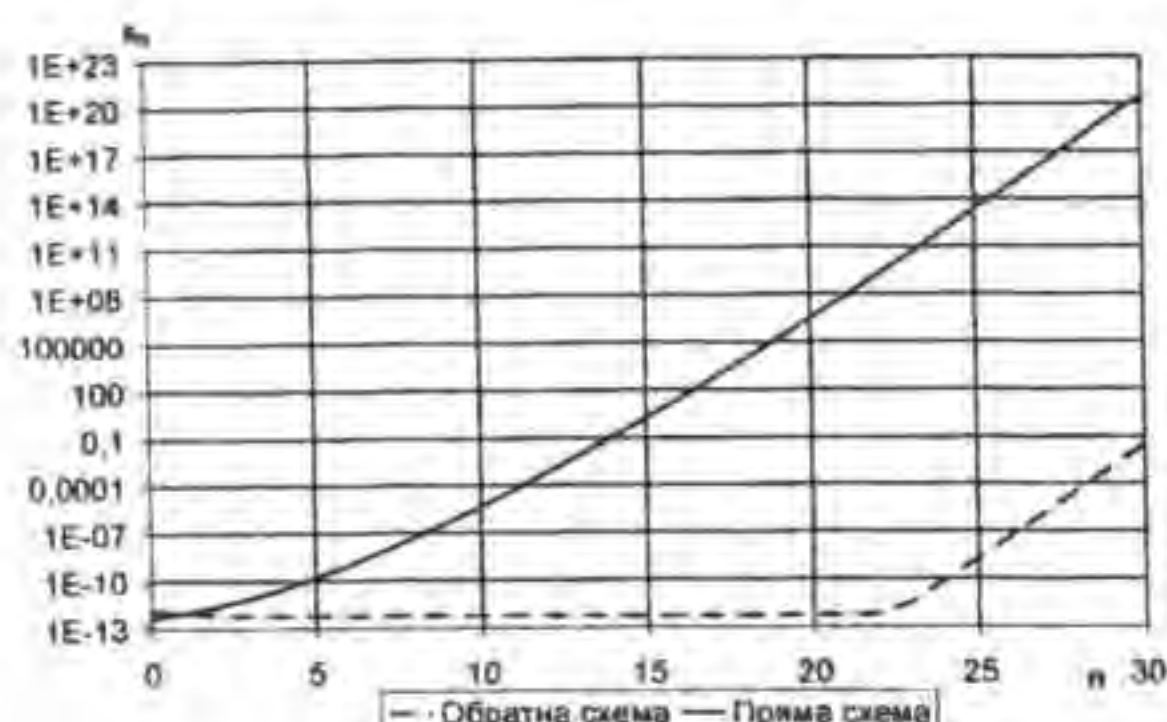


Рис. 4 Залежність шуму від n для прямої і зворотної схеми розрахунку.

Висновки

В даний час склад телеметричних параметрів, необхідний для оцінювання функціонування КА, значно розширився. Цьому сприяють розробка нових КА та електронно-обчислювальних

засобів, розширення кола вирішуваних завдань і розвиток сучасних інформаційних технологій, які дозволяють проводити обробку повного потоку ТМІ в реальному масштабі часу. Розроблювані інтелектуальні інформаційні технології висувають високі вимоги до вихідної вимірювальної інформації перш за все із-за обмеженості часу, відведеного на рішення завдання.

Запропонована в статті модель подання даних використовує інтервальне представлення вимірюваної величини та часу її одержання, містить динамічну характеристику, що дозволяє здійснювати автоматичний контроль вірогідності динамічно мінливих даних на основі афінно-інтервальної математики адаптивного планування запитів до зовнішніх пристроїв та автоматичний контроль вірогідності даних. Таким чином, застосування методів афінно-інтервальної математики вирішує важливе завдання забезпечення достовірності результатів обробки ТМІ.

Список використаних джерел

1. Современная телеметрия в теории и на практике. Учебный курс. / А.В. Назаров, Г.И. Козырев, И.Ш. Шитов и др. – СПб.: Наука и Техника, 2007. – 672 с.
2. PACKET TELEMETRY recommendation for space data system standards [електронний ресурс] // - режим доступа: <http://public.ccsds.org/publications/archive/102x0b5s.pdf>
3. Каргин В. А. Особенности обработки телеметрической информации ракет-носителей в реальном масштабе времени. / В. А. Каргин, Н. В. Нездоровин, Д. А. Николаев // Информация и космос. – 2009. – №4. – С. 77-82.
4. Смагин В.А. Информационный подход оценивания точности метрологических систем / В. А. Смагин, Р. О.Лавров, А. И.Пихтов // Информация и космос. – 2010. – №1. – С. 42-44.
5. Шерешевский Л.А. Методика контроля точности и достоверности расчетных данных в распределенных системах мониторинга и автоматизированного управления / Л. А.Шерешевский, А. А.Сидоров // Известия Самарского научного центра РАН. – 2002. – №1. – С. 69-77.
6. Смагин В.А. Элементарное введение в точностную теорию надежности программного обеспечения / В. А.Смагин

// Надежность. – 2004. – №4. – С. 33–39.

7. МИ 2174-91 ГСИ. Аттестация алгоритмов и программ обработки данных при измерениях. Основные положения [электронный ресурс]. введ. 1.01.1992. // - режим доступа: http://www.gametest.ru/doc/sw/2174_91.pdf
8. Шокин Ю.И. Интервальный анализ / Ю. И. Шокин // - Новосибирск: Наука, 1981. – 112 с.
9. Вошинин А.П. Интервальный метод калибровки / А.П. Вошинин, Н.В. Скибицкий // Системы и датчики. – 2000. – №7. – С. 52-60.
10. Stolfi J. Self-validated numerical methods and applications / J. Stolfi, L. H. de Figueiredo // - Brazilian Mathematics Colloquium monograph, IMPA, Rio de Janeiro, Brazil; 1997. – 116 p.
11. Stolfi J. An Introduction to Affine Arithmetic / J. Stolfi, L.H. Figueiredo // TEMA Tend. Mat. Apl. Comput. – 2003. – №3. – С. 297-312.
12. Бабушка И. Численные процессы решения дифференциальных уравнений / И. Бабушка, Э. Витасек, М. Прагер // - М.: 1969. – 368 с.

Надійшла до редколегії: 16.12.2010

УДК 004.056.55

Чайка^{*} Л.О., Жихаревич В.В., к.ф.-м.н., доц.,
Остапов С.Е., д.ф.-м.н., проф.

Криптостійкий генератор бінарної ключової послідовності на основі клітинних автоматів

У даній роботі розроблено та реалізовано генератор псевдовипадкової бінарної послідовності на основі одномірних клітинних автоматів. Запропоновано детермінований алгоритм вибору чергового біта послідовності, який може бути легко відтворений на приймальній стороні. Проведено статистичний аналіз отриманих результатів за допомогою пакету NIST-STS.

Ключові слова: бінарні послідовності, клітинні автомати, початковий стан, псевдовипадковий генератор.

^{*}E-mail: lidval@rambler.ru

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Сьогодні захист інформації набуває особливої актуальності у зв'язку з різким розвитком об'ємів інформації, що передаються відкритими мережами.

Для захисту такої інформації найчастіше використовуються потокові шифри, основним компонентом яких є генератор ключової послідовності. Такі генератори повинні бути простими і швидкодіючими, з метою їх використання для аудіо і відео інформації.

Існує досить багато генераторів бінарного ключового потоку, проте незважаючи на достатню степінь надійності, їх швидкодія бажає кращого. Саме тому розробка простих швидкодіючих та надійних генераторів псевдовипадкових бінарних послідовностей набуває актуальності.

Розрахунки та результати

Метою даної роботи є розробка генератора бінарної ключової послідовності, який би задовольняв наступним вимогам:

1. Згенерована послідовність повинна проходити статистичні тести на випадковість;
2. Послідовність повинна легко відтворюватись на приймальній стороні;
3. Швидкодія генератора повинна бути достатньою для потокового шифрування даних.

L. O. Chaika, V. V. Zhikharevich, Phd, S. E. Ostapov, doctor of science

Cryptoresistant generator of binary key sequences on the basis of cellular automata

In this paper the generator of pseudorandom binary consecution is developed and realized on the basis of unidimensional cellular automata. The determined algorithm of regular bit selection, which can be easily reproduced on a receiving side, is offered. The statistical analysis of the obtained results is conducted by the package of NIST-STS.

Key Words: binary sequences, cellular automata, initial state, pseudorandom number generator.

В якості генератора псевдовипадкової бінарної послідовності використовується одномірний клітинний автомат [1], який являє собою масив клітин $C_1, C_2, \dots, C_{256}$, стан яких в наступний момент часу визначається за правилом:

$$C'_i = C_{i-1} \oplus (C_i \vee C_{i+1}) \quad (1)$$

Псевдовипадкова послідовність бітів може генеруватись будь-якою з 256 клітин одновимірного масиву або з використанням комбінації бітів.

На першому етапі розробки [3] була проаналізована ймовірність генерації двійкових представлень чисел та проводився аналіз періодичних кореляцій – ймовірність появи «1» на кожному наступному кроці бінарної послідовності (0), через один крок (1), через два кроки (2) і т.д.

Проте прості статистичні тести не дозволяють судити про криптографічну стійкість згенерованої послідовності. Для в'яснення цього питання ми задіяли статистичний пакет NIST-STS, розроблений національним інститутом стандартів і технологій США [4] спеціально для тестування подібних генераторів. Отримана послідовність подається на вхід цього пакету. На виході ми отримуємо результати 189

статистичних тестів. Згідно з методикою тестування, якщо ймовірність проходження перевищує 0,9615 на масиві 10^6 , то тест вважається пройденим. Найкращі генератори повинні проходити всі тести NIST-STS.

На першому етапі роботи з пакетом NIST-STS було вирішено проаналізувати чи взагалі можливо розробити криптостійкий генератор на основі клітинних автоматів. Для цього

використовувалась наступна схема «random-30-random»: стартовий стан генератора вибирався випадковим чином, до згенерованого масиву застосовувалось правило 30 взаємодії клітинних автоматів, а біт для виведення у файл обирався псевдовипадковим чином. Отримані результати (рис.2) дозволили зробити висновки, що побудова такого генератора можлива.

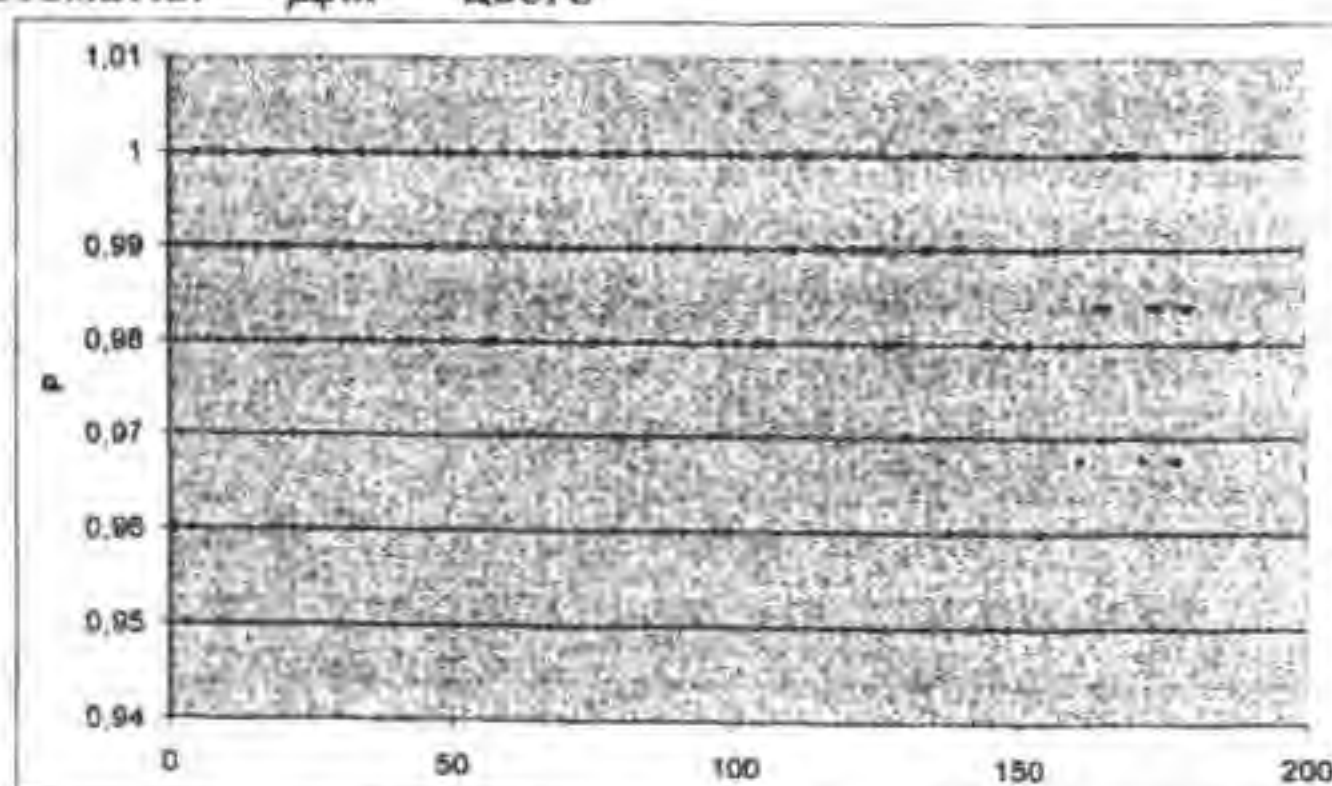


Рис. 1. Результат тестування статистичних характеристик псевдовипадкової бінарної послідовності за схемою «random-30-random»

Проте випадковий характер стартового стану та виведення результуючих бітів послідовності не дає можливості відтворити стан генератора на приймальній стороні. Розшифрування зашифрованого потоку даних неможливе. Тому було вирішено розробити псевдовипадкові алгоритми утворення стартового стану та виведення чергового біту ключової послідовності, які можна б було легко відтворити на приймальній стороні. Для цього було

проведено ряд досліджень на вплив способу виведення бітів послідовності.

На першому кроці стартове заповнення масиву забезпечувалось випадковим чином, до нього застосовувалось правило 30 та весь масив в 256 бітів виводився разом. Отриманий результат показаний на рис.3. З нього видно, що середнє значення ймовірності дорівнює 0,77, а це показує, що більшу частину тестів не пройдено.

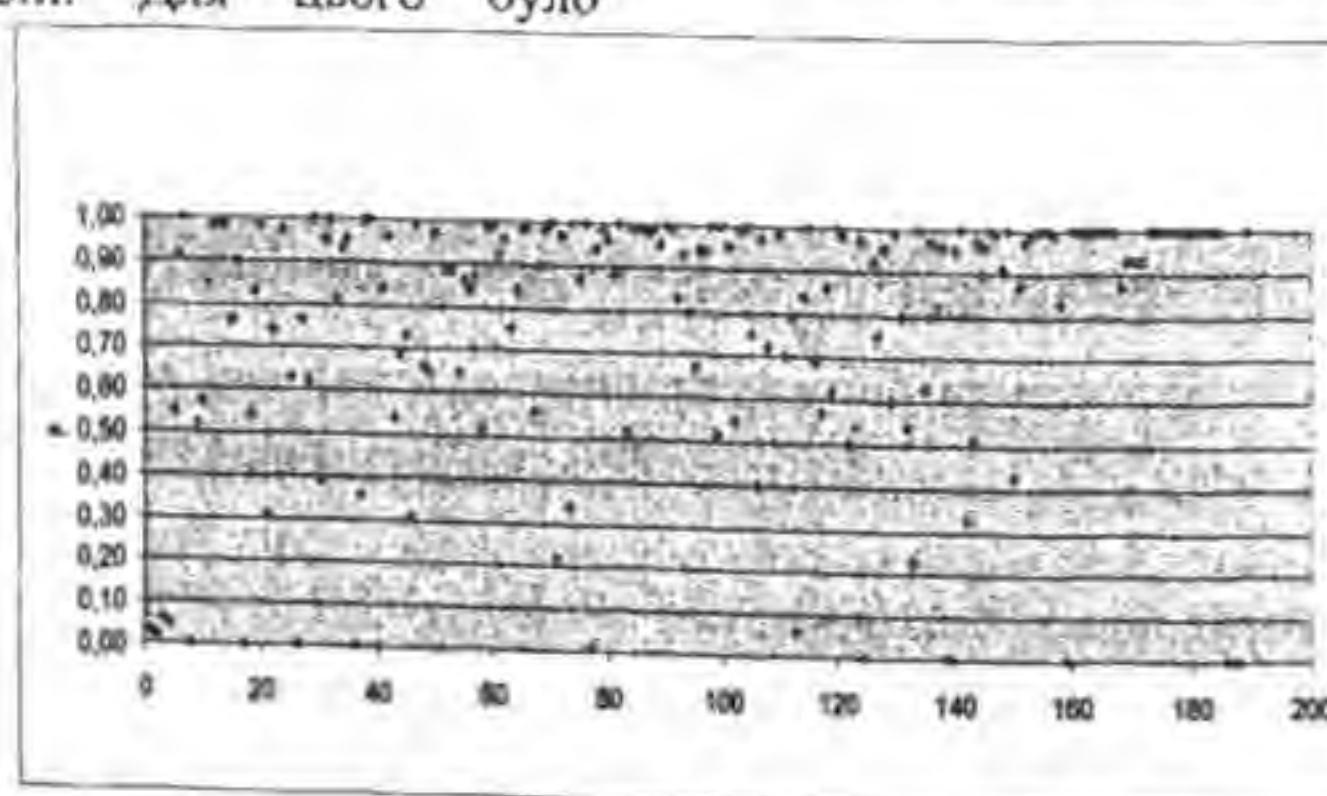


Рис. 2. Результат статистичного аналізу псевдовипадкової бінарної послідовності, згенерованої на основі правила 30 з суцільним виводом масиву

Для покращення результатів комбінувалися кілька бітів масиву за допомогою операції XOR за наступним правилом

$$M[w] \cdot \text{XOR} \cdot M[v] \cdot \text{XOR} \cdot M[t] \quad (2)$$

де w, v, t – номери бітів масиву.

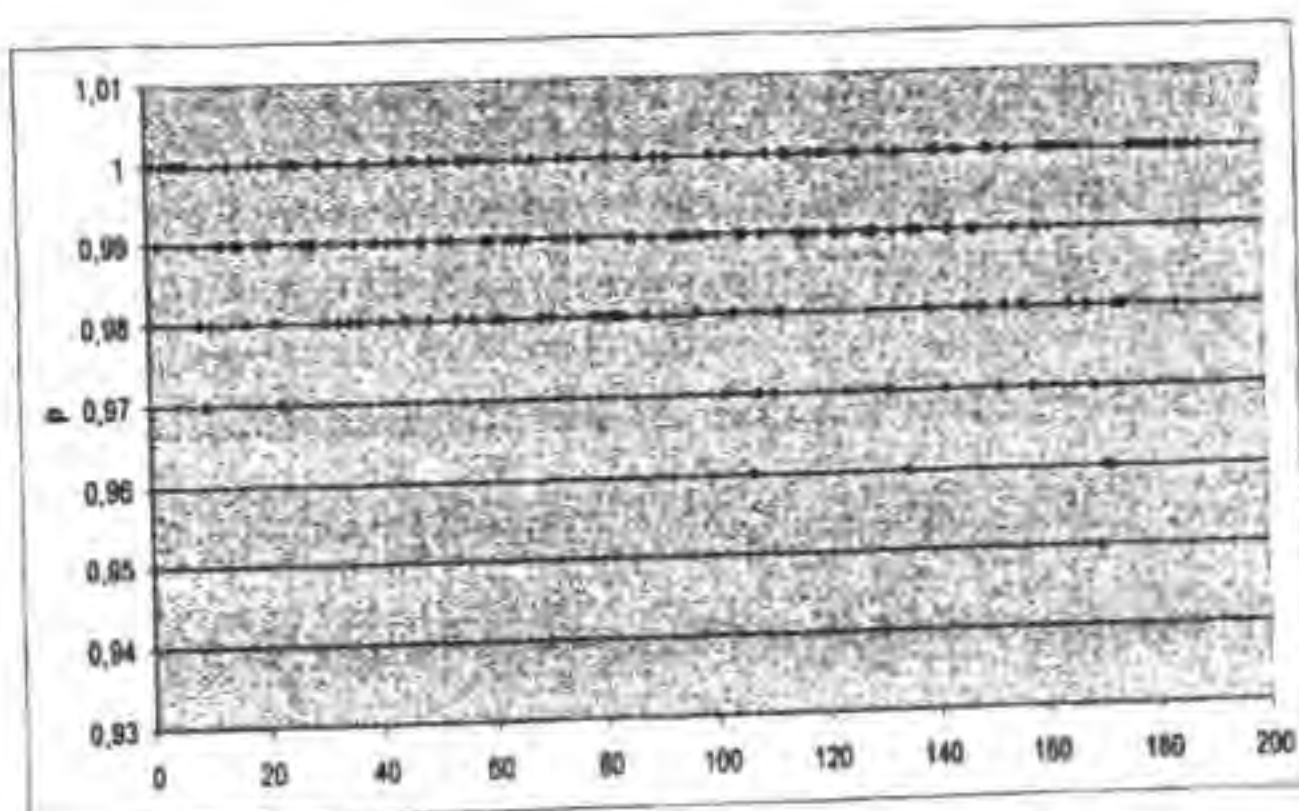


Рис. 3. Результат статистичного аналізу псевдовипадкової бінарної послідовності з комбінованим виводом масиву

Як видно з рис.4. отримані результати значно кращі попередніх, тільки два значення менші 0,96, отже спосіб виводу бітів послідовності значно впливає на криптостійкість генератора. Це підтверджує наступний крок – вибір чергового біта w з (2), задається формулою:

$$w = (i + 3 + \sum_{k=0}^{(\log_2 n)-1} C_{i+3+k} \cdot 2^k) \bmod n, \quad (3)$$

де C – значення $i+3+k$ -комірки, n – загальна кількість комірок.

Рис. 4 демонструє навіть кращі результати, ніж такі на рис. 3 – тільки одне значення менше потрібного 0,96.

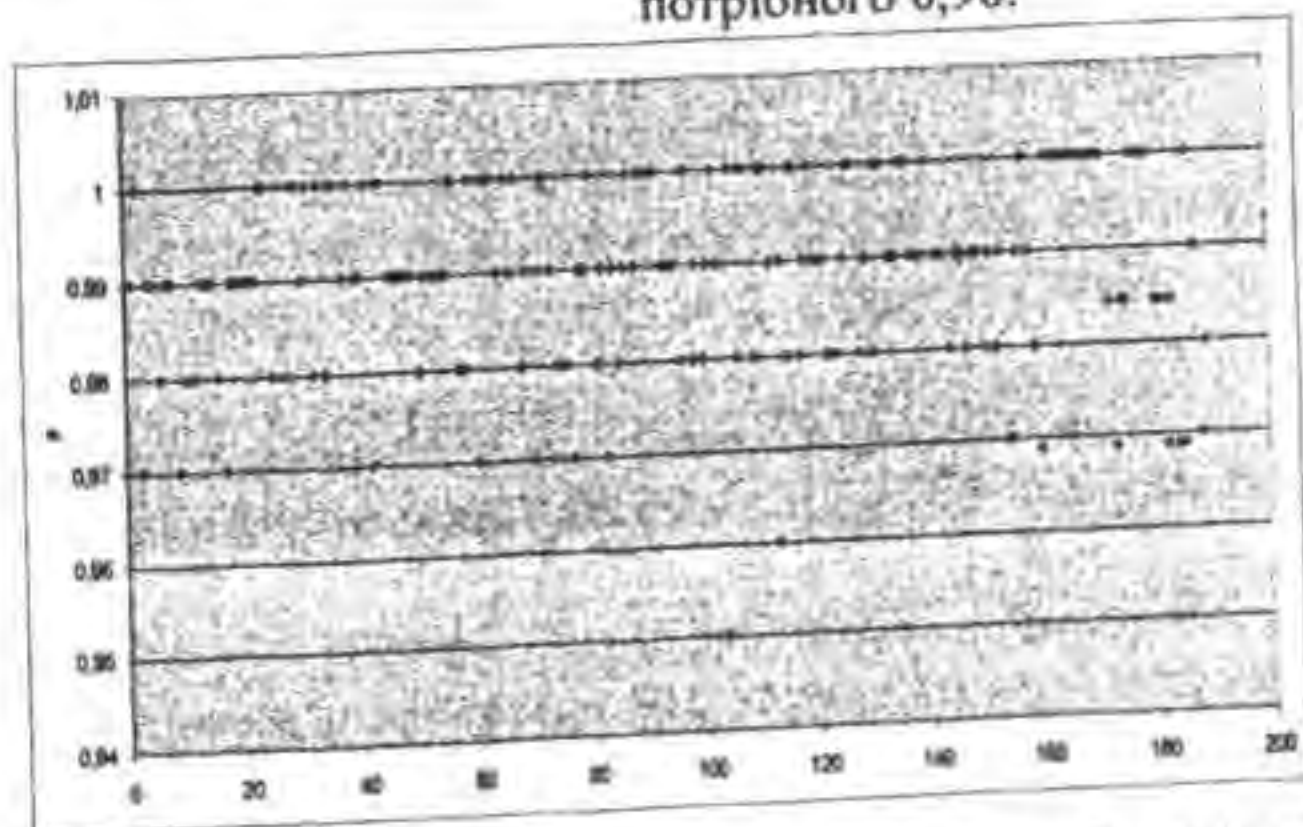


Рис. 4. Результат статистичного аналізу псевдовипадкової бінарної послідовності з комбінованим виводом масиву

Отже з проведених досліджень можна зробити наступні висновки: статистичні властивості генератора бінарного потоку сильно залежать від способу виведення чергового біту послідовності; згенерована послідовність проходить тести NIST-STS у випадку комбінованого виводу бітів і не проходить у випадку їх послідовного виведення.

Наступним етапом була зміна стартового стану генератора. Ініціювався масив з нульових елементів, у якому лише 128-й елемент дорівнював 1. Для виявлення оптимального

стартового стану виконувалось від 150 до 550 холостих циклів взаємодії клітин без виводу і лише потім починалась основна обробка масиву. Отримані результати (рис.6) показують, що чіткої залежності середнього значення тестів NIST-STS від кількості холостих циклів виявлено не було. Усі значення скупчені навколо 0,98 і помітно, що максимальне значення спостерігається при 250 циклах. Тобто статистичний портрет генератора в цьому випадку практично не залежить від його стартового стану.

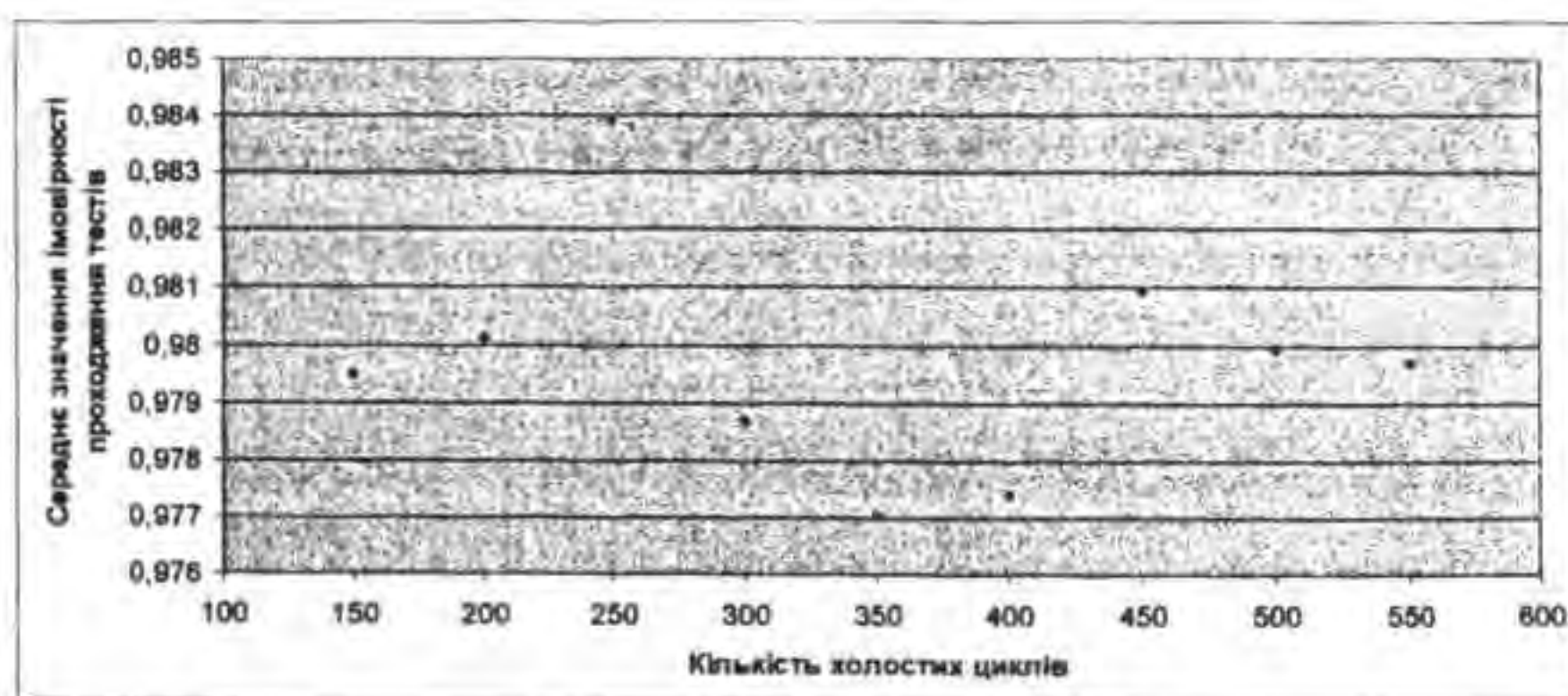


Рис.5. Залежність середнього значення тестів NIST-STS від кількості холостих циклів

Останнім кроком, було дослідження іншого механізму взаємодії клітин, який враховує так звану «далекодію», тобто взаємодію не з найближчими сусідами. Як і на попередньому кроці масив з 256 бітів заповнювався нулями і тільки в його 128 біт записувалась 1.

Виконувалось 250 холостих циклів взаємодії без виводу, після чого стартовий масив оброблявся за правилом 30 «з далекодією». Виведення бітів послідовності відбувалося за (2), тобто комбінацією кількох бітів масиву.

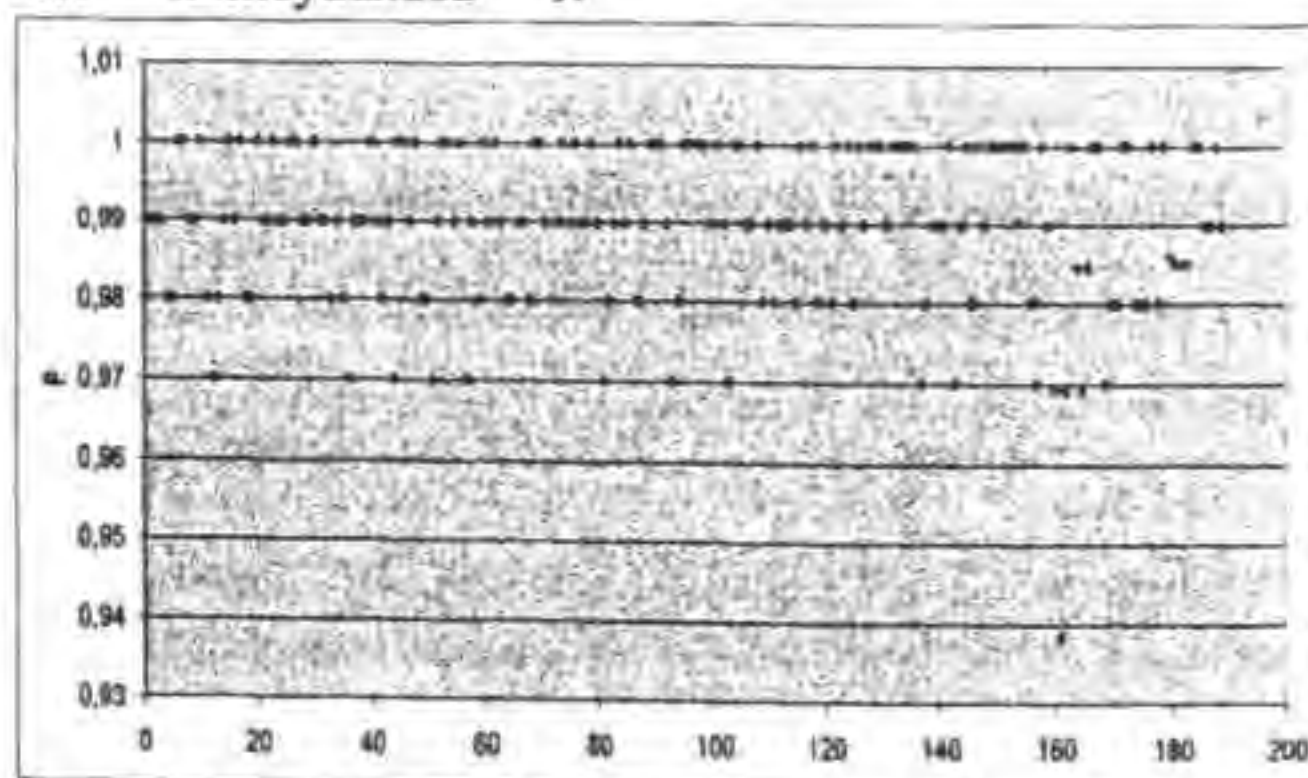


Рис. 6. Результат статистичного аналізу псевдовипадкової бінарної послідовності з «далекодією»

Результати показані на рис.6. є найкращими з отриманих, – тільки одна точка знаходиться нижче 0,96.

Аналогічні дослідження проводились і для правил 86 та 150. Результати статистичних аналізів отриманих послідовностей приведені на рис. 7 та 8 відповідно.

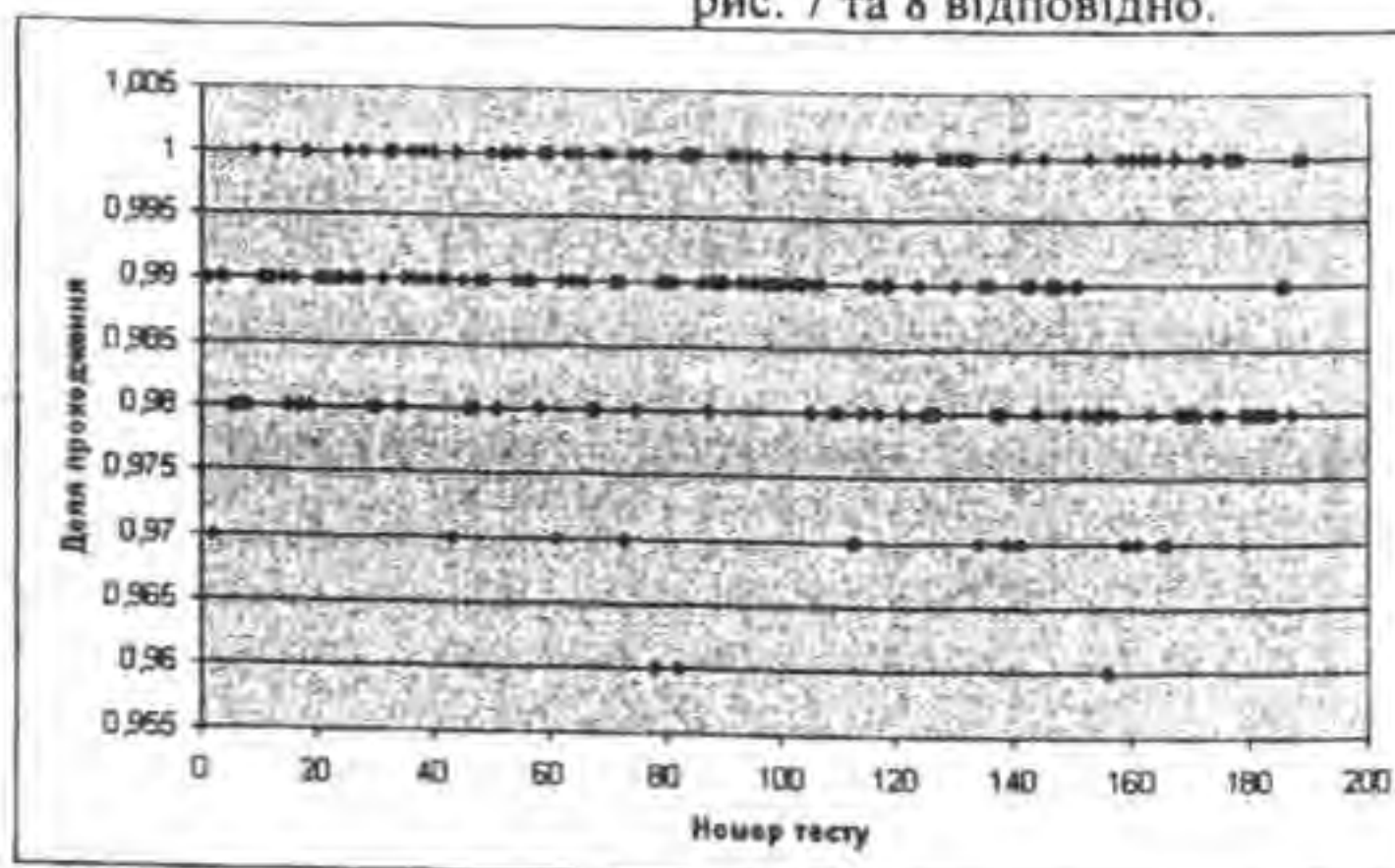


Рис. 7. Результат статистичного аналізу псевдовипадкової бінарної послідовності, згенерованої на основі правила 86

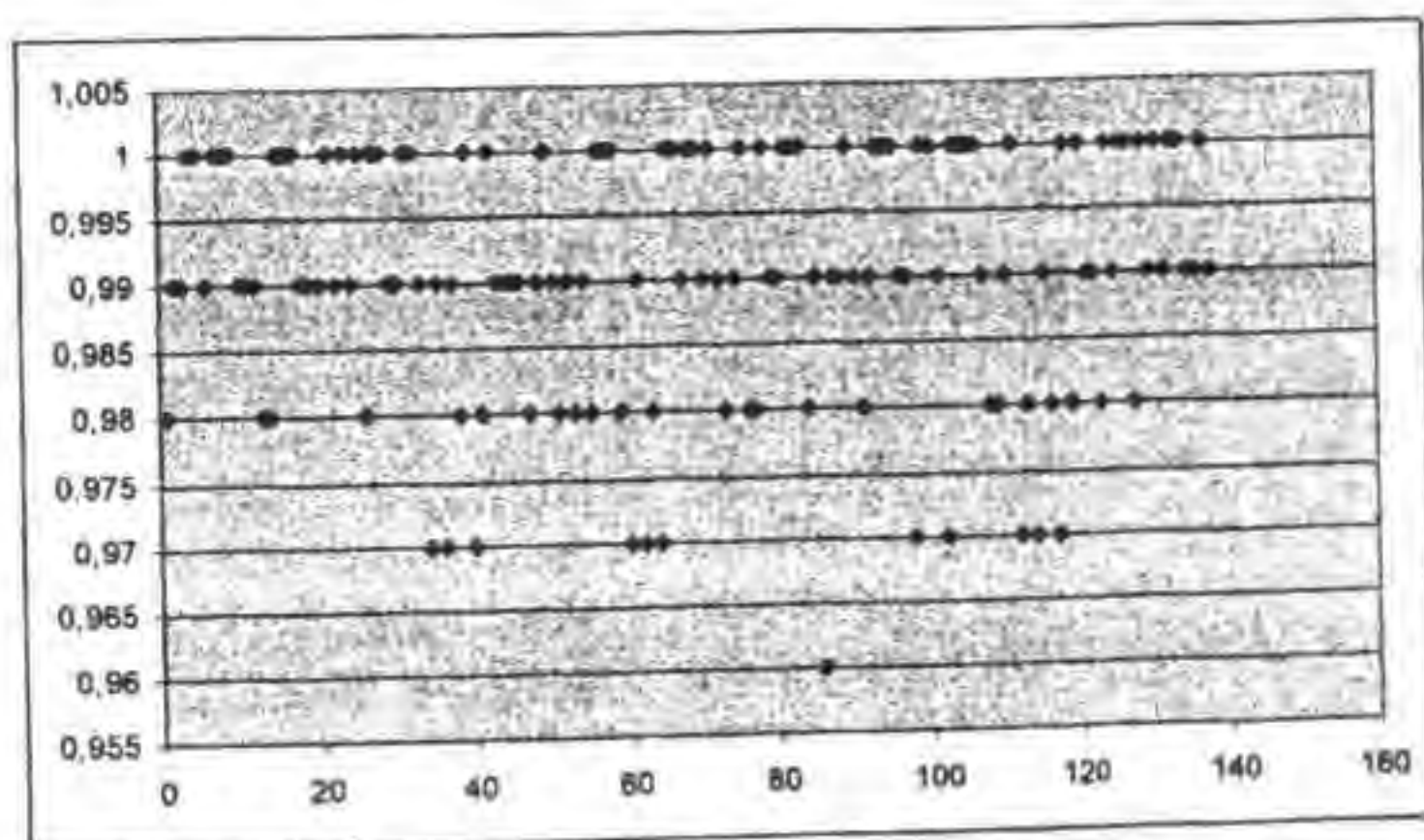


Рис. 8. Результат статистичного аналізу псевдовипадкової бінарної послідовності, згенерованої на основі правила 150

Висновки

Підсумовуючи проведені дослідження, можна зробити наступні висновки: розроблено та реалізовано програмний генератор псевдовипадкової бінарної ключової послідовності на основі клітинних автоматів (правил 30, 86, 150), як простого, так і з «далекодією»; виконано дослідження статистичного портрету генератора за допомогою

пакету NIST STS; виявлено сильну залежність якості ключової послідовності від способу виведення її бітів; не виявлено залежності якості послідовності від стартового стану генератора, що свідчить, на нашу думку, про досить високу його якість.

Розроблений генератор можна використати у системах потокового шифрування.

Список використаних джерел

1. S. Wolfram. Random sequence generation by cellular automata // *Advances in Applied Mathematics*. – 1986. – v.7. – P. 123-164.
2. W. Meier and O. Staffelbach. Analysis of pseudo random sequence generated by cellular automata // *Advances in Cryptology EUROCRYPT '91 Proceedings*, Springer-Verlag. – 1991. – P. 186-199.
3. Sergey Ostapov, Vladimir Zhikharevich, Lidiya Val'. Investigation of Properties of

Pseudorandom Binary Sequences Generator on the Basis of Cellular Automata. // *Proceeding of the 9-th International Conference on DAS*. – May 2008. – P. 115-117.

4. Andrew Rukhin, Juan Soto, James Nechvatal. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22; U.S. Government printing office Washington: 2000.

Надійшла до редколегії 14.12.10

УДК 004.42:512.58

Ченцов* О.І., к.ф.-м.н., н.с.

Програмна реалізація теорії категорій

У роботі розглядаються підходи до програмної реалізації теоретико-категорних конструкцій. Проводиться порівняння відомих підходів до формалізації категорій в реальних задачах, що знайшли свою програмну реалізацію. Запропонована програмна реалізація теорії категорій на основі обчислювального підходу засобами параметризованого програмування.

Ключові слова: теорія категорій, параметризоване програмування, формалізація.

*E-mail: chentsov@ukr.net

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

1. Огляд підходів до програмної реалізації категорій

Нижче перелічуються основні підходи до програмної реалізації категорій в цілому. Основна мета таких реалізацій дати базовий апарат, за допомогою якого можна будувати велику кількість різноманітних об'єктів (математичний універсум), досліджувати їх властивості та практично розв'язувати широке коло задач категорними методами.

Онтологічний підхід. Даний підхід до формалізації категорій був зреалізований Р. Кентом в проекті IFF [1, 2]. В ньому категорії строяться зверху класів (множин) та графів. Деякою особливістю формалізації є відмінне від традиційного представлення категорної композиції та тотожних морфізмів за допомогою 2-морфізмів графів та операцій на графах. Загальна онтологія категорій представляє середній рівень системи IFF. На її основі можна будувати спеціалізовані онтології. Для формалізації категорій використовується варіант SUO-KIF мови KIF, що є мовою для представлення висловлювань логіки першого порядку. Деякі шляхи побудови спеціалізованих онтологій на основі категорій були запропоновані Н.Глазуновим в роботі [3]. Приймається постулат, що онтологія може бути представлена за допомогою графу. Внаслідок цього онтологію можна категоризувати на основі теорії представлень колчанів. Існує декілька рівнів категорифікації, зокрема, розмічені колчани. Маючи категорне представлення онтології її можна досліджувати методами

O. I. Chentsov*, Cand. of Sc.

Implementation of category theory in generic programming

This paper considers approaches to formalization of category theory. Known implementations of category theory in computer-oriented languages are compared. Implementation of categories based on computational approach and generic programming is presented.

Key Words: category theory, generic programming, formalization.

гомологічної та гомотопічної алгебри. Обмеження онтологічного підходу до формалізації категорій є тим же, що і у застосування онтологій взагалі. Дана онтологія дає точну специфікацію понять теорії категорій, дозволяє на їх основі визначати інші поняття, але не є зручною та ефективною для виконання конкретних теоретико-категорних обчислень.

Логіко-алгебраїчний підхід до формалізації категорій. Перша така формалізація була виконана Дж. Гогеном на мові алгебраїчної специфікації та верифікації OBJ3[4]. OBJ є системою з впорядкованою багатоосновною еквациональною логікою. Програми OBJ мають денотативну семантику багатоосновних алгебр. Водночас операційною семантикою програм є перетворення на основі правил, що визначаються рівняннями. Система оперує двома поняттями – об'єктами та теоріями. Об'єкти інтерпретуються стандартними моделями, які є ініціальними алгебрами відповідних алгебраїчних систем. Тому твердження стосовно об'єктів можуть обчислюватися шляхом перетворень на стандартних моделях. Вирази при цьому редукуються до констант, наприклад, до true основи Bool. Теорії інтерпретуються множиною всіх моделей. Вони групують набір властивостей. Основне призначення теорій накладати обмеження на параметри, що фігурують у інших теоріях та об'єктах. Зв'язування теорій з об'єктами відбувається за допомогою представлень. В даному підході категорії формалізуються у вигляді теорій. Конкретні категорії будуються як об'єкти і зв'язуються із

теорією через представлення. Будь-яке обчислення в OBJ є доведенням деякої теореми. Отже однією з основних задач даного підходу до формалізації категорій є перевірка правильності побудов. Базуючись на отриманій формалізації категорій можна робити нові аксіоматичні побудови, перевіряти їх властивості. Також можливо перевіряти чи відповідають конкретні системи категорій аксіоматиці. Використання алгебраїчних специфікацій для ефективного обчислення алгоритмів є досить сумнівним.

У подальшому даний підхід був розвинений в інших мовах, зокрема, в Agda. Оскільки мова Agda ґрунтується на теорії типів Мартіна-Льофа [5] і реалізує значну її частину, то в ній може бути формалізована теорія категорій [6]. Іншими формалізаціями в системах алгебраїчних специфікацій та автоматичного доведення є проекти в системах Nuprl [7] та Aldor.

Обчислювальний підхід. Даний підхід був використаний в роботі [6]. В основі його лежить конструктивний обчислювальний аспект теорії категорій, що полягає у тому, що більшість понять та доведень в ній передбачають існування деякого алгоритму, за яким в явному вигляді будуватиметься відповідний математичний об'єкт. Для програмної реалізації цих алгоритмів найбільш адекватним автору видається параметризоване програмування, тому більш докладно про обчислювальний підхід йтиметься у підрозділі 3 цієї роботи.

Окрім універсальних підходів до реалізації значної частини теорії категорій існують також програмні розробки, що формалізують лише частину теорії або використовують її неявно. Про найважливіші з них йдеться нижче.

Категорна абстрактна машина (CAM). Як відомо теоретичною основою функціонального програмування є лямбда-числення. Програмам відповідають лямбда-вирази. З цього погляду виконання функціональної програми полягає у обчисленні відповідного виразу. Обчислення виразу відбувається за допомогою правил перетворення, що визначені в лямбда-численні. Певною складністю при цьому є робота із зв'язаними змінними. Для того, щоби обійти її переходять до такого представлення виразів, в якому змінні не фігурують. Один підхід полягає в використанні комбінаторної логіки. Інший – кодування Дебрейна [8]. Внаслідок відповідності Ламбека виразам лямбда-числення можна співставити морфізми деякої декартово-замкненої категорії. Властивості декартово-замкненої категорії визначають категорну

комбінаторну логіку (ККЛ). Код Дебрейна досить просто переводиться у терми ККЛ. Дані вирази можуть безпосередньо перетворюватися спеціальним автоматом – категорною абстрактною машиною (CAM). В ній символи трактуються як команди CAM. До цих команд належать скобки пар, морфізми проєкцій та аплікації, комбінатори абстракції та введення нової змінної. Хоча робота категорної абстрактної машини повністю ґрунтується на теорії категорій, сама теорія в ній не формалізується. Машина безпосередньо працює із термами ККЛ. При цьому використовуються властивості композиції та еквівалентні перетворення, що ґрунтуються на рівняннях для декартово замкненої категорії. Тобто властивості декартово замкненої категорії приховані у правилах перетворень.

Ідеї категорних машин були поглиблені у проекті Charity [9]. В ньому окрім використання CAM для виконання програм було також використано категорний підхід до побудови системи типів програмної мови. Дана система базується на ідеях співзвучних дисертаційному дослідженню Т. Хагіно [10], у якому було показано, що типи програмної системи можуть бути побудовані на основі функторних алгебр. Хагіно розглядав три рівні: категорних специфікацій (CSL), категорних типів даних (CDT) та категорної програмної мови (CPL). Рівень CSL дозволяє специфікувати категорії, функтори, функторні морфізми та спряження. На рівні CDT визначаються функторні типи даних як ініціальні діалгебри для специфікованих функторів. Нарешті існує рівень категорної програмної мови (CPL), що за своєю суттю поєднує CAM та функторні типи даних, тобто виходить за межі декартово замкненої категорії. Правила перетворення в цій версії категорної машини визначаються властивостями функторів та продукуються автоматично. Таким чином, структури управління програмної мови визначаються структурою типів даних. Внутрішнє представлення функторних типів даних залежить від реалізації. Наприклад, система типів Charity включає базові типи, типи ініціальних функторних алгебр та фінальних функторних коалгебр, для їх представлення використовуються базові структури даних – стеки та дерева.

На ідеях теорії категорій виникли та сформувалися такі напрямки як монадичне програмування та програмування параметризоване по типах даних [11]. Центральне

місце в них грає категорія, об'єктами якої є типи мови програмування, а морфізмами функції в цій мові. Для мови програмування Haskell ця категорія називається *Hask*. Подібно до *CAM* дані напрямки ґрунтуються на категорних ідеях, а не безпосередньо оперують категорними конструкціями.

Алгебраїчний підхід до перетворень графів. Даний підхід до перетворень графів був розвинутий німецько-італійською школою в роботі [12]. Проблема перетворення графів формулюється наступним чином.

Входженням графа H у граф G будемо називати підграф H' графа G та морфізм графів $m: H \rightarrow G$, такий що $m(H) = H'$. Нехай задано деякий граф G та продукції вигляду: $p: L \rightarrow R$. Продукції задають часткову відповідність між вершинами та ребрами графів L та R , визначаючи які мають бути видалені, а які додані. Застосування продукції $p: L \rightarrow R$ до графа G , полягає у заміні входження графа L в граф G на граф R . Внаслідок того, що входження може бути неізоморфним L , можливе виникнення колізій. Наприклад, для деяких вершин входження L у G правило може одночасно вимагати їх видалення і збереження, або правило може вимагати видалення вершини, що є інцидентною ребру з G , що не належить входженню.

Є два основні категорні підходи до розв'язання цієї проблеми:

- одиночних універсальних квадратів (SPO);
- двійних універсальних квадратів (DPO).

Підходи різняться тим, як вони обробляють колізії. DPO повністю забороняє подібні ситуації. Застосування продукції в ньому формалізується наступною діаграмою в категорії графів та тотальних морфізмів графів.

$$\begin{array}{ccccc} L & \xleftarrow{l} & K & \xrightarrow{r} & R \\ \downarrow m & (1) & \downarrow d & (2) & \downarrow m^* \\ G & \xleftarrow{l^*} & D & \xrightarrow{r^*} & H \end{array}$$

Продукцією тут є пара морфізмів $L \xleftarrow{l} K \xrightarrow{r} R$, де K – інтерфейсний граф. Результуючим графом є граф H . На морфізм m накладаються умови стягування, що виключають колізії. Обидва квадрати є універсальними. Зауважимо, що в цьому підході виникає проміжна задача пошуку доповнення пари (l, m) до універсального квадрату (1). Умови

стягування гарантують існування та єдиність (з точністю до ізоморфізму) такого доповнення.

При підході SPO продукція розглядається як морфізм к категорії графів та часткових морфізмів графів. Застосування продукції зводиться до побудови універсального квадрата для $G \xleftarrow{m} L \xrightarrow{p} R$, тобто побудови наступної комутативної діаграми.

$$\begin{array}{ccc} L & \xrightarrow{p} & R \\ \downarrow m & (3) & \downarrow m^* \\ G & \xrightarrow{p^*} & H \end{array}$$

SPO дозволяє колізії і там де є кілька варіантів надає перевагу видаленню вершин та ребер.

В вихідній роботі усі перетворення описані в термінах категорії графів. Проте підходи можуть бути узагальнені на HLR-категорії. Зауважимо, що визначальним для обох підходів є наявність в категорії універсальних квадратів (для DPO також необхідні доповнення до універсальних квадратів). Основною областю застосувань методів перетворень графів є задачі реінженерії баз даних, інформаційних систем та систем на основі ООП[13].

Алгебраїчний підхід до перетворення графів був зреалізований в програмному пакеті AGG [14], що розповсюджується вільно. Цей пакет дозволяє виконувати різні операції із скінченними графами. Для перетворення графів в ньому використовується підхід одиночних універсальних квадратів. Реалізація категорних конструкцій є важливою частиною системи, проте складає дуже незначну частку всього коду. Замість загальних категорних конструкцій в пакеті використовуються виключно реалізація категорії часткових графів та специфічні універсальні конструкції для таких графів.

2 Методологія параметризованого програмування

В даному підрозділі викладаються основи методології параметризованого програмування, актуальні для неї питання та її перспективи. Ці відомості необхідні для викладання матеріалу наступного підрозділу.

Існують два тлумачення терміна параметризоване програмування ("generic programming") — програмування з використанням концептів [15] та програмування параметризоване за типами даних або "орігамі-програмування" [11]. В даній роботі термін вживається в першому розумінні.

В вузькому прагматичному аспекті даній методології відводиться роль методології розробки компонент повторного використання або, простіше кажучи, бібліотек високого степеня абстракції. Порівняння параметризованого програмування з іншими методологіями більш глибоко досліджувалось у [16]. На практиці побудова бібліотек з використанням виключно параметризованого програмування не зустрічається (за виключенням STL). Наприклад, у Boost-бібліотеках також активно використовується шаблонне метапрограмування.

Параметризоване програмування [17] це піддисципліна комп'ютерних наук, що займається пошуком абстрактних представлень ефективних алгоритмів, структур даних та інших програмних елементів, а також їх систематичною організацією. Мета параметризованого програмування – виразити алгоритми та структури даних в адаптованій формі, що забезпечує простоту композиції та можливість безпосереднього використання в виробництві програмного забезпечення. Це дозволяє формувати розв'язок множини однотипних задач один раз. Центральним поняттям є поняття узагальненого алгоритму – параметризованої процедурної схеми, гранично незалежної від представлення даних, і отриманої з конкретних ефективних алгоритмів. Ключовими ідеями параметризованого програмування є такі:

- представлення алгоритмів у мінімальних припущеннях про дані і навпаки, цим досягаються гнучкість та широкі можливості композиції алгоритмів;
- поступовий підйом конкретного алгоритму на узагальнюючий рівень без втрати ефективності, тобто формулювання алгоритму у найбільш абстрактній формі, що при спеціалізації (мономорфізації) не поступається в ефективності оригінальному алгоритму;
- додавання більш загальної форми алгоритму, якщо наявна абстракція не охоплює всіх використань алгоритму (передбачас механізм диспетчеризації, що гарантує вибір найбільш ефективного варіанта алгоритму для конкретних типів-параметрів);
- побудова декількох узагальнених алгоритмів розв'язання однієї задачі одного рівня абстракції, якщо жоден з них не є переважним на довільних даних.

Для формулювання узагальнених алгоритмів потрібен спосіб вказування обмежень, що накладаються алгоритмом на типи-параметри. Коректність алгоритму формулюється в

припущеннях, що дотримані обмеження. Обмеження групуються у концепти. Таким чином, *концепти* — це предикати, задані на множині типів, тобто предикати другого порядку. Типи $T_1, T_2, \dots, T_n$ моделюють концепт, якщо вони задовольняють відповідним предикатам. Вважається, що концепт C' уточнює концепт C , якщо типи, що моделюють концепт C' , також моделюють концепт C .

Розвиток параметризованого програмування відбувається в наступній тріаді: таксономії концептів — дженерики — методологія параметризованого програмування. Таксономії — це пов'язані відношення уточнення набори концептів, що описують певну предметну область. *Дженерики* — засоби мови програмування для представлення конструкцій параметризованого програмування [16].

Вивчати таксономії зручніше в порядку їх уточнення. Виробляються ж таксономії концептів індуктивно від конкретних ефективних алгоритмів до їх узагальнених версій шляхом підйому рівня абстрагування.

Оскільки суттєва роль в параметризованім програмуванні відведена ефективності, то мова C++ стала стандартом де факто реалізації бібліотек з використанням представленої методології. На думку творців програмування з концептами на даний момент достатньо добре розроблена таксономія для базових узагальнених алгоритмів. Це таксономія концептів бібліотеки STL. Слідуючи ідеям STL, з'явилися бібліотеки ефективного обробки графів (Boost Graph Library) та матричних обчислень (MTL). На даний момент налічується більш 80 boost-бібліотек.

Як засвідчує аналіз STL для повноцінної реалізації ідей параметризованого програмування необхідно передбачити наступні можливості [16]:

- параметризація функцій;
- обмеження на типи-параметри:
 - сигнатурні обмеження — наявність у типів методів, зокрема, параметризованих за типами;
 - асоційовані типи та обмеження на них;
 - тотожність типів та приведення одних типів до інших;
- групування обмежень у концепти;
- анонімні функції;
- диспетчеризація по концептах — механізм, що дозволяє автоматично обирати з альтернативних параметризованих алгоритмів найбільш оптимальний;

- механізм створення абстрактних типів даних з властивостями вбудованих типів, тобто з екземплярами, що мають семантику значень.

В роботах [16,18] показано, що ООП не має достатньо засобів для реалізації описаних можливостей. Це пов'язано з «проблемою бінарного метода» — неможливістю довизначати в породжених класах методи з сигнатурою $X.f(X)$, де X це сам клас, а також з відсутністю можливості визначати асоційовані типи. В сучасних об'єктно-орієнтованих мовах дані проблеми вирішуються за допомогою віртуальних типів [19,20], що, по суті, є певною формою параметричного поліморфізму на противагу підтиповому поліморфізму.

Існують два основні способи представлення узагальнених алгоритмів: параметричний поліморфізм [21] та макросна параметризація [16]. Для макросної параметризації характерно створення екземпляра алгоритму для кожного конкретного набору типових параметрів. Параметричний поліморфізм та макросна параметризація порівнювані за такими позиціями:

- окрема перевірка типів;
- окрема компіляція;
- ефективність;
- диспетчеризація за фактичними типами.

Окрема перевірка типів можлива при обох підходах. Зокрема, в C++ вже запропоновані методи перевірки типів самого шаблону, а не кінцевих екземплярів, що значно спростить пошук помилок. Внаслідок особливостей макросної параметризації окрема компіляція узагальнених функцій у ньому неможлива. В свою чергу, оскільки при параметричному поліморфізмі для всіх типів обробка єдинообразна, то неможлива диспетчеризація за типами.

Актуальною проблемою параметризованого програмування є підтримка мовами програмування обмежень на типові параметри для представлення концептів. Дані обмеження можна поділити на семантичні, власне предикати з типовими параметрами, та синтаксичні — відповідність операцій на типі деякій сигнатурі, асоційовані типи, узгодженість типів.

Наразі мовами підтримуються багато синтаксичних обмежень. Для представлення концептів та передачі відношення моделювання використовують параметризацію структур даних (C++, Haskell) та абстрактні типи даних (ML). Частина мов робить перевірку синтаксичних обмежень динамічно, частина, що є більш

бажаним, статично. В C++ історично шаблони та відповідно параметризовані функції не мали жодних обмежень на фактичні типові параметри. Механізмом перевірки служила перевірка екземплярів. В BGL [22] були запропоновані класи перевірки на відповідність типів концептам, що розширюють мову. Вбудована у мову перевірка відповідності концептам плануються в майбутніх стандартах мови C++.

Інша річ семантичні обмеження. Для їх задання можна використовувати мови специфікації: Tecton, Z, OBJ [23]. Також деякі мови з елементами контрактного програмування, наприклад, Eiffel, передбачають конструкції для формулювання специфікації функції в предикативній формі. Типовим способом перевірки семантичної специфікації є динамічна перевірка, коли компілятор автоматично додає код, що контролює дотримання обмежень протягом виконання програми. Іншим способом перевірки є формальна верифікація. В роботі [24] Мюссер запропонував систему формальної верифікації узагальнених алгоритмів на основі прувера Athena. Головна проблема при такому підході — складність формалізації мови програмування, в даному випадку C++.

Відомо, що, використовуючи властивості вбудованих типів та операцій над ними, компілятор може виконувати оптимізацію програми. В перспективі очікується, що подібна оптимізація буде можлива також на основі інформації про семантичні обмеження на типи даних, що визначаються користувачем.

Резюмуючи даний підрозділ можна заключити наступне. Параметризоване програмування додає до арсеналу програміста новий рівень абстракції. Цим покращується розуміння суті проблем що розв'язуються, та скорочується об'єм непотрібної, часто пов'язаної із помилками роботи, на зразок, копіювання та корегування коду. В цілому методологія є усталеною та широко підтримується багатьма мовами програмування. Вірогідно, з часом концепти стануть повсякденним явищем, та поняття параметризованого програмування перестануть виділяти спеціальним чином, розчинившись в загальному предметі програмування.

3 Реалізація категорій засобами

параметризованого програмування

В даному підрозділі розглянуті та порівняні особливості реалізації категоричних конструкцій різними мовами програмування, що мають дженерики. Під дженериками ми, як було сказано

раніше, розуміємо засоби мови програмування, що дозволяють використовувати методологію параметризованого програмування [16,25]. Параметризоване програмування запозичило ідеї формулювання алгоритмів в термінах таксономій концептів саме з абстрактної алгебри. Через це дженерики є зручним та природнім інструментом для реалізації алгебраїчних абстракцій. Теорія категорій є універсальною мовою моделювання, що має високий ступінь абстракції. Вона носить конструктивний характер, що виявляється у тому, що в переважній більшості доведень в явному вигляді наводиться конструкція шуканого об'єкту. Таким чином, теорія категорій наповнена узагальненими алгоритмами.

Метою цієї частини роботи є реалізація бібліотеки підтримки категоричних конструкцій на алгоритмічних мовах: функціонального типу (Haskell), об'єктно-орієнтованій з підтримкою обмеженого параметричного поліморфізму (C#) та мові з засобами макросного поліморфізму (C++). Така бібліотека може стати основою для повноцінної boost-бібліотеки для роботи із категоріями. Маючи таку реалізацію можна на конкретному прикладі порівняти, які має недоліки та переваги щодо зручності, обмежень та продуктивності та чи інша алгоритмічна мова. На відміну від [26], де дженерики відомих мов програмування порівнюються у загальному сенсі за синтаксисом та способом реалізації параметризованого програмування на основі маленьких іноді штучних прикладів, тут маємо справу з повноцінною системою.

За основу в роботі взята бібліотека представлення сутностей теорії категорій на програмній мові SML (її варіації *mosml*) побудована Д.Райдхардом [6]. Структурно вона складається з кількох модулів: *basic*, *cat*, *adjoints*, *functorcat*, *comma*, *semitopos*, *topos*. У кожному з них розроблені параметризовані типи, що описують певні категорні поняття та узагальнені алгоритми, що є конструкціями доведення певних категоричних властивостей. Наприклад, в головному модулі *cat* вводяться типи категорії та функторів, дуальність, діаграм та граничних конусів. Центральним алгоритмом модулю є алгоритм побудови границь в категорії з визначеними термінальним об'єктом, декартовим добутком та порівнювачем, що є відомою теоремою про скінченну повноту відповідної категорії [27]. Основною моделлю, на якій демонструється робота бібліотеки, є деяка підкатегорія категорії скінченних множин *FinSet*.

В своїй основі бібліотека ґрунтується на принципах функціонального програмування: параметричному поліморфізмі та інтенсивному використанні функцій, зокрема можливостей перетворення функцій. Концепти як такі в ній не виділені, використовуються параметризовані типи та звичайні конструкції утворення композиційних типів такі як *n*-ки. Моделювання відбувається на основі конкретизації типів та функцій.

Переніс коду на Haskell не становить великих проблем[28] адже ця мова є доволі близької до SML, зокрема у підтримці параметричного поліморфізму. Відмінності полягають у деяких синтаксичних особливостях мов, а також способах оформлення модулів. Завдяки своїй популярності Haskell має велику базу бібліотек. Це дозволяє не переносити код підтримки операцій із множинами, а скористатися стандартними модулями експортованими в *Prelude*, а також *Data.Set*, *Data.Map*. В Haskell відсутній окремий механізм виключень, що є в SML та використовується в бібліотеці. Проте виключення можна моделювати за допомогою спеціальних розширень або використовуючи спеціальне «найменше» значення $\perp$, що належить усім типам (на основі якого зокрема сигналізується про помилки). Певною незручністю є відкладання обчислень та відсутність традиційних засобів відладки. Це ускладнює оцінку продуктивності та пошук помилок (на пошук простої описки, що полягала в хибному вказуванні координат у одній з ін'єкцій кодобутку *FinSet*, пішло кілька днів). Наступним кроком вдосконалення реалізації може бути переформулювання концептів у вигляді класів типів Haskell подібно до того як зроблено в [29].

Для реалізації коду на C++ використовується зовнішня адаптація (подібно до BGL). При перенесенні бібліотеки виникають дві проблеми. Перша пов'язана із значним використанням в вихідному коді функцій, зокрема їх композицій. Простіший приклад: для реалізації моделі *FinSet* необхідно обчислювати композицію функцій. В C++ для розв'язання проблеми використовуються функціональні об'єкти, тобто функтори. Із ними можна виконувати маніпуляції аналогічні тим, що виконується з функціями в SML. Друга проблема виходить з обмежень *set*-контейнерів, що грають роль об'єктів *FinSet*. Вони мають бути параметризовані за певним типом. Вимога скінченної повноти та коповноти *FinSet* призводить до рекурсивності цього типу-параметру. Безпосередня підтримка рекурсивних

типів даних в C++ відсутня. Використання динамічних структур становить клопіт і пов'язане із помилками. Проте цього вдається уникнути скориставшись контейнером *boost-variant*, що дозволяє безпечно будувати варіантне утворення з різних типів з можливістю рекурсивного визначення.

Базовими дженериками в C# є параметризовані інтерфейси. Для моделювання концептів інтерфейс реалізується відповідним класом. Перевагою C# є підтримка функцій як об'єктів першого класу. Втім залишається проблема рекурсивних типів даних, які треба реалізовувати окремо.

4 Висновки

Як було сказано раніше перспективою даної роботи можна вважати розробку стандартизованої *boost*-бібліотеки для роботи з категоріями. Для такого переходу необхідно внести ряд удосконалень. В вихідному варіанті бібліотека спирається на власну спрощену реалізацію графової структури категорії та графів

для представлення діаграм. Її доцільно замінити на стандартні концепти, що надає BGL. З метою зменшення сутностей можна виразити (ко-)границі діаграм через (ко-)границі функторів. Для цього слід додати моделі скінченних категорій. Ефективність деяких алгоритмів можна покращити переробивши їх з рекурсивних на ітеративні. В цьому велику роль грають ітератори, якими має бути поповнений код.

Найбільш привабливим вдосконаленням даної системи є оснащення її апаратом категорної логіки за допомогою додаткового рівня трансляції, що виконуватиме перетворення категорних виразів на внутрішню мову і навпаки. Це дозволить підключати прувери та використовувати методи автоматизації пошуку доведень, значно розширивши можливості системи.

Список використаних джерел

1. Kent R. The IFF Foundation for Ontological Knowledge Organization / R. Kent // *Cataloging & Classification Quarterly* – 2004. – V. 37, no. 1–2. – Pp. 187–203.
2. Kent R. A KIF Formalization for the IFF Category Theory Ontology / R. Kent // *Proceedings of the IJCAI-2001 Workshop on the IEEE Standard Upper Ontology* – Seattle, Washington, USA, 2001.
3. Glazunov N. Toward a Category Theory Design of Ontological Knowledge Bases / N. Glazunov. – Cornell University Library, arXiv. – 2009. – 10pp. – (E-prints / arXiv:0906.1694).
4. Goguen J. Introducing OBJ : Technical Report / J. Goguen; SRI International, Computer Science Laboratory. – SRI-CSL-92-03. – 1992. – 107pp.
5. Nordstrom B. Programming in Martin-Lof's Type Theory: An Introduction / B. Nordstrom, K. Petersson, J. Smith – Oxford University Press, 1990. – 211pp.
6. Rydeheard D. Computational Category Theory / D. Rydeheard, R. Burstall. – New York, London: Prentice Hall, 1988. – 257 pp.
7. Weber T. Program Transformations in Nuprl : M.Sc. Thesis / T. Weber; University of Wyoming, Laramie. – 2002. – 122pp.
8. Wolfengagen V. Combinatory logic in programming. Computations with objects through examples and exercises [2-nd ed.] / V. Wolfengagen. – M.: "Center JurlInfoR" Ltd., 2003. – 337 pp.
9. Zeng M. An Implementation of Charity : M.Sc. Thesis / M. Zeng; The University of Calgary. – 2003. – 120pp.
10. Hagino T. A Categorical Programming Language : Ph.D. Thesis / T. Hagino; University of Edinburgh. – 1987. – 140pp.
11. Gibbons J. Datatype-generic programming / J. Gibbons // *Lecture Notes in Computer Science*. – 2007. – vol. 4719. – Pp. 1–71.
12. Montanari U. Algebraic Approaches to Graph Transformation, Part I: Basic Concepts and Double Pushout Approach : Technical Report / U. Montanari, H. Ehrig, M. Loewe; University of Pisa. – TR-96-17. – 1996. – 80pp.
13. Loewe M. Refactoring information systems – a formal framework / M. Loewe, H. Koenig, C. Schulz // *Proceedings WMSCI-2006*. – 2006. – Volume 1. – Pp. 75–80.
14. Taentzer G. AGG: A Graph Transformation Environment for Modeling and Validation of Software / G. Taentzer // *Lecture Notes in Computer Science* – 2004. – vol. 3062. – Pp. 446–453.
15. Stepanov A. Elements of programming / Alexander Stepanov, Paul McJones. – Addison-Wesley, 2009 – 288 pp.

16. Siek J. A language for generic programming: Ph.D. Thesis / J. Siek; Indiana University. – 2005. — 260 pp.
17. Generic Programming / [Mehdi Jazayeri, R. Loos, David Musser, and Alexander Stepanov] // Report of the Dagstuhl Seminar on Generic Programming. – Dagstuhl Schloss, Germany, 1998. — <http://www.cs.rpi.edu/~musser/gp/dagstuhl/gpdag.ps>.
18. On binary methods / [Kim Bruce, Luca Cardelli, et al.] // Theory and practice of object systems. – 1995. — Vol. 1, no.3. — Pp. 221-242.
19. Ernst E. Family polymorphism / E. Ernst // Lecture Notes in Computer Science. – 2001. – Vol. 2072. – Pp. 303-326.
20. Madsen O.L. Virtual classes: a powerful mechanism in object-oriented programming / O.L. Madsen, B. Moller-Pedersen // Conference proceedings on Object-oriented programming systems, languages and applications (OOPSLA 89). – New Orleans, US, 1989. – Pp. 397-406.
21. Strachey C. Fundamental concepts in programming languages / C. Strachey // Lecture Notes, International Summer School in Computer Programming. – Copenhagen, August 1967. – Pp. 1-49.
22. Siek J. Concept Checking: Binding Parametric Polymorphism in C++ / Jeremy Siek, Andrew Lumsdaine // First Workshop on C++ Template Programming. – Erfurt, Germany, 2000. — <http://www.oonumerics.org/tmpw00/siek.pdf>.
23. Kapur D. Tecton: a framework for specifying and verifying generic system components: Technical Report / D. Kapur, D. Musser / Department of Computer Science, Rensselaer Polytechnic Institute. – RPI-92-20. – New York, July 1992.
24. Musser D. Generic programming and formal methods / D. Musser // Workshop on The Verification Grand Challenge. – Menlo Park, CA, 2005. — <http://www.csl.sri.com/users/shankar/VGC05/>.
25. Ченцов А. Параметризованное программирование: современное состояние и перспективы / А. Ченцов // Тези доп. міжн. конф. "Теоретичні та прикладні аспекти побудови програмних систем" (КНУ, 8-10 грудня 2009). – Київ, 2009. – С.159-164.
26. An extended comparative study of language support for generic programming / R. Garcia, J. Jaarvi, A. Lumsdaine et al. // Journal of Functional Programming. – 2005. – Pp. 115-134.
27. Маклейн С. Категории для работающего математика / С. Маклейн. – М.: ФИЗМАТЛИТ, 2004. – 352 с.
28. Ченцов О. Реалізація конструкцій абстрактної алгебри засобами параметризованого програмування / О. Ченцов // Тези доп. міжн. конф. "Теоретичні та прикладні аспекти побудови програмних систем" (КНУ, 4-8 жовтня 2010). – Київ, 2010. – С.477-479.
29. Sozeau M. First-class type classes / M. Sozeau, N. Oury // Proceedings of the 21st International Conference on Theorem Proving in Higher Order Logics. – 2008. – Pp. 278-293.

Надійшла до редколегії 15.12.10

УДК 681.3.062

Є.А.Чичкарьов, к.т.н, О.В. Кривенко, к.т.н., доц.

Розробка сервера обчислень і web-інтерфейсу для віддаленої роботи систем комп'ютерної математики в середовищі пакету Moodle

Проаналізовано існуючі рішення з розробки web-інтерфейсу до обчислювальних пакетів з відкритим кодом, а також переваги і недоліки аналогічних пропрієтарних рішень для пакетів Maple, Matlab, MathCad. Представлені результати розробки і випробування обчислювального сервера для математичного пакету Maxima, розрахованого на роботу спільно з системою дистанційного навчання Moodle.

Ключові слова: сервер обчислень, система комп'ютерної математики, дистанційне навчання, web-інтерфейс.

E-mail: influence@meta.ua

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Сучасне дистанційне навчання (ДН) будується на використанні наступних основних елементів:

- середовища передачі інформації (пошта, телебачення, радіо, інформаційні комунікаційні мережі),
- методів, залежних від технічного середовища обміну інформацією.

Технології дистанційного навчання дозволяють:

- знизити витрати на проведення навчання (не потрібно витрат на оренду приміщень, поїздок до місця навчання як учнів, так і викладачів);
- проводити навчання великої кількості осіб;
- підвищити якість навчання за рахунок застосування сучасних засобів, об'ємних електронних бібліотек і т.д.
- створити єдине освітнє середовище (особливо актуально для корпоративного навчання).

Використання дистанційного навчання для викладання різних розділів математики та навчальних дисциплін, що включають елементи математичного моделювання, дозволяє посилити їх прикладну і практичну спрямованість і

E.A. Chichkarev, Phd, O.V. Krivenko, Phd, docent

Development of Calculation Server and Web-based Interface for Remote Operation of Computer mathematics in the Environment of Moodle

The existing solutions to develop web-interface to open-source calculating packages, as well as the strengths and weaknesses of similar proprietary solutions for packages Maple, Matlab, MathCad, are analyzed. The results of development and testing of a calculation server for a mathematical Maxima package, which designed to work together with the system of distance learning Moodle, are presented.

Key Words: calculation server, system of computer mathematics, distance learning, web-interface.

створює умови для реалізації індивідуального підходу на якісно новому рівні.

В даний час існує велика кількість web-інтерфейсів до різних інформаційних систем, в т.ч. і математичним пакетам. Більшість комерційних пакетів, таких як MatLab, Maple, MathCad, Mathematica, мають інтерфейси, що дозволяють використовувати їх функціональність віддалено. Інший характерний приклад - Mathcad Calculation Server, що дозволяє працювати з MathCad-документами в мережі. Аналогічні розширення існують і для Maple (MapleNet) [1] або Mathematica (webMathematica).

З появою системи дистанційного навчання Moodle все більше проектів організації ДН стали орієнтуватися на дану систему як на функціональну оболонку. Принципи функціонування системи Moodle дозволили з достатньою легкістю вести розробку, орієнтуючись спочатку на використання в якості базових функцій API даної системи як базового середовища для інтеграції. Зокрема, з'явилися проекти, які є надбудовами над системою дистанційного навчання Moodle.

Web-інтерфейс до середовища MatLab дозволив розробникам створити проект, який був

модулем фільтра в Moodle і використовував MWS як шлюз до обчислювальних здібностей MatLab. Даний фільтр дозволяє вбудовувати команди MatLab в документи і виконувати їх перед передачею на сторону клієнта (в браузер користувача) [2, 3]. Однак слід зазначити, що компанія MathWorks відмовилася від свого проекту MWS, в результаті чого він став недоступний для останніх версій MatLab.

Існують проекти, засновані на використанні системи ДН Moodle та пакету MapleNET [4]. Дане рішення було розроблено на основі вже відомій ідеї «вбудовування» команд в документи курсів, лабораторних робіт і тестів, обчислення їх з наступним відображенням результатів.

Технологічно засоби організації віддаленого доступу не однакові, але характерним прикладом може бути webMathematica, заснована на технологіях сервлетів та JSP.

У даній області існують рішення, що базуються виключно на opensource-програмних продуктах, а саме – системі ДН Moodle і математичному пакеті Maxima (приклад – пакет STACK – System for Teaching and Assessment using a Computer algebra Kernel [5]). Для обчислення виразів і побудови графіків використовується обчислювальне ядро системи комп'ютерної математики (СКМ) Maxima [5, 6]. Для відображення результатів використовується бібліотека MathML, якщо вони представлені в текстовому вигляді і програма Gnuplot для роботи із зображеннями і перетворенням їх в файли формату «png». Саме середовище STACK написано з використанням скриптової мови PHP і використовує для своєї роботи СУБД MySQL, завдяки чому дана система успішно працює з платформами Linux, Windows, Mac OS X.

Можливості видаленої роботи з обчислювальними пакетами забезпечує і пакет Sage (<http://sagemath.org/>). За допомогою Sage-notebook можна видалено працювати з цілим рядом математичних пакетів, що входять до складу Sage, а також публікувати в Інтернет інтерактивні документи.

Враховуючи технологію реалізації СДН Moodle (пакет розроблений на php), при запуску обчислювального ядра того або іншого пакету на сервері важливою проблемою є фільтрація команд користувача для блокування потенційно небезпечних функцій або команд конкретного пакету, що вимагають доступу до файлової системи сервера. Крім того, у ряді відкритих реалізацій веб-сервера-інтерфейсу до математичних пакетів для звернення до ядра

пакету використовуються потенційно небезпечні функції php – `exec()`, `passthru()` і тому подібне. Нарешті, серйозною проблемою може бути масштабування проекту із зростанням об'єму використання СДН.

У даній роботі проведено дослідження особливостей роботи видаленого доступу до СКМ Maxima з документів, розміщених в СДН Moodle.

Для проведення досліджень було створено середовище у вигляді сегменту мережі. Для регулювання параметрів серверу була використана його віртуалізація засобами програмного забезпечення VMware Workstation. В якості серверної операційної системи була обрана Windows2003 Server. На дану систему був встановлений web-сервер Apache і СУБД MySQL. Віртуальне середовище з'єднувалося з операційною системою-хостером шляхом емуляції мережі з коефіцієнтом втрати пакетів, рівним 0,001 %. Операційна система-хостер так само виконувала роль моста для з'єднання (вже через фізичне середовище) з іншими робочими станціями (на момент проведення експерименту їх було чотири).

Завантаження оперативної пам'яті сервера при виконанні завдань різної складності показана в табл. 1.

Таблиця 1.

Обсяг оперативної пам'яті, який займають програми

Процес	Обсяг оперативної пам'яті (Мб)
Httpd.exe (ядро Apache)	15,45
Httpd.exe (MOODLE)	43,08
Httpd.exe (PHP Server Side script)	20,432
Mysqld-max-nt.exe (СУБД MySQL)	7,144

З даних, наведених у табл. 1 видно, що компоненти системи без навантаження на них займають простір в оперативній пам'яті сервера рівний 87 Мб.

Для дослідження параметрів навантаження сервера при різних типах завдань, виконуваних пакетом MAXIMA було виділено 5 завдань різного ступеня складності:

- складання двох чисел;
- обчислення не табличного інтеграла;
- обчислення суми ряду;
- розрахунок рішення системи двох звичайних диференціальних рівнянь;

- розрахунок рішення системи чотирьох звичайних диференціальних рівнянь (ДР).
Результати експериментів наведені в табл. 2.

Таблиця 2.

Час обчислення і обсяг оперативної пам'яті для завдань різної складності

Складність задачі	Обсяг оперативної пам'яті (Мб)	Час відклику (сек)
складання	2,014	0,555
інтеграл	29,673	1,754
сума ряду	2,517	0,727
система ДР (2)	46,862	3,709
система ДР (4)	50,231	4,104

Для дослідження роботи сервера з різним навантаженням були створені два віртуальні сервера (512 Мб і 1024 Мб ОЗП відповідно; однопроцесорна архітектура з тактовою частотою 1,86 GHz).

Звернення до обчислювального ядра MAXIMA здійснювалося з віддаленого терміналу за допомогою серверного скрипта на PHP. У результаті проведення обчислювального експерименту встановлено, що незалежно від кількості паралельних процесів MAXIMA обсяг оперативної пам'яті, займаний кожним з них, змінюється незначно. Незначні зміни обумовлені впливом на систему фонових задач операційної системи, в рамках якої проводився експеримент. Інтервал між запусками примірників не перевищував 0,023 секунди, що істотно менше часу обчислення тестової задачі (близько 1,5 с для одиничного примірника MAXIMA).

На рис. 1 показана залежність часу відгуку сервера від кількості одночасно запущених екземплярів. Зростання до кінця кривої відгуку обумовлено тим, що час відгуку сервера різко збільшується при граничних значеннях величин ресурсів, що виділяються системі на обробку завдань.

Для сервера з двоядерною архітектурою (1024 Мб ОЗП, тактова частота 2GHz) різке зростання часу відгуку в міру зростання паралельно запущених процесів не було відзначено.

Запуск MAXIMA на сервері здійснювався

кількома способами:

- з використанням функції PHP passthru ();
- з використанням сокетів.

Останній варіант є кращим, тому що дозволяє більш зручно контролювати навантаження сервера без різкого зростання часу відгуку всіх запущених процесів.

Для організації віртуальних та практичних робіт в СДН Moodle елементи графічного інтерфейсу користувача реалізовані з використанням технології Ajax. У систему MOODLE включені кілька бібліотек, необхідних для створення інтерфейсу і ряду модулів:

- офіційна бібліотека JavaScript для Moodle - YUI (розробка Yahoo Developer Network; представляє собою набір утиліт і елементів управління, написаних з використанням JavaScript і CSS, для створення інтерфейсу web-додатків з використанням таких методів, як DOM-сценарій, DHTML і AJAX) [7];
- власна бібліотека JavaScript коду (lib / JavaScript-static.js);
- редактор TinyMCE та його бібліотеки для роботи з HTML-документами всередині середовища MOODLE.

Аналогічний підхід до організації web-інтерфейсу і виконання обчислень у документах MOODLE може використовуватися і для інших СКМ (Octave, Scilab і т.п.).

Однак, як показали результати дослідження роботи віртуального сервера, до віддаленого запуску обчислювальних програм слід ставитися з обережністю, тому що успішне виконання розрахунків вимагає використання досить потужного сервера, оснащеного достатнім обсягом пам'яті і продуктивним багатоядерним процесором.

На даний час розробляється комплект документації для організації навчального процесу на базі СКМ (методичні вказівки з різних дисциплін, керівництво з лабораторного практикуму та ін.).

Дистанційне застосування СКМ у викладанні математичних курсів сприяє інтересу студентів до досліджуваного за їх допомогою матеріалу.

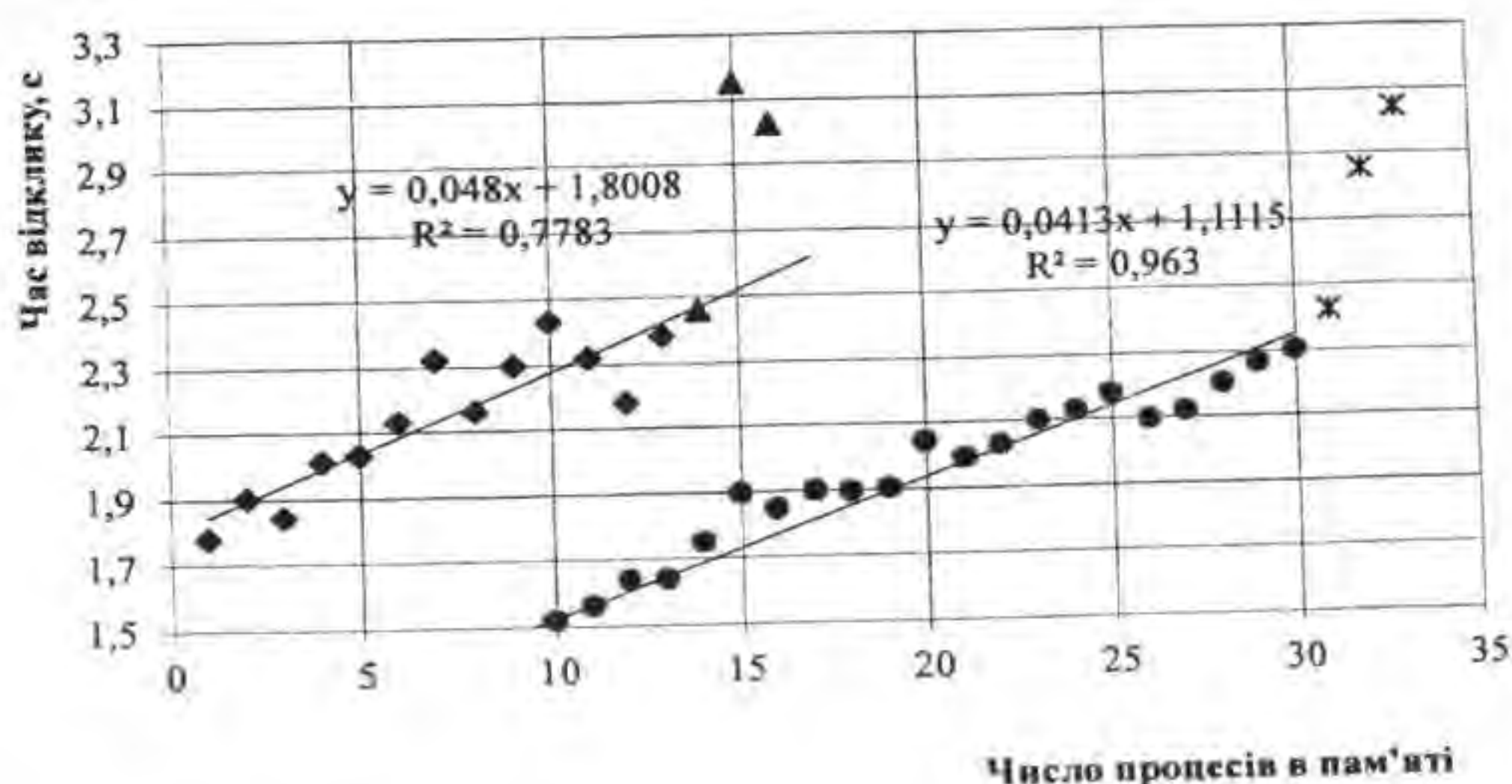


Рис. 1. Графік залежності часу відклику програми від кількості одночасно працюючих її екземплярів

Високий рівень візуалізації обчислень, дозволяє легше зрозуміти математичну (або фізичну або технічну - при вивченні питань моделювання) сутність досліджуваних методів та алгоритмів. Багатьом учням цікаво вивчення саме цих систем, які є великими інтелектуальними програмними продуктами. Крім того, використання СКМ дозволяє зробити процес навчання більш швидким і насиченим, краще

розмістити в рамках навчального графіка передбачений програмою обсяг матеріалів, збагачувати курси, які читаються, новими методами і поняттями.

Список використаних джерел

1. Дьяконов В.П. Maple 10 в математике, физике и образовании /В.П. Дьяконов. - Москва: Солон-Пресс, 2006. - 720 с.
2. Котельников И.А. MATLAB Web Server: вычисления в Интернете /Котельников И.А., Черкасский В.С./ Exponenta Pro. Математика в приложениях. №1(5). 2004. - С.4-11.
3. Котельников И.А. Разработка и использование моделирующих программ для дистанционного и самостоятельного

- обучения с использованием Matlab Web Server/ Котельников И.А., Матвеев А.Н., Черкасский В.С. // Тезисы докл. международной науч.-метод. конф. «Новые информационные технологии в университетском образовании» (Новосибирск, 23-24 сентября 2003). - Новосибирск: СибГУТИ, 2003. - С. 92-94.
4. http://www.colum.edu/Project_AIM.
5. <http://stack.asu.edu/>.
6. <http://maxima.sourceforge.net/>.
7. <http://dev.yahoo.com>.

Надійшла до редколегії 15.12.10

УДК 004.4

Шкуліпа І.Ю., аспірант,
Погорілий С.Д.*, д.т.н., проф.

Система автоматизованого параметричного проектування алгоритмів

Запропоновано концепцію створення системи автоматизованого параметричного проектування паралельних алгоритмів. Описано функціональні складові системи – підсистема трансформації САА-М-схем алгоритмів, підсистема синтезу програм, асоційованих з алгоритмом, підсистема виконання та збору статистичних даних, підсистема генерації САА-М-схем на основі UML-діаграм та підсистема моделювання роботи схем алгоритмів. Наведено підходи до параметризації системи та інструментальні засоби її створення.

Ключові слова: алгоритм, САА-М, формалізація, паралельне програмування, паралелізм.

Shkulipa I.Yu., PhD student,
Pogoriliy S.D.*, dr. sci. tech., Prof.

Automated parametric algorithm design system

The concepts of automated parametric parallel algorithm designing system are proposed. Functional components of the system are described – SAA-M-schemes transformation subsystem, program synthesis subsystem, execution and statistic data collecting subsystem, subsystem of SAA-M-schemes generation based on UML-diagrams and subsystem of algorithms schemes modeling. The approaches for parameterization and instruments for practical realization are proposed.

Keywords: algorithm, SAA-M, formalization, parallel programming, parallelism.

E-mail: *Igor.Shkulipa@gmail.com, **SDP@rpd.univ.kiev.ua
Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ

Сучасні розробки в галузі обчислювальних систем ті їх орієнтація на паралельне виконання алгоритмів ставлять задачу ефективного проектування таких алгоритмів. Тому створення засобів розробки, перетворення та дослідження алгоритмів на етапі проектування є важливою темою досліджень сьогодення.

Актуальною науковою задачею досліджень є автоматизація процесу створення паралельних алгоритмів на основі послідовних та генерація відповідного програмного забезпечення, що реалізує заданий алгоритм.

Задача створення системи автоматизованого параметричного проектування алгоритмів САПП [1] полягає у формуванні наукових засад для створення програмного комплексу, що дозволяв би у автоматизованому режимі виконувати проектування, аналіз алгоритмів та генерацію

відповідних паралельних програм різними мовами програмування, а також проводити моделювання їх роботи та збір результатів.

Опис системи

Створення САПП можна розділити на три основні задачі автоматизації процесів:

- створення паралельних версій алгоритмів
- створення відповідних програм, з використанням різних мов програмування та різних парадигм паралельного програмування
- виконання програм та вибору найкращого алгоритму за заданими критеріями;

і дві додаткові задачі, які виникли в процесі створення САПП:

- Зручність проектування вихідного алгоритму. Інтеграція процесу проектування

алгоритму з відомими зручними засобами алгоритмічного проектування (UML-діаграми)

автоматизованого перетворювача алгоритмів та отримання їх паралельних версій.

• Моделювання роботи отриманих модифікованих алгоритмів до виконання на реальних програмно-апаратних платформах. Для того щоб не використовувати ресурси кластера лише для моделювання.

З метою формалізованого опису алгоритмів на етапі алгоритмічного проектування пропонується використання математичного апарату модифікованих систем алгоритмічних алгебр (САА-М) В.М. Глушкова [2,3]. САА-М допускають подання будь-якого алгоритму у аналітичному вигляді – у вигляді САА-М-схеми. В свою чергу САА-М-схеми можуть бути формально модифіковані шляхом застосування до них еквівалентних перетворень, що дає зручну та потужну основу для створення

Використання цих засобів для алгоритмів є можливим завдяки тотожним трансформаціям, які допомагають спростити схему і дають можливість проводити поглиблену подальшу її трансформацію. Це особливо важливо при створенні нових схем на базі вже існуючих. Поєднання цих засобів – це потужний механізм для встановлення зв'язків між різними алгоритмами, а також для конструювання нових алгоритмів, більш ефективних з тих чи інших критеріїв.

Організаційна структура САПП має вигляд, представлений на рис.1.



Рис.1.

За допомогою модуля проектування САА-схем (або вручну в аналітичному вигляді) проектується вихідна САА-схема алгоритму. Далі вона подається на вхід перетворювача САА-схем, який, застосовуючи правила перетворення схем у кількості та послідовності, заданій користувачем, дає на виході низку модифікованих САА-М-схем, з яких можна, або згенерувати відповідні програми, для подальшого їх виконання на кластері та збору результатів роботи, або згенерувати відповідні мережі Петрі, для моделювання їх роботи та отримання

відносних часових характеристик без завантаження реальних апаратно-програмних платформ.

Підсистема генерації САА-М-схем на основі UML-діаграм

Модуль генерації САА-М-схем на основі UML-діаграм виконує функцію ланки етапу проектування алгоритму, що зв'язує зручний та наочний засіб проектування алгоритмів у вигляді UML-діаграм та формування САА-схеми алгоритму.

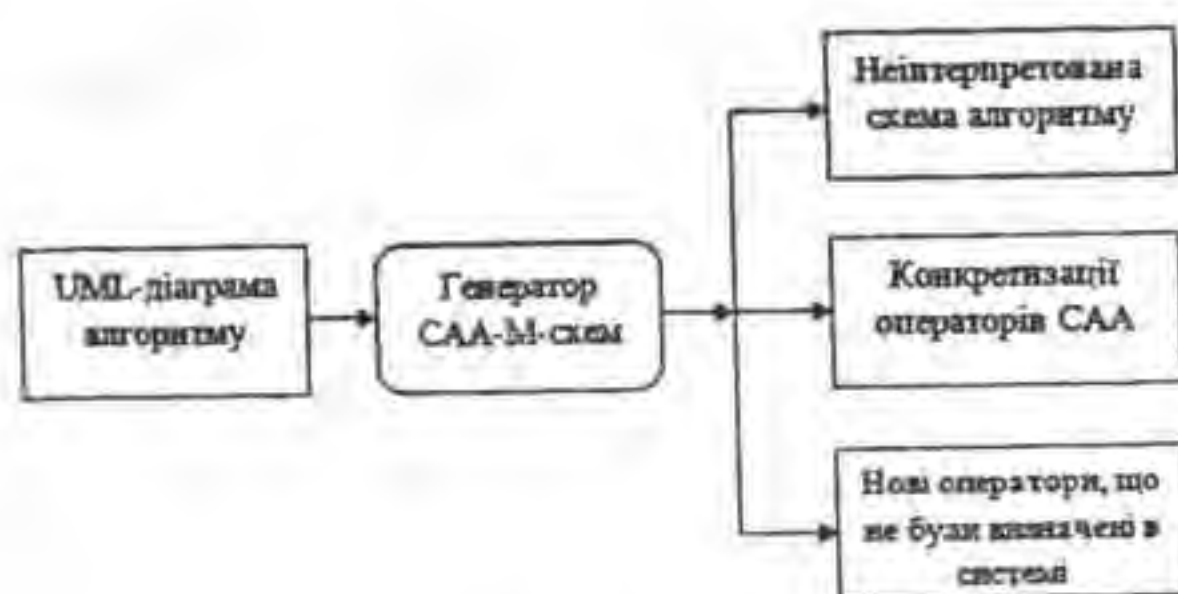


Рис.2.

Рис.2. ілюструє призначення модуля генерації САА-схеми алгоритму на основі UML-діаграми. На вхід модуля подається досліджуваний алгоритм у вигляді UML-діаграми, а на виході отримується САА-схема цього алгоритму та додаткові співставлення операторів та умов, що використовуються у схемі з їх реалізаціями мовами програмування, що необхідно далі у процесі генерації програмного коду.

Підсистема перетворення САА-М-схем алгоритмів

Підсистема перетворення схем алгоритмів [4] – це функціональна складова САПП, що призначена для перетворення та оптимізації вхідного алгоритму з метою формування нової або паралельної версії.

Аналітичний вигляд алгоритму базується на його представленні у вигляді формалізованої

САА-М-схеми. У термінах САА-М еквівалентні перетворення – це набір теорем і аксіом, що описують правила перетворення частин алгоритмічної схеми у нові операторні конструкції.

Алгоритм, що застосовується в системі, є ітераційним і схематично являє собою наступне:

- Перевіряється коректність запису вхідної схеми, відповідно до описаних вище правил
- До вихідної схеми застосовуються вибрані правила перетворень
- До ряду схем, отриманих після попереднього кроку, застосовуються наступні за списком правила перетворень
- Крок 2 повторюється для всіх схем кроку 1 і так далі для результуючих схем кожного кроку

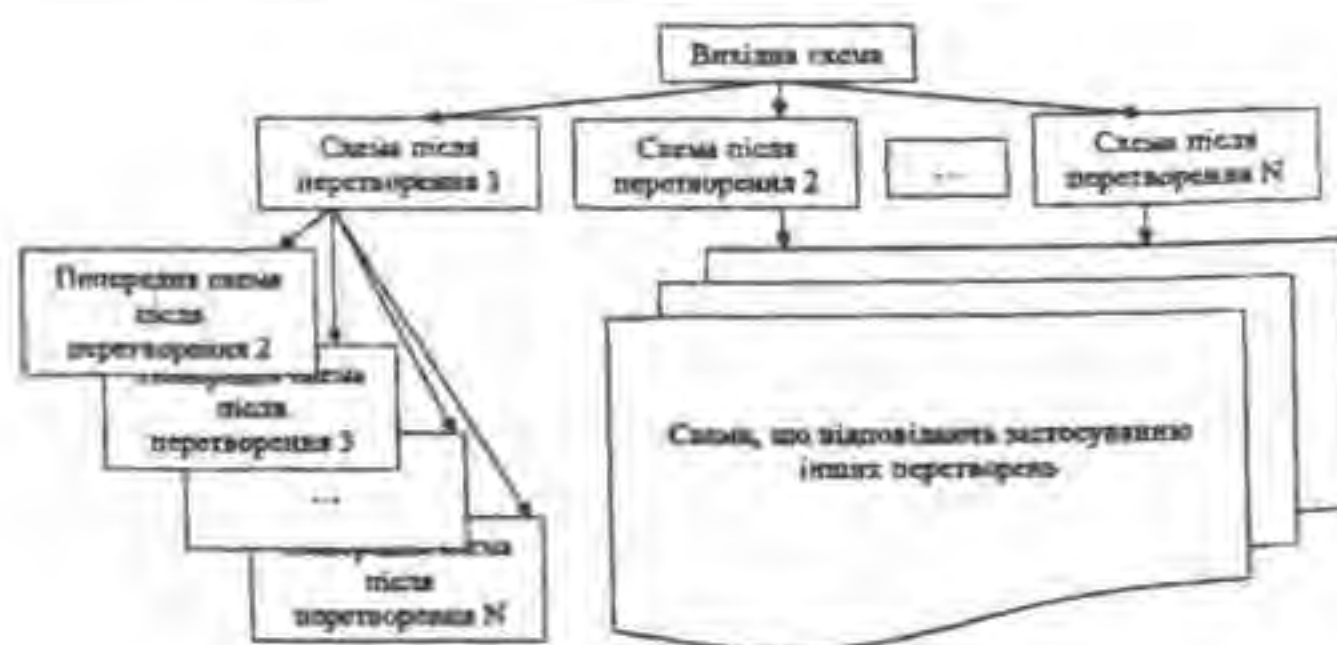


Рис. 3.

Результатом роботи підсистеми перетворення буде ряд схем алгоритму (рис.2.), які були отримані шляхом послідовного

застосування правил еквівалентних перетворень до вихідної схемою алгоритму. Таким чином, кількість результуючих схем алгоритмів буде дорівнювати

$$S = N^2 - T + 1,$$

де $T = 0..N^2$ - кількість еквівалентних схем, яка обумовлюється тим фактом, що не кожне з перетворень може бути застосовано до вхідної схеми алгоритму.

Підсистема генерації коду

Однією з основних компонент САПІ є генератор коду програм. Задача генератора програмного коду полягає у синтезі за САА-М-

схемами паралельних алгоритмів програмного коду **цільовими мовами програмування**. Оскільки в описаній системі нотація алгоритмів заснована на їхньому уявленні у вигляді формалізованих САА-М-схем, то створення такого генератора коду фактично зводиться до побудови компілятора, що перетворює САА-М-схему в програмний код цільовими мовами програмування.

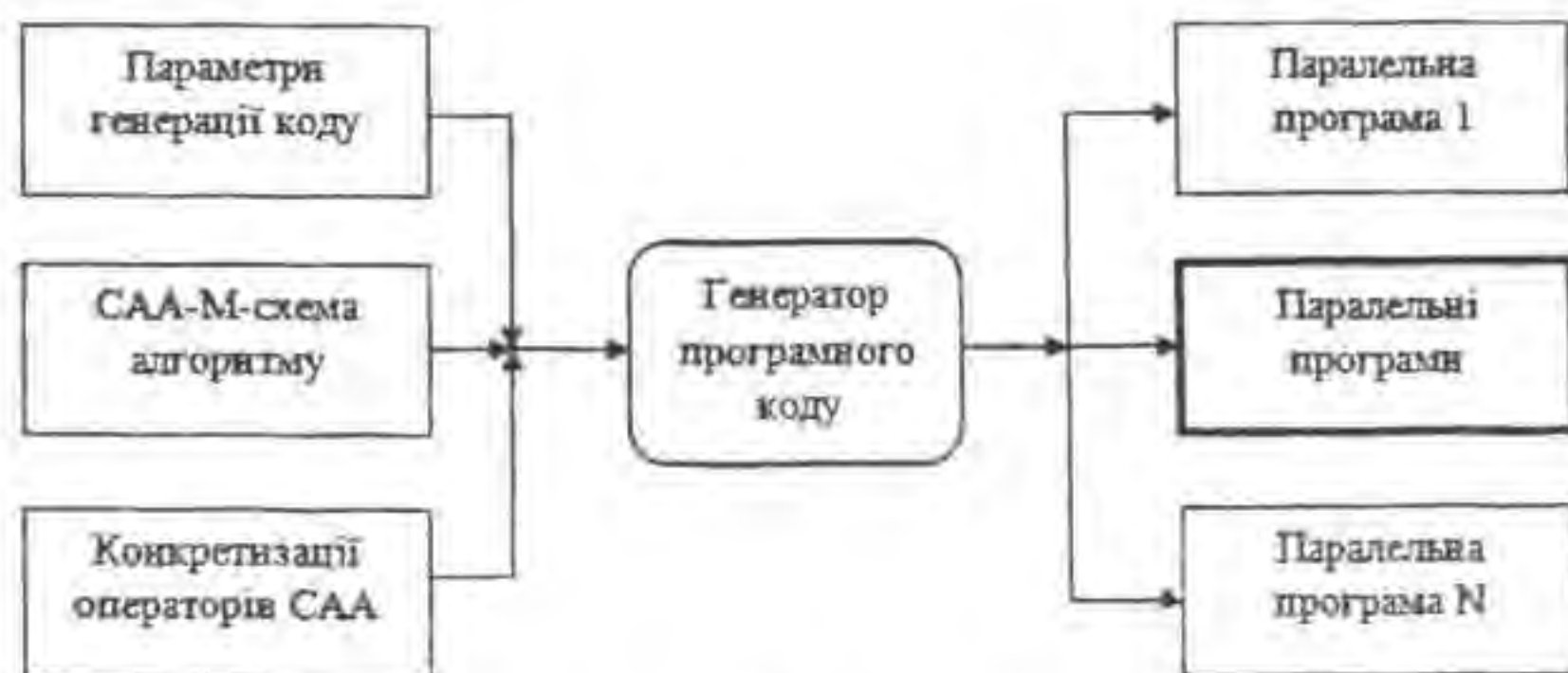


Рис.4.

На вхід генератора програмного коду подається не інтерпретована САА-М-схема алгоритму, а на виході отримується низка паралельних програм, що реалізуються заданий алгоритм. Для коректної генерації програмного коду генератору також необхідно мати конкретизації операторів САА, що використовуються в схемі та параметри генерації коду, такі як вибір цільової мови програмування, парадигми паралельного програмування, програмної платформи обчислювальної системи та ін.

Програмна реалізація генератора коду здійснює синтез програмних реалізацій алгоритмів мовами програмування C з застосуванням процес-орієнтованої, потік-орієнтованої та MPI - орієнтованої парадигм паралельного програмування, а також мовою програмування Java з застосуванням парадигми Java Multithreading.

Підсистема виконання та збору статистичних даних

Підсистема виконання та збору статистичних даних САПІ призначена для компіляції файлів програм, записаних цільовою

мовою, виконання та збору параметрів роботи програм, що були отримані після компіляції схем алгоритмів у програмний код. Цей компонент системи включає в себе також інтерфейс користувача, який призначений для слідкування за процесом перетворення схем та участі користувача у процесі створення програмної реалізації алгоритму.

Підсистема виконання та збору статистичних даних виконує генерацію сценаріїв для виконання програмних реалізацій алгоритмів на кластерах Київського національного університету імені Тараса Шевченка для програмних платформ Microsoft Compute Cluster Server [5] та Linux [6].

Підсистема моделювання роботи схем

Ця підсистема складається з двох компонентів: модуля генерації мережі Петрі на основі САА-М-схеми та середовища для моделювання її роботи.

На рис.5. зображено цикл моделювання роботи САА-М-схеми алгоритму. За допомогою генератора мереж Петрі створюється файл з відповідною мережею, потім, за допомогою середовища моделювання мереж Петрі

проводиться відповідне моделювання роботи мережі та отримуються часові характеристики роботи алгоритму у відносних одиницях, в залежності від кількості паралельних гілок та

кількості процесорів у апаратній архітектурі обчислювальної системи.

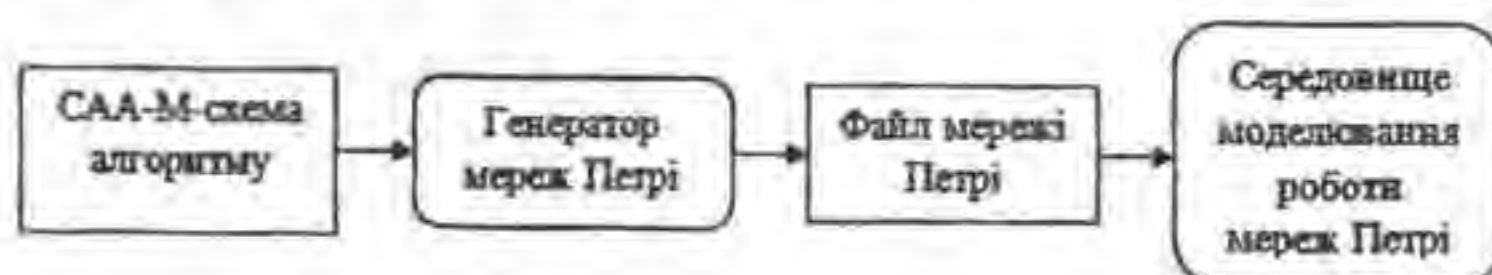


Рис.5.

Інструментальні засоби розробки

Реалізація системи САПП виконана на платформі Microsoft .Net Framework мовою програмування C#. Як система керування базами даних використовується Microsoft Office Access 2007. Платформа Microsoft .Net має вбудовані потужні засоби для обробки регулярних виразів, засоби для побудови дерев та роботи з базами даних.

Тестування САПП та її експериментальні випробування проводяться на кластерах Київського національного університету імені Тараса Шевченка під управлінням платформ Linux та Microsoft Compute Cluster Server.

В якості середовища для моделювання мереж Петрі використовується CPN Tools.

Список використаних джерел

1. Погорілий С.Д. Концепция создания автоматизированной параметрической системы проектирования параллельных алгоритмов и их программных реализаций / Погорілий С.Д., Шкуліпа І.Ю. // Кибернетика и системный анализ. -2009. №6. -стр. 118-124.

2. Е.Л.Ющенко Многоуровневое структурное проектирование программ / Е.Л.Ющенко, Г.Е.Цейтлин, В.П.Грицай, Т.К.Терзян // М. 1989, 208 стр.

3. К.Л. Ющенко Алгоритмічні алгебри / К.Л. Ющенко, С.В. Суржко, Г.О. Цейтлін, А.І. Шевченко // К. – 1997, 480 стор.

4. Шкуліпа І.Ю. Методика автоматизованої трансформації схем алгоритмів / Шкуліпа І.Ю.,

Висновки

Апарат САА В. М. Глушкова, як засіб для аналітичного подання алгоритмів та отримання на їх основі нових більш ефективних версій, є зручною і потужною основою для розробки автоматизованого інструментарію перетворення алгоритмів.

Створення такого інструментарію дозволяє значно полегшити роботу при створенні та дослідженні паралельних версій алгоритмів та значно скоротити час, що витрачається на аналіз та реалізацію підходів до оптимізації того чи іншого алгоритму.

Погорілий С.Д. // Проблеми програмування. - 2010. № 2-3. -Спеціальний випуск. Матеріали міжнародної конференції УкрПРОГ 2010. -стор. 349-354.

5. С.Д. Погорілий Концепція створення гнучких гомогенних архітектур кластерних систем / С.Д. Погорілий, Ю.В. Бойко, Д.Б. Грязнов, О.Д. Ломакін, В.А. Мар'яновський // Проблеми програмування. 2008. № 2-3. Спеціальний випуск. Матеріали міжнародної конференції УкрПРОГ 2008.

6. Інформація про обчислювальний кластер Інформаційно-Обчислювального центру Київського національного університету імені Тараса Шевченка [Електронний ресурс] : <http://cluster.univ.kiev.ua>

УДК 004.272:004.31

І. Д. Яковлева, к.т.н., І. Д. Лісовенко,
З. Р. Кудринський*

Автоматизована верифікація VHDL-моделей алгоритмічних операційних пристроїв швидкого перетворення Фур'є з фіксованою комою

У даній роботі запропоновано метод автоматизованої верифікації VHDL-моделей алгоритмічних операційних пристроїв (АОП) швидкого перетворення Фур'є (ШПФ) з фіксованою комою. Метод реалізовано в складі програмного забезпечення для автоматизованого синтезу з графічного рівня VHDL-моделей АОП комплексного ШПФ з частотним проріджуванням за основою 2 з фіксованою комою.

Ключові слова: алгоритмічний операційний пристрій, ШПФ, верифікація.

*E-mail: i.yakovleva@chnu.edu.ua, liss_i@mail.ru, kudrynskyi@gmail.com

Статтю представив с.н.с., д.ф.-м.н. Буй Д.Б.

Вступ. За допомогою аналізу Фур'є фахівці з аеронавтики і досліджень космічного простору підвищують чіткість зображень небесних тіл, сфотографованих з космічних апаратів [1]. Автоматичні міжпланетні станції і штучні супутники Землі передають інформацію на Землю у вигляді послідовностей радіоімпульсів, комп'ютери обробляють ці імпульси за допомогою швидкого перетворення Фур'є (ШПФ), і при цьому виконується модулювання окремих компонентів кожного перетворення, щоб чітко виділити одні властивості і усунути інші, аналогічно тому, як за допомогою перетворення Фур'є усувається шум з сигналу музичного запису. Зрештою, змінені таким чином дані знову перетворюються до початкової форми, і тим самим відновлюється зображення. За допомогою описаного процесу виконується фокусування зображення, фільтрація туманного фону та регулюється контрастність.

Перетворення Фур'є відіграє також дуже важливу роль у фізиці плазми і напівпровідникових матеріалів, мікрохвильовій акустиці, сейсмології, океанографії, радіолокації і медичних обстеженнях [1]. Серед численних застосувань в хімії можна назвати використання перетворення Фур'є в спектрометричному аналізі.

I. D. Iakovlieva, Ph.D., I. D. Lisovenko, Z.R.
Kudrynskyj

The Automated Verification of VHDL-Models of Algorithmic Operation Fast Fourier Transform Devices With Fixed Point

The method of the automated verification of VHDL-models of algorithmic operation Fast Fourier Transform (FFT) devices (AOD) with fixed point is proposed. The method is implemented as a part of the software for automated synthesis from graphical level of AOD VHDL-models for complex FFT with the frequency sampling on a basis 2 with the fixed point

Key Words: algorithmic operation device, FFT, verification.

В останні роки, завдяки досягненням в області інтегральної технології, суттєво зросла сміливість інтегральних схем і з'явилась можливість реалізації в них доволі складних комп'ютерних пристроїв. До таких пристроїв можна віднести і паралельні процесори ШПФ, що застосовуються при вирішенні задач цифрової обробки сигналів і зображень [2].

Крім основних вимог до засобів проектування комп'ютерних пристроїв: скорочення часу проектування та їх швидкодії, актуальним залишається верифікація розроблених пристроїв.

В даній роботі запропоновано для скорочення часу проектування пристроїв ШПФ та їх верифікації визначати похибку пристроїв ШПФ шляхом автоматизації порівняння результатів тестових послідовностей з виходу розробленого пристрою ШПФ та середовища Matlab.

Огляд літератури. Висока швидкодія комп'ютерних пристроїв забезпечується використанням алгоритмічних операційних пристроїв (АОП), в яких алгоритм виконання операції реалізовується апаратно.

В [3] запропоновано виконувати проектування АОП, який передбачає апаратну реалізацію потокового графа алгоритму (ПГА)

шляхом відображення комбінаційними схемами вершин графа алгоритму, і з'єднанні їх між собою відповідно до цього графа.

ПГА ШПФ відповідно до [4] подається структурною матрицею (СМ), а перехід від СМ до структурного опису АОП відбувається шляхом вибору з СМ ненульових елементів, значення яких відповідає номеру функціонального оператора, і співставленням їм комбінаційних схем із VHDL-бібліотеки, а також опису зв'язків між ними [3]. Користувач задає конфігураційні параметри АОП ШПФ, такі як розмірність перетворення, внутрішню та зовнішню розрядність даних, ступінь конвеєризації, тип перетворення (пряме або зворотне), порядок вхідних та вихідних даних (прямий або двійково-інверсний). Згідно цих параметрів синтезується VHDL-модель АОП ШПФ з фіксованою комою, що являють собою проект середовища Active-HDL.

Існуючі засоби проектування забезпечують автоматизацію синтезу АОП з графічного рівня, коли функціональну схему АОП подають однією з мов опису апаратних засобів, зокрема VHDL, Verilog [5, 6] тощо.

В основі структурно-функціональних підходів до проектування та верифікації радіoeлектронної апаратури лежить дуалізм коду мови опису апаратних засобів.

З одного боку, така модель – це опис у явному вигляді на алгоритмічній мові з формалізацією мовних конструкцій та наявністю спеціалізованих середовищ розробки. Формальність конструкцій та їх перетворення полегшує аналіз коду. Відкомпільований та проаналізований опис виконується (моделюється) з візуальним відображенням результатів у середовищі, яке імітує реалізацію паралельної роботи апаратури.

З іншого боку, – це опис цифрового пристрою. Стандартизовані фрагменти коду перетворюються у схемні реалізації, які використовують бібліотечні елементи системи синтезу. Синтезовані моделі перетворюються у схемні реалізації пристрою (логіки, що програмується користувачем, замовних HBIC (надвеликих інтегральних схемах), мікропроцесорних структур).

Таким чином, опис пристрою на мові опису апаратури – це фактично схема, і до неї можуть бути застосовані методи схемного аналізу,

синтезу тестів, побудови алгоритмів пошуку дефектів, які достатньо добре розроблені для цифрових схем. Але системи верифікації, які використовуються у світовій практиці (HDL Score, Verix, Hammer 100, SpyGlass, Questa AFV), не працюють у режимі функціональної верифікації на високому рівні абстракції [7, 8].

Постановка задачі. Оскільки, розроблена система автоматизованого синтезу АОП ШПФ дозволяє синтезувати пристрої з багатьма параметрами, а існуючі засоби не дають можливості виконувати верифікацію розроблених АОП, то постала задача створення системи тестування та верифікації синтезованих пристроїв.

Автоматизована верифікація VHDL-моделей алгоритмічних операційних пристроїв швидкого перетворення Фур'є. Найбільш трудомістким процесом при виконанні тестування розроблених пристроїв – є процес створення файлу вхідних векторів, які поступають на вхід модуля тестування та аналіз одержаних результатів.

Окрім того, зміна параметрів розроблених пристроїв вимагає відповідних змін у системі тестування. Для даних пристроїв задача ускладнюється ще й тим, що їх вхідні вектори представляються числами з фіксованою комою у спеціальному форматі, а отже, і система тестування вимагає перетворення чисел у відповідні формати. Таким чином, процес розробки тестів та аналізу їх результатів перетворюється у довготривалий та кропіткий процес, а людський фактор значно підвищує ймовірність виникнення помилки.

З урахуванням всіх перерахованих вище причин, доцільною являється автоматизація процесу розробки тестів, їх виконання та аналізу результатів.

У даній роботі перевірка функціонування автоматично синтезованих АОП ШПФ здійснюється методом моделювання його роботи з допомогою т.зв. тестбенча [9] та автоматичного порівняння цих результатів з результатами, обчисленими засобами середовища Matlab, які приймаються за еталонні. Після чого робиться висновок про коректність роботи автоматично синтезованих пристроїв.

Алгоритм автоматизованої верифікації синтезованих VHDL-моделей АОП ШПФ з фіксованою комою показано на рис.1.



Рис. 1. Алгоритм автоматизованої верифікації синтезованих VHDL-моделей АОП ШПФ з фіксованою комою

При автоматизованому синтезі АОП користувач вибирає число тестів, які будуть згенеровані для розробленого пристрою. Кожен такий тест містить n комплексних чисел, згенерованих псевдо-випадковим чином, де n – розмірність ШПФ (число точок відліку). Дані тестові послідовності зберігаються у двох форматах, один з яких, призначений для обробки в середовищі Matlab, а інший для тестбенча, реалізованого засобами VHDL. Таким чином, програма створює два файли з однаковими

тестовими послідовностями, але представленими у різних форматах: `vhdl_inputs.txt` (призначений для тестбенча) та `matlab_inputs.txt` (призначений для середовища Matlab). Після цього виконується синтез тестбенча засобами мови VHDL, відповідно до параметрів розробленого пристрою.

Тестбенч реалізовано у вигляді VHDL-модуля, що входить до VHDL-проекту АОП ШПФ. Даний модуль містить в якості під-модуля синтезований АОП ШПФ. Крім запису результатів моделювання у файл `vhdl_outputs.txt`, процес тестування також відображається на часовій діаграмі, яка є доступною для візуального перегляду підчас процесу моделювання.

На наступному етапі, тестування виконується за допомогою середовища Matlab [8, 9]. Програма для верифікації АОП ШПФ написана мовою Matlab, представлена у вигляді т.зв. М-сценарію і використовує функцію `fft()/ifft()`. Результати отримані в Matlab приймаються за еталонні та порівнюються з результатами, одержаними на попередньому етапі і збереженими в Matlab.

Для кожного тестового набору вхідних даних обчислюється абсолютна похибка, а після виконання всіх обчислень розраховується максимальне значення абсолютної похибки та середнє значення абсолютної похибки (рис.2).

```

Absolute_Error =
0
0
0
0
0.000202658222223 + 0.003708370845261i
0.003703591777779 + 0.000197871154739i
0.000089071058067 + 0.001769084101116i
0.003817178941935 + 0.002137165898887i
    
```

```

Max_Error =
0.004282333729927
    
```

```

Average_Absolute_Error =
9.738479193321432e-004
    
```

Рис. 2. Результат автоматизованої верифікації синтезованих VHDL-моделей АОП ШПФ з фіксованою комою для $n=8$

З допомогою автоматизованої системи верифікації було визначено точність спроектованих АОП ШПФ. Для цього були синтезовані VHDL-моделі АОП ШПФ з фіксованою комою з наступними параметрами: тип перетворення – пряме; порядок вхідних даних – прямий; порядок вихідних даних – двійково-інверсний; зовнішня розрядність – 16; внутрішня розрядність – 32; ступінь конвеєризації – максимальна. Для кожної VHDL-моделі згенеровано 1000 тестових наборів, на яких була промодельована їх робота. Як видно на рис. 3, зі збільшенням розрядності ШПФ зростає і похибка, що зумовлено операціями множення, при яких відбувається відкидання надлишкових молодших розрядів. Різне зростання значень похибки на початку графіка зумовлено тим, що при малих розмірностях ШПФ більшість метеликів являються виродженими, тобто такими в яких не виконується операція множення, а отже і їх результати є досить точними.

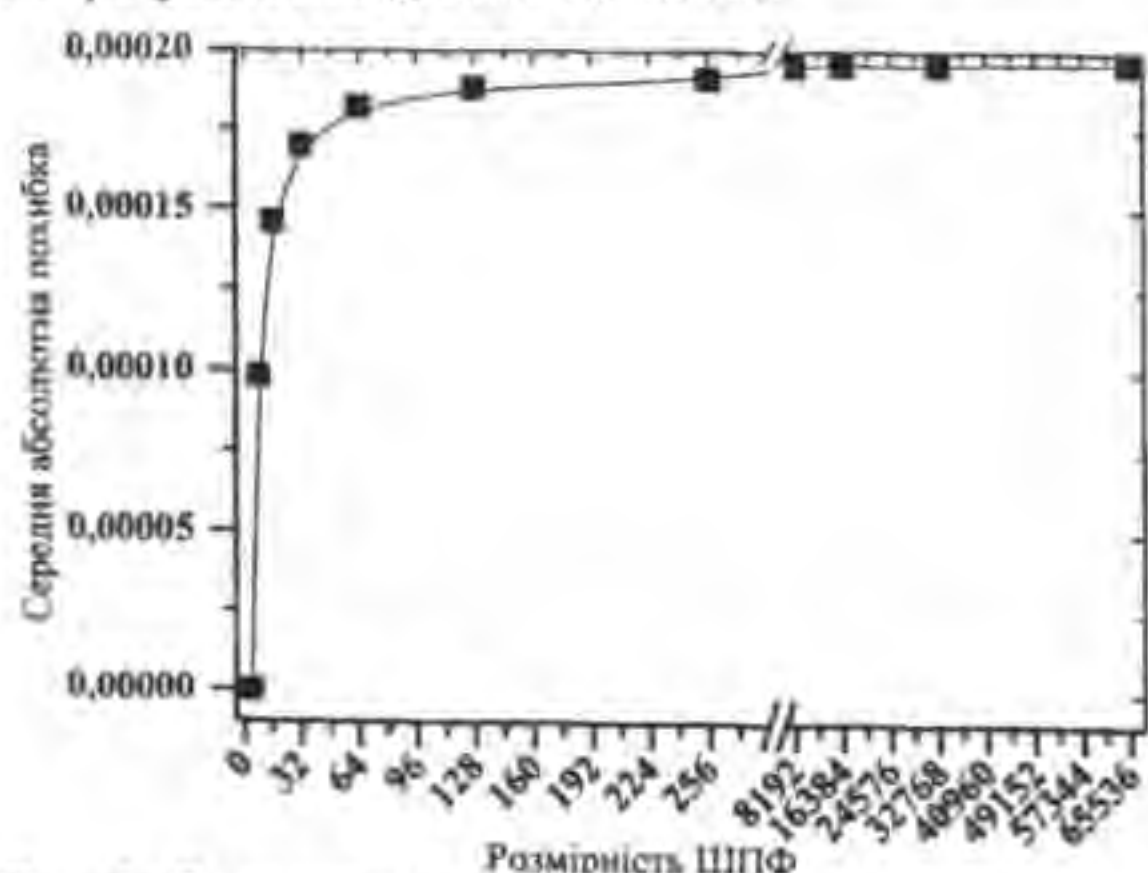


Рис. 3. Середня абсолютна похибка при різній розмірності ШПФ

Висновки. У даній роботі розроблено метод автоматизованої верифікації VHDL-моделей алгоритмічних операційних пристроїв швидкого перетворення Фур'є з фіксованою комою. Система автоматизованої верифікації полягає у поданні вхідних векторів даних, що є комплексними числами у спеціальному форматі, на вхід АОП ШПФ і Matlab, із подальшим визначенням похибок. На підставі отриманих результатів робиться висновок про коректність роботи синтезованих АОП.

Метод реалізовано в складі програмного забезпечення для автоматизованого синтезу з графічного рівня VHDL-моделей АОП комплексного ШПФ з частотним

проріджуванням за основою 2 з фіксованою комою.

Список використаних джерел

1. Bracewell R. N. The Fourier Transform / Ronald N. 1. Bracewell // Scientific American. – June 1989. – 62-69 p.
2. Мельник А. О. Спеціалізовані комп'ютерні системи реального часу / А. О. Мельник – Львів: Вид-во Національного університету "Львівська політехніка", 1996. – 60 с.
3. Мельник А. О. Метод перетворення графічного подання алгоритму в його апаратну модель / А. О. Мельник, І. Д. Яковлева // Науковий вісник Чернівецького ун-ту. Фізика. Електроніка. Вип. 423. – Чернівці: Чернівецький національний університет імені Юрія Федьковича. – 2008. – С. 19–23.
4. Мельник А.О. Особливості побудови структурної матриці потокових графів алгоритмів з множинними операціями / А.О. Мельник, І.Д. Яковлева // Науковий журнал "Технічні науки" – Хмельницький: Хмельницький національний університет, 2008. – №5 – С. 117–120.
5. IEEE, Standard VHDL Language Reference Manual. Standard 1076 –1993, NY: IEEE, New York, 1993.
6. IEEE, Standard Verilog Hardware Description Language Reference Manual. Standard 1364 – 1995, NY: IEEE, New York, 1995.
7. SpyGlass. Early Design Analysis for Logic Designers [Електронний ресурс] : – Режим доступу: <http://www.newplus.com.cn/products/atrenta/Spyglass.pdf>
8. Questa [Електронний ресурс]: Режим доступу: <http://www.triadmicro.com/pdfs/eda/FPGA/questa-SV-Datasheet.pdf>
9. Семенец В.В. Проектирование цифровых систем с использованием языка VHDL / В.В. Семенец, И.В. Хаханова, В.И. Хаханов. – Харків: ХНУРЕ, 2003. – 492 с.
10. Мартынов Н.Н. MATLAB 7. Элементарное введение / Н.Н. Мартынов. – М.: КУДИЦ-ОБРАЗ, 2005. – 416 с.
11. Дашенко А.Ф. MATLAB в инженерных и научных расчетах / А.Ф. Дашенко, В.Х. Кирилов, Л.В. Коломиец, В.Ф. Оробей. – Одесса: Астропринт, 2003. – 214 с.

Надійшла до редколегії 15.12.10

РАДІОФІЗИКА

УДК 004.94:533.9

Богданов Р.В., аспірант

**Комп'ютерне моделювання
розпорошувального магнетронного
розряду: енергетичний розподіл
бомбардуючих іонів**

Була розроблена комп'ютерна модель плоского розпорошувального магнетронного розряду. Рух частинок описувався методом Монте-Карло. В результаті моделювання були одержані енергетичні спектри іонів, які бомбардують катод. Досліджені залежності позицій іонізацій, кутів падіння іонів, часу прольоту та їхньої кількості від тиску робочого газу та напруженості магнітного поля.

Ключові слова: магнетронний розряд, комп'ютерне моделювання, метод Монте-Карло, позиції іонізації, енергетичний спектр.

Bogdanov R.V., post grad. stud.

**The computer simulation of the sputtering
magnetron discharge: the energy
distribution of impacting ions**

Computer model of planar sputtering magnetron discharge has been developed. Monte Carlo method has been used for describe the motion of particles. Energy spectra of ions which bombard the cathode were obtained as the results of simulation. Dependences of ionization positions, impact angles of ions, the transit time and their quantity from working gas pressure and magnetic field intensity have been investigated.

Key Words: magnetron discharge, computer simulation, Monte Carlo method, ionization positions, energy spectrum.

*E-mail: RSemsterFX@gmail.com,

Статтю представив д. ф.-м. н., проф. Анісімов І.О.

Вступ

Магнетронний розряд постійного струму є різновидом жевріючого розряду, в якому існує область зі схрещеними електричними та магнітними полями [1, 2]. Метод магнетронного розпорошення матеріалів є одним з найбільш ефективних методів нанесення плівок у планарній технології. У роботі [3] досліджена можливість утворення плівок, що мають у собі та на своїй поверхні вуглецеві нанотрубки та фулерени C_{60} за умов розпорошення графітової мішені в аргоні у магнетронному розряді.

Розпорошення матеріалів у магнетроні відбувається за рахунок бомбардування поверхні катоду-мішені іонами робочого газу, прискореними прикатодним падінням потенціалу. Їх енергія повинна перевищувати енергію сублімації твердого матеріалу і становить десятки електрон-вольт. Таким чином, для оцінки потоку розпорошеного матеріалу з поверхні мішені, потрібно визначити залежність енергії, середнього кута падіння, кількості падаючих іонів від напруженості полів, тиску та сорту робочого газу.

Вимірний зондовим методом розподіл потенціалу в реальному плоскому магнетроні наведений на рис. 1 [4]. Як видно з рис. 1, на прикатодній ділянці розряду спадає його більша

частина, а на іншій ділянці – лише декілька десятків вольт, як у жевріючому розряді без магнітного поля.

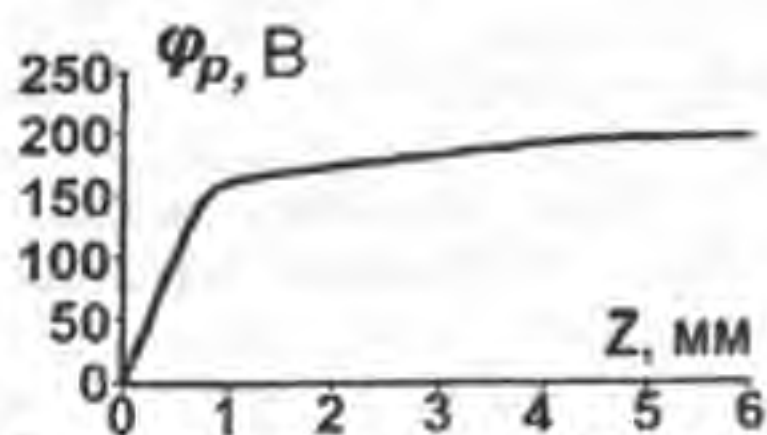


Рис. 1. Хід електричного потенціалу в реальному магнетроні в аргоні при 0,4 Па та 225В [4].

У найпростішому випадку припускається, що залежність напруженості електричного поля у прикатодному шарі має наступний вигляд [2]:

$$E(z) = -\left(\frac{V_0}{d_E}\right) \cdot \left(1 - \frac{z}{d_E}\right), \quad (1)$$

де V_0 – прикатодне падіння напруги, d_E – ширина області катодного падіння. За типових умов процесів розпорошення, значення напруги, що спадає на прикатодному шарі просторового заряду, – 150-600 В.

Магнітні поля в технологічних магнетронах складають $B = 0,01-0,1$ Т [1] та мають складну конфігурацію. Для аналізу простіше розглянути одновимірне магнітне поле, що є паралельним до

поверхні катоду-мішені і нормальним до електричного поля в розряді.

За відсутності зіткнень у схрещених електричних і магнітних полях, електрони рухаються по циклоїді та періодично повертаються на поверхню катода. Електронна складова струму в магнетроні можлива лише за умов переходу електронів між траєкторіями в напрямку електричного поля при зіткненнях. При максимальному віддаленні від катода, що дорівнює висоті циклоїди h_c (приблизно два циклотронні радіуси), електрон набуває в електричному полі енергії W_c , що становить

$$W_c = e\varphi_c = e \cdot h_c \cdot E_c, \quad (2)$$

де e – заряд електрона; φ_c – потенціал у точках розрядного проміжку, віддалених від поверхні катода на відстань h_c ; E_c – напруженість електричного поля в цьому проміжку.

Висота циклоїди h_c визначається виразом [1, 4]:

$$h_c = 2m_e E_c / eB^2, \quad (3)$$

де m_e – маса електрона, B – напруженість магнітного поля.

Враховавши (2), звідки $h_c = \varphi_c / E_c$, отримаємо величину напруженості електричного поля біля катода:

$$E_c = B \sqrt{e\varphi_c / 2m_e}. \quad (4)$$

При φ_c , що дорівнює потенціалу іонізації атома аргону (15,8 eV), та типових значеннях величини індукції магнітного поля для магнетрона на постійному струмі $B = 0,05$ Т, з (3) маємо напруженість електричного поля $E_c = 5,9 \times 10^4$ В/м. При напруженості поля, меншій за цю величину, енергія електронів, що будуть рухатися по циклоїді, буде меншою за потенціал іонізації, і самостійний розряд неможливий. Тоді максимальна ширина темного катодного простору $h_c = 0,27$ мм. Ширину області інтенсивної іонізації d , можна визначити за умов однорідного електричного поля при напрузі розряду $V_0 = 400$ В:

$$d, \leq V_0 / E_c \approx 6,7 \text{ мм}. \quad (5)$$

Таким чином, основні процеси відбуваються у тонкій прикатодній області магнетрона.

Метою даної роботи є побудова комп'ютерної моделі розпорошувального магнетронного розряду для нанотехнологій. На даному етапі було розроблено алгоритм, що

дозволяє якісно проаналізувати процеси іонізації та прискорення частинок робочого газу в прикатодному шарі магнетронного розряду, виходячи з його параметрів: напруги живлення, напруженості магнітного поля, робочого тиску.

Алгоритм моделювання магнетронного розряду

Опис газового розряду методом частинок потребує розгляду зіткнень між частинками.

У даній роботі враховано такі види зіткнень у робочому газі аргоні: пружні зіткнення електронів з атомами, іонізація електронним ударом (враховуються таунсендівські електрони), збудження атомарних рівнів електронами (переріз сумарний для рівнів $4s-4p$, іонізацією та рекомбінацією збуджених атомів нехтувалося [1, 7]). Для іонів розглядалися зіткнення з атомами робочого газу, серед яких: пружні зіткнення, резонансна перезарядка, поляризаційні зіткнення (для малих енергій). Дані про перерізи зіткнень електронів та важких частинок в аргоні були взяті з [2, 5].

Модель магнетрона, подібна до [6, 7, 8], є оптимізованою для моделювання та являє собою два паралельних електроди (рис. 2). Початок координат вибраний у точці на катодній площині так, як зображено на рис. 2. Між електродами створюється електричне поле, обране у вигляді (1), що антипаралельне до вісі z . Коли відстань до катода перевищує d_E , напруженість електричного поля прийняте $E = -25$ В/м. Магнітне поле B обране постійним, антипаралельним осі y , і, таким чином, воно є тангенціальним до поверхні катоду та нормальним до електричного поля.

Електрони в такій системі дрейфують у від'ємному напрямку осі x .



Рис. 2. Модель магнетрона.

Алгоритм моделювання заснований на інтегруванні рівняння руху частинок та

розрахунку ймовірності зіткнень методом Монте-Карло:

1. На початку моделювання розглядаються електрони, емітовані з поверхні катода в області по вісі x : $\{-30 \text{ мм}; 30 \text{ мм}\}$, по y : $\{-10 \text{ мм}; 10 \text{ мм}\}$. Розрахунок починається з поверхні катода $z=0$, а позиції виходу електронів з поверхні обираються випадково на основі рівномірного розподілу. Енергії електронів, що вилітають з катода, дорівнюють 0 для виключення явища захоплення на катод [1]. Прийнято, що електрон при досягненні катода відбивається від нього.

2. Поміж зіткненнями електрони та іони рухаються у схрещених електричних та магнітних полях, де рівняння руху

$$\dot{\vec{v}} = \left(\frac{q}{m} \right) (\vec{E} + \vec{v} \times \vec{B}) \quad (6)$$

інтегрується методом Рунге-Кутта 4-го порядку. Фіксований крок інтегрування за часом обирається достатньо малим для забезпечення необхідної точності та оптимального часу розрахунку. Для електронів часовий крок дорівнює $dt = 10^{-11} \text{ с}$ ($dt \ll \tau_c$, де τ_c – період циклотронного обертання електрона у тангенціальній до катода компоненті магнітного поля). Для іонів аргону $dT = 7,408 \times 10^{-9} \text{ с}$. Ці значення кроку оптимальні для широкого діапазону зміни магнітного поля – від 0 до 0,2 Т.

3. Ймовірність зіткнень визначалася методом Монте-Карло на шляху Δs пройденому електроном за один часовий крок, виходячи з відомого перерізу зіткнень [7, 8]. Звідси критерій зіткнення:

$$RN_i < 1 - \exp(-\Delta s n_a \sigma), \quad (7)$$

де RN_i – випадкове число в межах від 0 до 1, що генерується в процесі розрахунку та визначає момент зіткнення, n_a – концентрація частинок робочого газу, σ – переріз зіткнення.

При пружних зіткненнях електрон змінює напрямок швидкості випадковим чином. При іонізації енергія електрона зменшується на величину енергії іонізації атома аргону – 15,8 еВ. За цих умов утворюється однократно іонізований атом у відповідній позиції та таунсендівський електрон. У подальшому в цій точці генерується випадковий напрямок руху для електронів та іонів із розподілу Максвелла. При збудженні атома електрон втрачає енергію 11,7 еВ.

4. Моделювання іонів починається зі створення іонів у відповідних позиціях іонізації, визначених на основі моделювання для

електронів. Ймовірність зіткнень визначається аналогічно до пункту 3. Пружні зіткнення розглядаються в моделі твердих сфер з відповідними масами частинок. Нейтралі мають теплові швидкості, розподілені за Максвеллом. Втрата енергії іоном аргону при зіткненні з нейтралом при пружній взаємодії оцінюється як

$$\Delta W = W_0 \sin^2 \theta, \quad (8)$$

де W_0 – енергія іона до зіткнення, ΔW – втрата енергії налітаючою частинкою, θ – кут розсіювання, що обирається з виразу $\cos \theta = R_1$; азимутальний кут розсіювання $\phi = 2\pi R_2$, де R_1, R_2 – випадкові числа в межах від 0 до 1. Кут, під яким розсіюється атом газу, дорівнює $(\pi/2 - \theta)$. Кути перераховуються у лабораторну систему координат, пов'язану з магнетроном (рис. 2). Поляризаційні зіткнення [2] враховувалися подібним чином. При резонансній перезарядці у відповідній точці створюється новий іон з тепловою швидкістю, а іон, що налітає, втрачає заряд.

5. Розрахунок для частинки зупиняється, коли частинка виходить за межі розряду (90 мм × 40 мм × 20 мм). Для зменшення часу розрахунку можливо нехтувати рухом важких незаряджених частинок, енергія яких менше 0,5 еВ, оскільки їх кількість не зможе значно вплинути на загальний енергетичний розподіл фонових частинок. Також за межами прикатодного падіння потенціалу нехтується електронами, що мають енергію менше потенціалу іонізації, оскільки вони не зможуть набрати достатньо енергії у слабкому полі.

Програма написана мовою C# 2.0 (2008). На розрахунок при 3000 стартових електронах витрачається біля 25 хвилин на одному ядрі процесора AMD Athlon 64x2 з частотою 3,1 ГГц.

Результати моделювання

Моделювання було проведено для двох тисків: 3 Па, 10 Па. Прикатодне падіння напруги становить $V_0 = 300 \text{ В}$ та 600 В, область катодного падіння $d_E = 1 \text{ мм}$. Наступні напруженості магнітного поля B встановлювалися: 0 Т, 0,02 Т, 0,05 Т, 0,08 Т. Запускалося 3000 стартових електронів. У статті переважно наводяться результати для $V_0 = 600 \text{ В}$, $P = 10 \text{ Па}$. При $V_0 = 300 \text{ В}$ процеси подібні.

Визначалися такі величини (табл. 1): кількість частинок, що впали на поверхню катода, N_n ; число іонів, створених в об'ємі розряду, N_i ; число зіткнень іонів з нейтралами,

що спостерігалися, N_{col} , середній кут падіння $\langle \theta \rangle$ (відраховується від нормалі та усереднюється по всім частинкам, що впали на катод); середній час прольоту іонів до катода $\langle T \rangle$; середня енергія падаючих частинок $\langle W \rangle$. Окремо можна виділити усереднену позицію іонізацій вздовж осі $z - \langle Z_i \rangle$:

$$\langle Z_i \rangle = \frac{1}{N_i} \sum_{i=1}^{N_i} z_i(x, y), \quad (9)$$

де $z_i(x, y)$ – позиція іонізації з номером i по осі z , що у площині (x, y) , N_i – кількість іонізацій [6].

Розподіл іонізацій вздовж осі z

За межами прикатодного падіння потенціалу електрони втрачають енергію при зіткненнях та не можуть отримати значного прискорення у слабкому електричному полі, аж до моменту коли будуть нездатні іонізувати атоми (рис. 3).

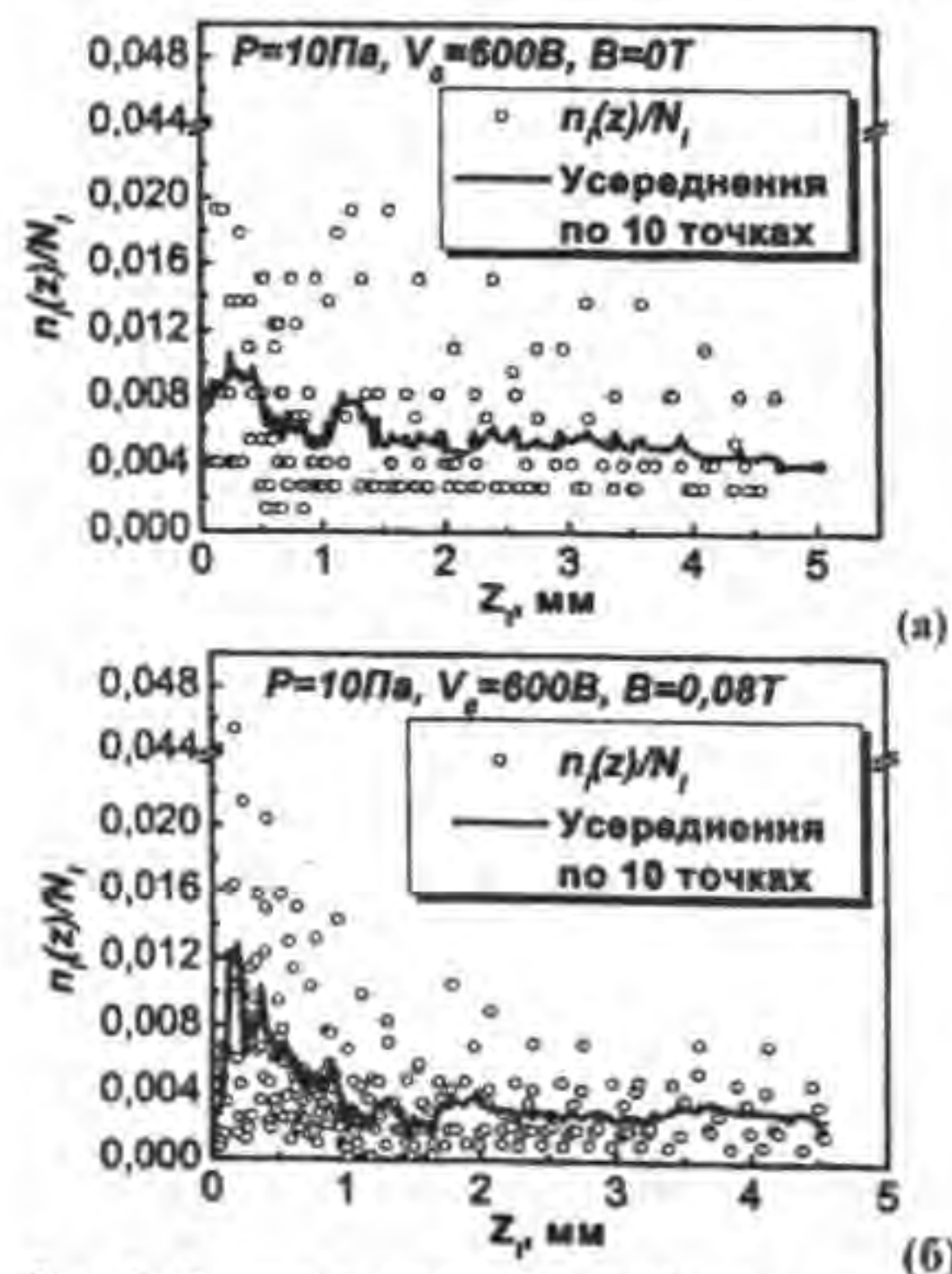


Рис. 3. Розподіл іонізацій $n_i(z)/N_i$ по осі z при $P=10$ Па, 3000 стартових електронах, $V_0=600$ В, прикатодному шарі $d_k=1$ мм та напруженості магнітного поля (а) $B=0,0$ Т, (б) $B=0,08$ Т. Для наочності виконано усереднення в програмі Origin 7.0.

За тиску 10 Па без магнітного поля (рис. 3, а), кількість іонізацій майже рівномірно

розподілена аж до відстані $z=5$ мм, що є меншою за ширину розрядного проміжку (20 мм). З магнітним полем (рис. 3, б) кількість іонізацій має максимум поблизу катода, що відповідає виразу (3), кількість іонізацій поблизу катода зростає. Область інтенсивної іонізації скорочується (див. оцінку (5)). За умов зіткнень, електрони рухаються в напрямку електричного поля [1, 2], тому скорочення цієї області незначне.

Енергетичний розподіл бомбардуючих частинок

При збільшенні магнітного поля максимальна висота, яку може досягнути електрон, зменшується (рис. 3, б). Це, у свою чергу, зменшує енергію, що можуть набрати створені іони на шляху до катода (рис. 4, б).

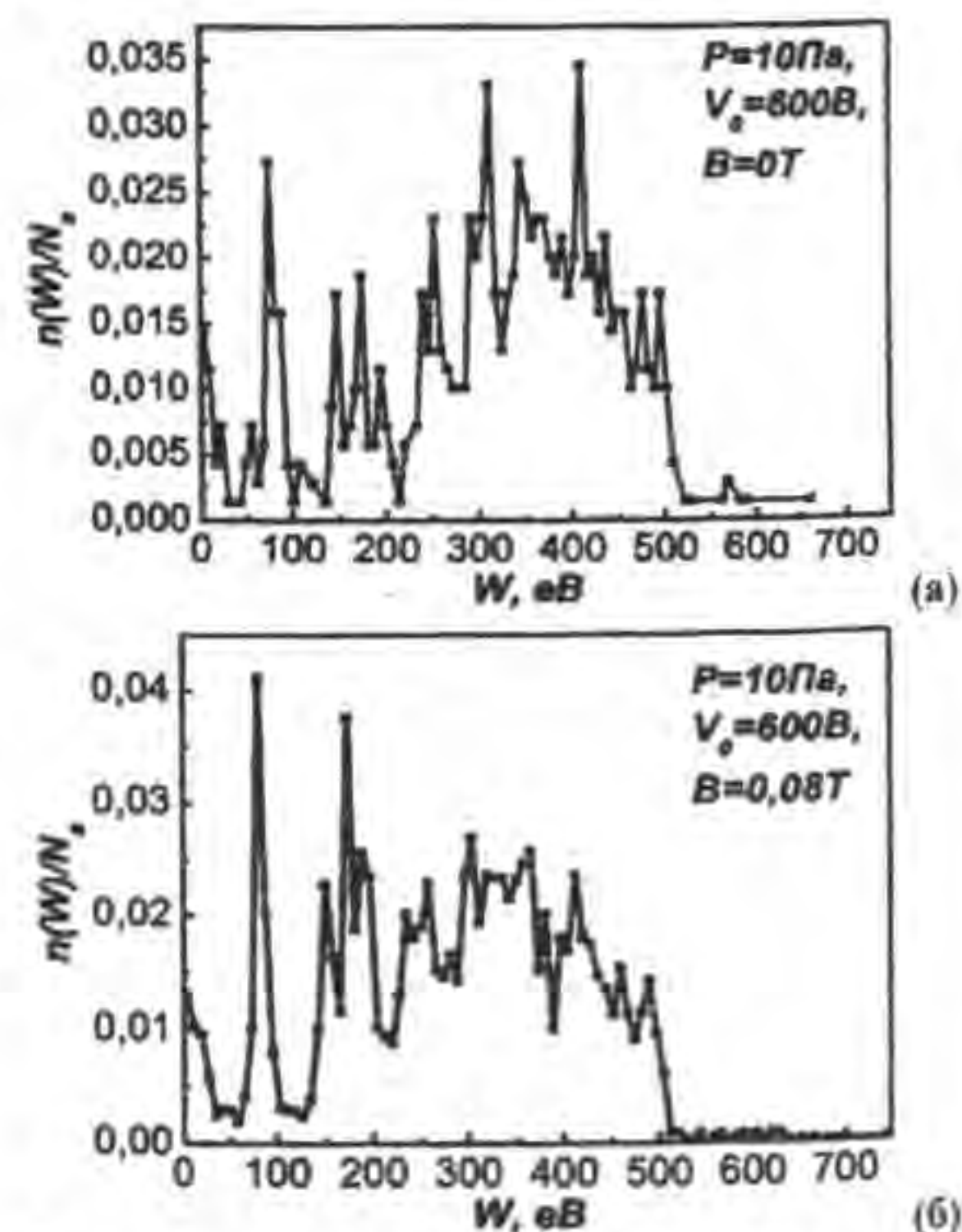


Рис. 4. Розподіл енергії бомбардуючих частинок $n(W)/N_i$, нормований на всю кількість частинок на поверхні при: $P=10$ Па, 3000 стартових електронах, $V_0=600$ В, прикатодному шарі $d_k=1$ мм: (а) $B=0,0$ Т, (б) $B=0,08$ Т.

При збільшенні магнітного поля кількість іонізацій поблизу катода збільшується (рис. 3, б), за рахунок магнітного утримання. Шлях пройдений частиною іонів у прискорювальному полі зменшується, що зменшує їх загальну енергію (рис. 4, б) – у порівнянні з (рис. 4, а)

помітно зростають піки біля 100 eV та 200 eV. Але в загальному випадку, енергетичні спектри (рис. 4) виглядають більш-менш подібно (існують частинки з енергіями, близькими до

$eV_0 \sim 600$ eV), оскільки більшість іонів створюються за межами прискорювального проміжку.

Таблиця 1.

Результати моделювання для магнетронного розряду при різних тисках

P , Па	B , Т	$\langle Z_i \rangle$, Мм	$\langle W \rangle$, eV	N_s	N_i	N_s/N_i	N_{col}	N_{col}/N_i	$\langle \theta \rangle$, градусів	$\langle T \rangle$, мкс
3	0	1,704652	340,59	200	220	0,91	70	0,318	2,4134	3,653
3	0,02	1,703016	343,46	199	220	0,905	68	0,31	1,9609	3,682
3	0,05	1,674625	332,57	974	1072	0,909	387	0,361	1,1188	3,642
3	0,08	1,570069	327,59	1764	1851	0,953	683	0,369	0,7743	3,463
10	0	1,642616	307,76	696	728	0,956	1007	1,383	3,6588	3,465
10	0,02	1,501797	306,08	783	817	0,958	974	1,192	2,6874	3,068
10	0,05	1,484179	302,6	4061	4205	0,966	5253	1,249	3,553	3,145
10	0,08	1,166445	279,29	9276	9482	0,978	10576	1,115	3,6553	2,257

Катодне падіння $V_0=600$ В, прикатодний шар $d_E=1$ мм, запускалися 3000 електронів

Вплив тиску та магнітного поля на загальні результати

При зміні тиску від 3 Па до 10 Па за рахунок зміни довжин вільного пробігу електронів, кількість іонізацій N_i при відсутності магнітного поля збільшується приблизно у три рази (табл. 1).

За наявності магнітного поля кількість іонізацій N_i зростає майже як B_2^2/B_1^2 , де $B_2 > B_1$ (табл. 1). Цей факт можна пов'язати зі зменшенням рухливості електронів $b_{e\perp}$ поперек магнітного поля [1]:

$$\vec{v}_{eE} = b_{e\perp} \vec{E}, \quad (10a)$$

$$b_{e\perp} = \frac{m_e}{eB^2\tau_e}, \quad (10)$$

де $\vec{v}_{eE}$ – швидкість руху електрона у напрямку електричного поля E (при зіткненнях), m_e – маса електрона (або приведена маса), e – елементарний заряд, τ_e – час між зіткненнями електронів у газі. Оскільки електричне поле витягує електрони на катод, зменшення рухливості подовжує час перебування електронів у розряді і, таким чином, збільшує кількість іонізацій.

Можна навести простий критерій ефективності магнітного утримання електронів [1]:

$$\omega_e \tau_e \gg 1, \quad (11)$$

де ω_e – циклотронна частота електрона, τ_e – час між зіткненнями електрона з молекулами

робочого газу за відсутності магнітного поля. Якщо прийняти, що повний переріз зіткнень складається з перерізів пружних зіткнень, іонізації електронним ударом, збудження рівнів становить $\sigma = 2,81 \times 10^{-19} \text{ м}^2$, то при енергії електронів біля 32 eV, що відповідають за іонізацію і тиску 10 Па, величина $\omega_e \tau_e$ дорівнює 1,49, 3,74, 5,98 для $B=0,02$ Т, 0,05 Т, 0,08 Т відповідно. При 3 Па $\omega_e \tau_e$ змінюється як 4,98, 12,4, 19,9 для тих самих значень магнітного поля. Таким чином, при 10 Па та 0,02 Т рух електрона по циклоїді не спостерігається. Тому кількість іонізацій майже така як випадку без магнітного поля. Очевидно, що зростання магнітного поля може компенсувати вплив зіткнень. При тиску 3 Па умови утримання кращі, час між зіткненнями τ_e більший, що зменшує рухливість $b_{e\perp}$ у порівнянні з 10 Па. Це повинно подовжувати час перебування електрона у розряді. Але розглянута модель не передбачає замкнений дрейф, що впливає на отриманий результат.

Зростання магнітного поля зменшує висоту підняття електронів над катодом, що зменшує $\langle Z_i \rangle$ (табл. 1). Зростання тиску також наближає позиції іонізацій до катода [2], але $\langle Z_i \rangle$ компенсується збільшенням кількості іонізацій та рухом електронів до анода при зіткненнях, як було вказано вище. При 10 Па вплив магнітного утримання електронів на $\langle Z_i \rangle$ помітніший ніж при 3 Па, що виражається у інтенсивнішій зміні $\langle Z_i \rangle$ при зростанні магнітного поля. Скорочення шляху $\langle Z_i \rangle$, пройденого створеними іонами в прискорювальному електричному полі зменшує

їх середню енергію $\langle W \rangle$ (табл. 1). Водночас можна помітити, що кожен іон зазнав зіткнень у випадку 10 Па, ($N_{col}/N_i > 1$), а при 3 Па зіткненнями переважно можна нехтувати. Це повинно впливати на їх середню енергію, яка зменшується при збільшенні тиску (табл. 1). Певна кількість частинок розсіюється за рахунок зіткнень і не потрапляє на катод, тому $N_s < N_i$.

Середній кут падіння $\langle \theta \rangle$ при 3 Па поступово зменшується, через наближення позицій іонізацій до катода, що зменшує і так незначне відхилення іонів у магнітному полі [1, 2]. У випадку тиску 10 Па більшу роль відіграють зіткнення, а не магнітне поле, тому $\langle \theta \rangle$ при 0,05 Т та 0,08 Т мало відрізняються від випадку без магнітного поля. Значення $\langle \theta \rangle$ при $P=10$ Па та $B=0,02$ Т можна вважати наслідком недостатньої кількості частинок за цих умов, що погіршує точність. Але в загальному випадку ця величина все одно більше за значення, що спостерігалися при $P=3$ Па (табл. 1).

Середній час прольоту $\langle T \rangle$ до катода змінюється слабо та відповідає зміні $\langle Z_i \rangle$ у випадку тиску 3 Па. При тиску 10 Па та напруженостях магнітного поля 0,02 Т та 0,05 Т $\langle T \rangle$ демонструє невелике зростання при зменшенні $\langle Z_i \rangle$, але часи майже однакові. Така поведінка може бути наслідком недостатньої точності у випадку $B=0,02$ Т.

Список використаних джерел

1. Кузьмичев А.Н. Магнетронные распылительные системы. Книга - 1. Введение в физику и технику магнетронного распыления. – Киев: Аверс, 2008. – 244с.
2. Райзер Ю.П. Физика газового разряда: Учебное руководство для вузов. – 2е изд. перераб. и доп. – М.: Наука, 1992. – 536с.
3. Антоненко С.В., Мальцев С.Н., Тимофеев А.А. ПЭМ исследования углеродных нанотрубок, полученных методом магнетронного распыления на постоянном токе // МИФИ <http://library.mephi.ru/data/scientific-sessions/2001/4/1501.html>.
4. Жуков В.В., Кривобоков В.П., Янин С.Н. Свойства магнетронного разряда на постоянном токе. Ч.1. Механизм распыления мишени // Известия Томского политехнического университета. – 2005. – 308, – №6, – С. 69-74.

Висновки

У результаті роботи була побудована комп'ютерна модель магнетронного розряду в аргоні, заснована на методі Монте-Карло.

За допомогою програми були отримані розподіли позицій іонізації вздовж осі від катода до анода, енергетичні спектри важких частинок, що потрапили на катод-мішень, часи їх прольоту до катода та їх кути падіння на катод, кількість частинок, що бомбардують катод. Проаналізовано вплив на ці величини різних тисків робочого газу та напруженостей магнітних полів. Зростання тиску збільшує кількість іонізацій, але зменшує енергію бомбардуючих частинок через більшу кількість зіткнень та наближення позицій іонізацій до катода. Зростання тангенціального до катода магнітного поля покращує утримання електронів в області інтенсивної іонізації, але водночас наближає позиції іонізації до катода, що зменшує енергію, набрану іонами, у прискорювальному електричному полі. Отримані результати відповідають поведінці реального магнетронного розряду.

Алгоритм буде використаний у подальшому розгляді процесу магнетронного розпорошення та умов формування наночастинок.

5. Ткачев А.Н., Феденев А.А., Яковленко С.И. Коэффициент Таунсенда, кривая ухода и эффективность формирования пучка убегających электронов в аргоне // Журнал технической физики. – М.: 2007. – 77, вып. 6. – С. 22-27.
6. Li Jun, Properties of ions impacting upon the cathode surface in a sputtering magnetron discharge // J. Phys. D: Appl. Phys., – 1999, – №29, – P. 1386-1390.
7. Musschoot J., Depla D., Buyle G., et al. Influence of the geometrical configuration on the plasma ionization distribution and erosion profile of a rotating cylindrical magnetron: a Monte Carlo simulation // Journal of Physics D: Applied Phys., – 2006, – №39, – P.3989-3993.
8. Musschoot J., Depla D., Buyle G., et al. Investigation of the sustaining mechanisms of dc magnetron discharges and consequences for I-V characteristics // Journal of Physics D: Applied Phys., – 2008, – №41, – P.1-5.

Надійшла до редколегії 19.10.10

УДК 537.8.029.6, 538.97-405, 621.382.092.6

Костенко В. І.,
Нечипорук О. Ю., к. ф.-м. н., доц.,
Сорочак А. М., асп.,
Чамор Т. Г., к. ф.-м. н.,
Чевнюк Л. В., к. ф.-м. н.

Експериментальне дослідження високочастотних властивостей магнітного гістерезису одновісних монокристалічних гексаферитів

Представлені оригінальні експериментальні результати в області досліджень природи механізмів магнітного гістерезису монокристалічних гексагональних феритів М-типу. З'ясовано, що зі збільшенням величини максимального постійного поля підмагнічування H_{0max} вище певного значення H_0 , до полів $H_{0max} > H_0$ (поле анізотропії), для конкретного зразка, значення поля зриву H_{zp} практично не змінюється. Сучасними уявленнями про існуючі механізми гістерезису неможливо обґрунтувати це явище, тому запропонований адекватний механізм для його пояснення.

Ключові слова: монокристалічний гексаферит, феромагнітний резонанс, магнітний гістерезис, частотно-польова залежність, поле зриву.

E-mail: sorandrew@gmail.com

Статтю представив д.ф.-м.н. Скришевський В. А.

Вступ

Необхідність поглибленого розуміння високочастотних та магнітних властивостей високоанізотропних гексаферитових матеріалів викликана потребою просування практичних застосувань сучасної радіофізики на якісно новий рівень. У зв'язку з цим, останніми роками значно підвищився інтерес до експериментального вивчення процесів перемагнічування та високочастотних властивостей гістерезису саме в об'ємних гексаферитових монокристалічних матеріалах [1, 8].

В роботах [2,3] приводяться результати експериментальних досліджень магнітних та високочастотних властивостей плівок $BaFe_{12}O_{19}$ вирощених методом рідкофазної епітаксії на різних підкладках. Виміряні статичні криві перемагнічування повністю узгоджуються з високочастотними. У плівки $t=2.1$ мкм [3], що

Kostenko V. I.,
Nechyporuk O. Yu., Ph.D., As. Prof.,
Soroachak A. M., Ph.D. stud.,
Chamor T. G., Ph.D.,
Chevnyuk L.V., Ph.D.

Experimental research of magnetic hysteresis high-frequency properties in uniaxial single-crystal hexaferrites

The original experimental results are presented in the researches area of nature of magnetic hysteresis mechanisms in single-crystal M-type hexagonal ferrites. It is founded out, that with the increasing of maximal permanent magnetic field H_{0max} value higher than the defined value of H_0 to the fields $H_{0max} > H_0$ (anisotropy field), for a concrete sample, the value of breaking field H_{br} doesn't change practically. It is impossible to explain this phenomenon by modern representations of existent hysteresis mechanisms, that why the adequate mechanism is offered for his explanation.

Key Words: single-crystal hexaferrite, ferromagnetic resonance, magnetic hysteresis, frequency-field dependence, breaking field.

вирощувалася методом Чохральського на підкладці $Sr_{1.03}Ga_{10.81}Mg_{0.58}Zn_{0.58}O_{19-8}$ заміщеного гексагалату стронцію (ГГС), ширина лінії ФМР $2\Delta H=45$ Е, отримана на частоті 53 ГГц. Для плівки товщиною $t=23$ мкм, вирощеної на підкладці MgO , найвужча ширина лінії ФМР, зафіксована у насиченні, становила $2\Delta H=27$ Е [2]. При проведенні високочастотних вимірювань гексаферитових плівок [2,3], виявлено гістерезис частотно-польових залежностей феромагнітного резонансу (ФМР), у випадку підмагнічування зовнішнім магнітним полем H_0 вздовж легкої вісі намагнічування ($H_0 \parallel [111]$). Необхідно відмітити, що для вищезгаданих досліджуваних зразків, перехід частотно-польових залежностей ФМР з насичення в багатодоменний стан відбувався не у "від'ємному" значенні H_0 [4], а у полі зриву H_{zp} , меншому поля насичення $H_{нас}$. Також, до спільних важливих особливостей належить той

факт, що цей перехід має плавний, монотонний характер.

Результати експериментальних досліджень, представлені у роботах [5,6], отримані для плівок чистого та заміщеного $\text{BaFe}_{12}\text{O}_{19}$, вирощених із застосуванням методів рідкофазної епітаксії і осадження лазерним екстрагуванням з перемінних цілей. Дані рентгеноструктурного аналізу усіх гексаферитових зразків, показали високу ступінь кристалографічної впорядкованості. В результаті проведених вимірювань статичних кривих перемагнічування у широкому діапазоні H_0 , отримано незначний гістерезис. Однак, авторами робіт [5,6], експериментальні дослідження відповідних частотно-польових залежностей ФМР, зокрема у області з доменною структурою, не були проведені.

Робота [7] присвячена проблемі поглибленого вивчення фізичної суті механізмів перемагнічування в епітаксійних плівках гексафериту барію, використовуючи методику вимірювання спектрів ФМР. Щодо особливості утворення гістерезису у плівках, з вищезгаданим характерним переходом частотно-польової залежності ФМР, робота [7], має особливу наукову цінність. У ній, на прикладі епітаксійної плівки $\text{BaFe}_{12}\text{O}_{19}$ $t=3$ мкм, вирощеної на підкладинці з ГТС, було дано адекватну фізичну інтерпретацію цьому явищу, що не зроблено у відомих нам попередніх роботах, зокрема [2,3]. Суть її полягає у тому, що плівка не представляє собою однозв'язної магнітної області, а може складатися з окремих блоків, перехід у доменний стан в яких відбувається не одночасно, а при різних полях H_{zp} , через незначний розкид параметрів в кожному окремому блоці. Відмітимо, що отримана у насиченні для даної плівки найвужча ширина лінії ФМР $2\Delta H=60$ Е, близька до типових значень для якісних гексаферитових плівок [2,3,5].

В роботі [8] вперше представлені результати експериментальних досліджень петель високочастотного та магнітного гістерезису у платівках барієвого гексафериту виготовлених з об'ємного монокристалічного матеріалу. Отримано хорошу кореляцію основних параметрів гістерезису. У цій роботі з'ясовано, що суттєвою відмінністю гістерезису об'ємних зразків від плівкових є стрибкоподібний перехід з насичення в багатодоменний стан в полі H_{br} . Припускалося, що ці процеси пов'язані з механізмом затримки утворення і росту зародків зворотної намагніченості [4]. Однак, в роботі

[10], для платівки об'ємного монокристалічного $\text{BaFe}_{12}\text{O}_{19}$, спостерігався експериментальний факт коли при певних умовах підмагнічування величина H_{zp} не залежала від магнітної передісторії.

Вплив температури на високочастотний гістерезис монокристалічного гексаферитового зразка експериментально досліджувався у роботі [11]. Тут показано, що при пониженні температури зразка від $T=300$ К до $T=77$ К, параметри високочастотного гістерезису практично не змінюються. Тобто, це означає, що значне пониження температури не призводить до "заморожування" центрів доменуутворення.

Експериментальні результати та їх аналіз

В пропонованій роботі, продовжені експериментальні дослідження високочастотних властивостей процесів доменуутворення в одновісних монокристалічних гексаферитах. В експериментах використані зразки прямокутної форми: магнетоплюмбіту ($\text{PbFe}_{12}\text{O}_{19}$), барієвого та стронцієвого ($\text{SrFe}_{12}\text{O}_{19}$) гексаферитів, виготовлені з об'ємного монокристалічного матеріалу і таким чином, щоб ВЛН була перпендикулярною до площини платівки. Кристали гексаферитів вирощувалися методом кристалізації із розчину в розплаві. Усі експерименти виконані у випадку $H_0 \parallel \text{ВЛН}$, для нерегуляризованої доменної структури (ДС) в початковому стані.

Методом вимірювання петель гістерезису частотно-польових залежностей ФМР, для різних значень максимального постійного поля підмагнічування $H_{0\text{макс}}$ зразка, отримана залежність поля зриву H_{zd} від $H_{0\text{макс}}$.

Гістерезисні петлі вимірювалися циклічним чином при поступовому збільшенні $H_{0\text{макс}}$ від значення поля $H_{0\text{нас}} \approx H_{\text{нас}}$ до $H_{0\text{макс}} = 21$ кЕ. На рис.1. приведений гістерезис частотно-польової залежності ФМР низькочастотної гілки ω_1 , для двох різних значень прикладеного $H_{0\text{макс}}$ зразка $\text{SrFe}_{12}\text{O}_{19}$ ($t = 28$ мкм, $a \times b = 1.4 \times 1.2$ мм², $H_a = 19$ кЕ, $M = 0.370$ кГс), в інтервалі $H_0 = 0-7$ кЕ. Як видно з рис. 2, для даного зразка, починаючи із значення $H_{0\text{макс}1}$, зрив гістерезисної кривої відбувається в полі H_{zd1} . Потім, при подальшому збільшенні поля $H_{0\text{макс}}$ до значення $H_{0\text{макс}2}$, зрив вже не відбувається в H_{zd1} , а в полі H_{zd2} і при подальшому збільшенні $H_{0\text{макс}}$ значення $H_{zd} \approx H_{zd2}$ практично не змінюється. Також необхідно відмітити, що інтервал $H_{0\text{макс}}$ ($\approx 15-16.4$ кЕ), котрий охоплює ділянку з практично незмінним значенням H_{zp} , доволі значний, у порівнянні з

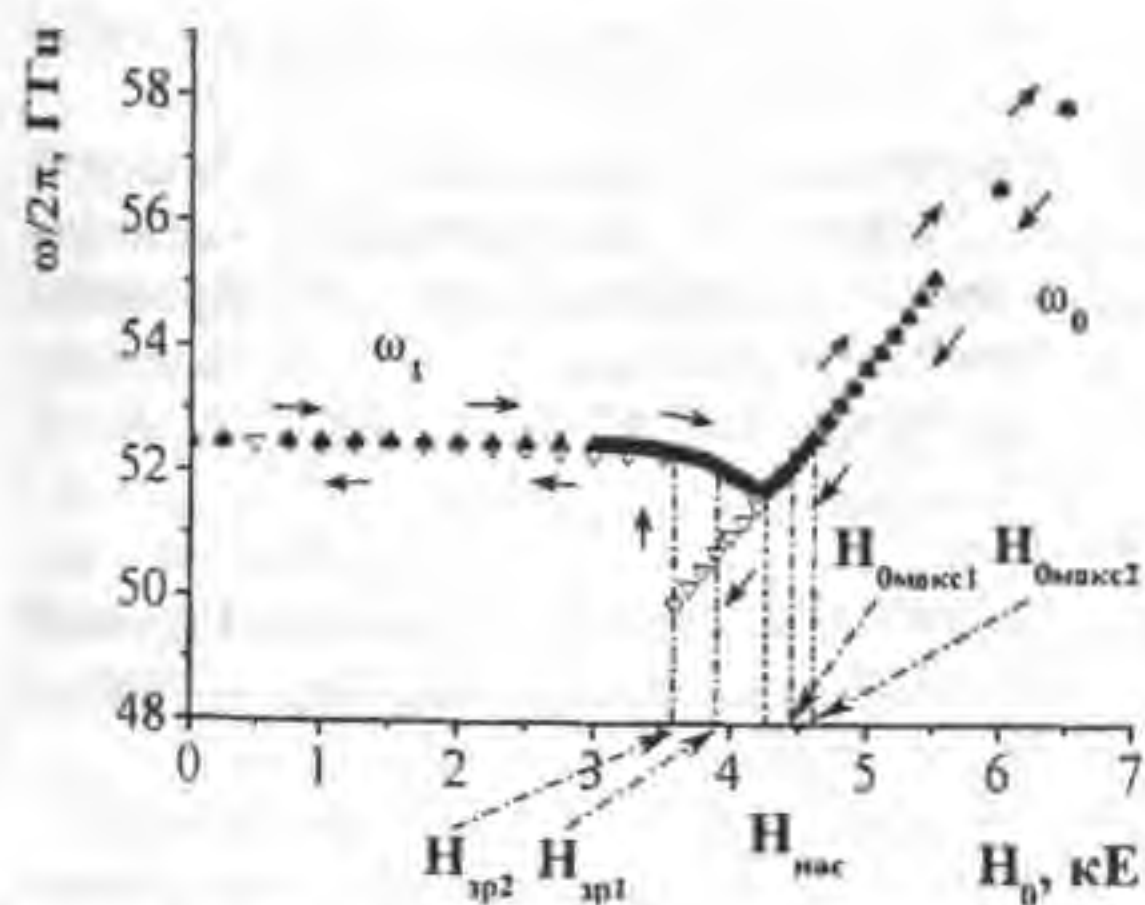


Рис. 1. Гістерезис частотно-польової залежності ФМР в монокристалічному зразку $\text{SrFe}_{12}\text{O}_{19}$. Приведені значення $H_{зр}$ для відповідних величин підмагнічування $H_{0макс}$. ω_1 – низькочастотна гілка в доменній області, ω_0 – в області насичення. $H_{нас}$ – поле насичення.

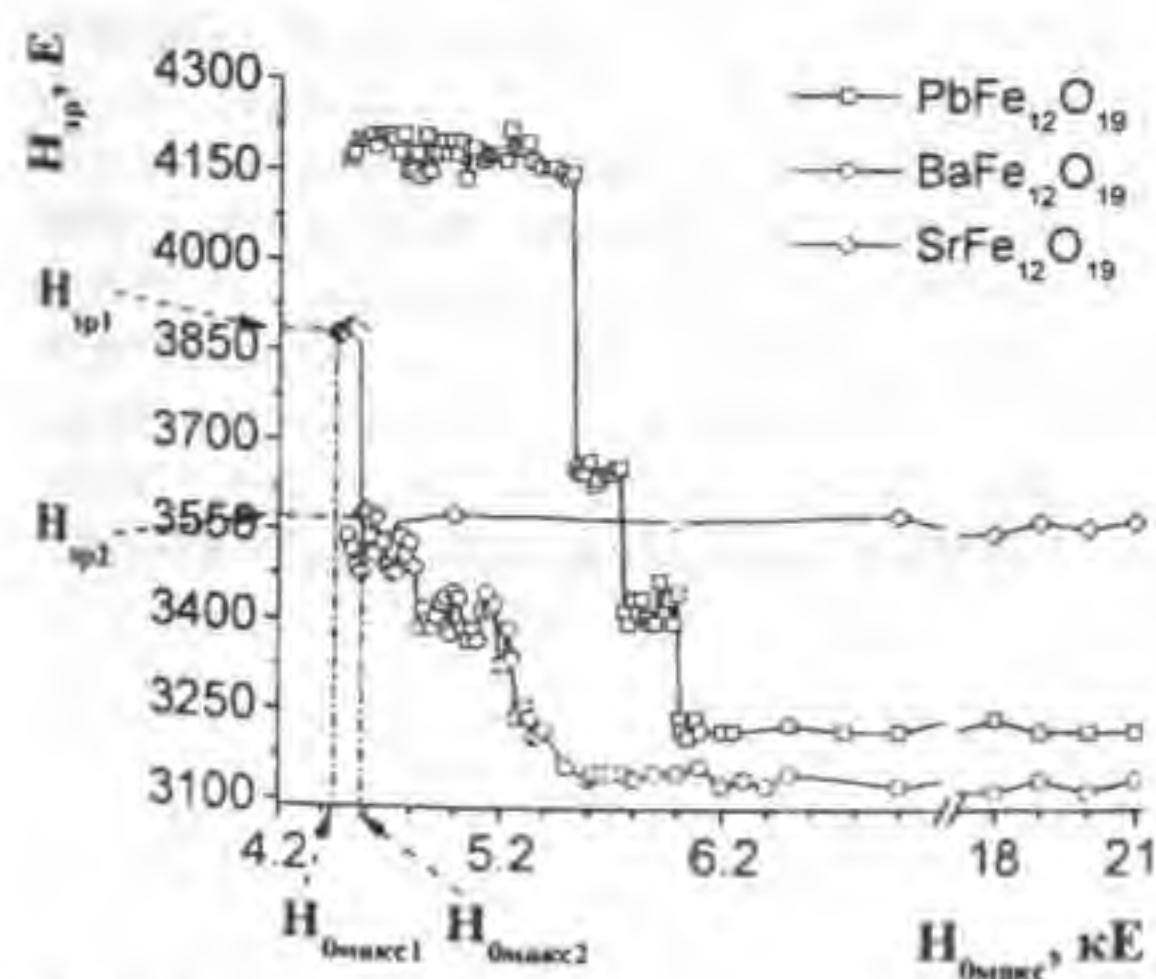


Рис.2. Залежність зміни $H_{зр}$ від прикладеного поля $H_{0макс}$ для прямокутних монокристалічних зразків $\text{PbFe}_{12}\text{O}_{19}$, $\text{BaFe}_{12}\text{O}_{19}$, $\text{SrFe}_{12}\text{O}_{19}$.

інтервалом $H_{0макс}$ ($\approx 0.1-1.6$ кЕ) із стрибкоподібною поведінкою $H_{зр}$. Аналогічні закономірності поведінки при дослідженні залежності $H_{зр}$ від величини $H_{0макс}$ експериментально спостерігалися і в зразках $\text{BaFe}_{12}\text{O}_{19}$ і $\text{PbFe}_{12}\text{O}_{19}$ (рис.2).

Суттєвою особливістю високочастотного гістерезису в приведеному експерименті є різна кількість сходінок для кожного конкретного зразка (див. рис. 2). Виходячи з припущення, що у області виникнення сходінокподібної залежності $H_{зр}$ від $H_{0макс}$, гістерезис пов'язаний з затримкою утворення та росту зародків зворотної намагніченості на дефектах кристалічної ґратки

зразка і залежить від величини H_0 , ми пропонуємо наступний механізм пояснення цього феномену.

Для кожного зразка існує певна кількість зародків зворотної намагніченості (і певний розподіл по площі зразка), умовно кажучи, магнітних неоднорідностей різного типу [4,9]. Мається на увазі, що це можуть бути дефекти кристалічної ґратки – вакансії, прониклі чужорідні атоми, дислокації, також раковини, тріщини і т.д. Кожній вузькій сходінці на кривій $H_{зр}$ від $H_{0макс}$ для конкретного зразка, відповідає один з таких центрів зародкоутворення (рис. 2). Тобто, його участь в процесі утворення доменів зі зворотною намагніченістю має домінуючу роль і завдяки ньому відбувається лавиноподібне розтікання доменів по всьому об'ємі зразка [9]. Отже, число сходінок залежності $H_{зр}$ від $H_{0макс}$ визначається саме кількістю магнітних неоднорідностей конкретного зразка, завдяки котрим відбуваються процеси перемагнічування ще до "насичення" $H_{зр}$ (рис. 2).

Поведінку залежності $H_{зр}$ від $H_{0макс}$ у випадку коли $H_{0макс}$ змінюється в межах від 6 кЕ до 21 кЕ, для зразка $\text{PbFe}_{12}\text{O}_{19}$ (див. рис.2), де дана залежність має "насичення", не вдається пояснити вищезгаданим механізмом. Ми зробили спробу знайти адекватне пояснення цьому явищу, беручи до уваги один дуже важливий факт отриманої експериментальної залежності на рис.2. Зокрема те, що для всіх досліджуваних зразків, значення поля $H_{0макс}$ досягало значень, котрі були навіть більшими значень поля власної кристалографічної анізотропії H_a .

Наслідком дії внутрішнього поля одновісної анізотропії H_a , на кожний окремо взятий магнітний момент в об'ємі одновісного кристалу, є вишикування кожного з них вздовж напрямку ВЛН. Однак, в реальних кристалах, при наявності в них магнітних неоднорідностей, відбувається розорієнтація локалізованих в цій області магнітних моментів відносно ВЛН, тобто встановлення енергетичної рівноваги. Як відомо, відбувається це завдяки фундаментальному закону прямивання будь якої фізичної системи до мінімуму її сумарної енергії. Передача системі магнітних моментів одновісного кристалу енергії зовнішнього поля H_0 , значення якої перевищує H_a , призводить до примушування розорієнтованих моментів орієнтуватися вздовж напрямку ВЛН. Тобто, при даних умовах, вже відбувається "затирання" абсолютно кожної магнітної неоднорідності кристалу.

Наведена фізична інтерпретація експериментальних результатів в інтервалі "насичення" залежності $H_{зр}$, дає змогу вважати, що при зменшенні H_0 , кристал гексафериту перебуває у стані "вигнутих неоднорідностей" до значення $H_{зр}$. І тепер, значення $H_{зр}$, де мають місце процеси переходу в стан з ДС, визначається тільки розмагнічуючими полями та магнітними параметрами кристалу.

Необхідно також зробити ще один важливий висновок з даного дослідження. Як впливає з вищенаведеного, при прикладанні поля $H_{0\max}$, що "насичується", для кожного із зразків: 4.6 кЕ - $\text{SrFe}_{12}\text{O}_{19}$, 5.5 кЕ - $\text{BaFe}_{12}\text{O}_{19}$ та 6 кЕ - $\text{PbFe}_{12}\text{O}_{19}$, вище якого значення $H_{зр}$ не змінюються (рис.2), зразок знаходиться в стані з однорідною намагніченістю у всьому його об'ємі.

Виявлені ефекти можуть слугувати критерієм оцінки якості зразка в процесах технологічного виготовлення гексаферитових матеріалів, а також у застосуваннях функціональної спін-хвильової електроніки мм-діапазону.

Список використаних джерел

1. *V. G. Harris, et al.* Recent advances in processing and applications of microwave ferrites // *J. Magn. Magn. Mater.* – 2009. – 321. – P. 2035-2047.
2. *Wang S.G., Yoon S.D., Vittoria C.* Microwave and magnetic properties of double-sided hexaferrite films on (111) magnesium oxide substrates // *J. Appl. Phys.* – 2002. – 92, № 11. – P. 6728-6728.
3. *Dotsch H., Mateika D., Roschman P., Tolksdorf W.* Growth and properties of epitaxial barium hexaferrite films // *Mat. Res. Bull.* – 1983. – 18. – P. 1209-1216.
4. *Кандаурова Г.С.* Природа магнітного гистерезиса // Соросовский образовательный журнал. – 1997. – № 1. – С. 100-106.
5. *Geiler A. L. et al.* Alternating target laser ablation deposition of high quality barium hexaferrite thin films from barium monoferrite and hematite targets // *J. Appl. Phys.* – 2008. – 103. – P. 07B914-1-3.
6. *Geiler A. L. et al.* Atomic scale design and control of cation distribution in hexagonal ferrites // *Phys. Rev. Lett.* . – 2008. – 101. – P. 067201-1-4.
7. *Зависляк И. В., Костенко В. И., Чамор Т. Г., Чевнюк Л. В.* Ферромагнитный резонанс в эпитаксиальных пленках

одноосных бариовых гексаферритов // *ЖТФ.* – 2005. – 75, № 4. – С. 128-130.

8. *Костенко В. И., Сорочак А. М., Чамор Т. Г., Чевнюк Л. В.* Гистерезис частотно-полевых зависимостей и процессы перемагничивания в бариовом гексаферрите // *ЖТФ.* – 2008. – 78, № 10. – С. 132-134.
9. *Kooy C., Enz U.* Experimental and theoretical study of the domain configuration in thin layers // *Philips Res. Rep.* – 1960. – 15. – P. 7-29.
10. *Костенко В.И., Сорочак А.М., Чамор Т.Г., Чевнюк Л.В.* Особенности магнитного гистерезиса в монокристаллических образцах одноосных бариовых гексаферритов // Сбор. мат. III-й Межд. молод. науч.-техн. конф. "Современные проблемы радиотехники и телекоммуникаций "РТ-2007", – Севастополь. – 2007. – С. 294.
11. *Костенко В. И., Пятница А. И., Сорочак А. М., Чамор Т. Г., Чевнюк Л. В.* Температурные исследования частотно-полевых зависимостей ФМР в одноосном гексаферрите $\text{SrFe}_{11.2}\text{Al}_{0.8}\text{O}_{19}$ // 20-я Международная Крымская конференция "СВЧ-техника и телекоммуникационные технологии" (КрыМиКо'2010), Севастополь: Вебер. – 2010. – С. 783-784.

Надійшла до редколегії 20.10.10

УДК 537.523.5, 536.462

Недибалюк* О.А., аспірант, Ольшевський С.В.,
к.ф.-м.н., с.н.с., Черняк В.Я., д.ф.-м.н., проф.,
Забашта Ю.Ф., д.ф.-м.н., проф., Актан О.Ю.,
к.ф.-м.н., н.с., Свечнікова О.С., к.ф.-м.н., с.н.с.,
Орловська С.Г., к.ф.-м.н., доц., Карімова Ф.Ф.,
к.ф.-м.н., н.с., Шкороподо М.С., аспірант

Стимульоване плазмою спалення парафіну в плазмово-динамічній системі

Досліджено процеси спалення суміші
парафінів за допомогою плазми поперечної дуги.
Виміряно вольт-амперні характеристики
розряду для режимів за наявності та
відсутності парафіну в діапазоні струмів від
200 до 400 мА та потоків повітря від 55 до 275
см³/с. Виміряно емісійні спектри факелу плазми
та полум'я в діапазоні довжин хвиль від 200 до
1000 нм. Обраховано температуру заселення
збуджених обертальних рівнів T_r^* . Виміряно
температуру полум'я при стимульованому
плазмою згоранні стеарину і температуру
полум'я свічки. Виміряно модулі зсуву докозана і
октодекана в процесі плавлення.

Ключові слова: розряд, плазмово-динамічна
система, емісійний спектр, парафін.

Nedybaliuk* O.A., postgraduate student,
Olzewski S.V., Ph. D., Chernyak V.Ya., Dr. Sc.,
Zabashta Yu.F., Dr. Sc., Aktan O.Yu., Ph. D.,
Svechnikova O.S., Ph. D., Orlovska S.G., Ph. D.,
Karimova F.F., Ph. D., Shkoropado M.S.,
postgraduate student

Plasma assisted paraffin combustion in plasma-dynamic system

Process of plasma assisted paraffin mixture
combustion by transversal blowing arc was
investigated. Discharge volt-ampere characteristics
in regimes with and without paraffin at the current
range from 200 to 300 mA and air flow at the range
of 55-275 cm³/s were measured. Emission spectra of
plasma torch and flame at the wavelength range
from 200 to 1100 nm were registered. Excitation
rotational temperature T_r^* was calculated.
Temperatures of flame at plasma assisted stearin
combustion and candle flame were measured. The
shear modulus of the docosane and octodecane
melts was measured.

Key Words: discharge, plasma-dynamic
system, emission spectrum, paraffin.

*E-mail: oadedybaliuk@gmail.com

Статтю представив д.ф.-м.н. Анісімов І.О.

Вступ

Парафіни та циклопарафіни є основними
компонентами багатьох груп палив нафтового
походження та синтетичних. Сьогодні самі по
собі парафіни та їх суміші розглядаються як
перспективні альтернативні палива, що пов'язано
з їх високою калорійністю та хімічною
інертністю [3]. Висока калорійність парафінів
визначається тим, що вони відносяться до
вуглеводнів з простими зв'язками C_nH_{2n+2} , в яких
найбільше співвідношення водню до вуглецю
H/C. Але для спалення парафінів, у зв'язку з їх
хімічною інертністю, необхідна додаткова
активація горіння. Як відомо, найбільш
ефективним засобом активації процесів спалення
є плазмове стимулювання горіння. Але для
парафінових палив даний підхід практично не
досліджений.

В даній роботі наведено результати
досліджень стимулювання процесів спалення
суміші парафінів за допомогою плазми

поперечної дуги [5]. В якості модельної суміші
парафінів використано стеарин.

Експериментальна установка

Для роботи використано промисловий
стеарин з якого виготовляють свічки. Схема
плазмово-динамічної системи по покращенню
горіння стеарину зображена на рис. 1а. Вона
складається з головної сталевий камери (1) в яку
крізь діелектрики (3) вводяться два мідні
електроди (2) діаметром 4 мм. Розряд
запалюється між мідними електродами. Напруга
між електродами подається за допомогою блока
живлення (БЖ), який видає напругу в діапазоні
від 0 до 10 кВ. Перпендикулярно до електродів
подається потік повітря (рис. 1а), що роздуває
дугу, формуючи плазмовий факел (4). Потік
повітря формувався компресором і вимірювався
за допомогою ротаметра. Плазмовий факел
оточений металевою сіткою (5) у формі циліндра
діаметром 1 см і довжиною 4 см. Сітка кріпиться

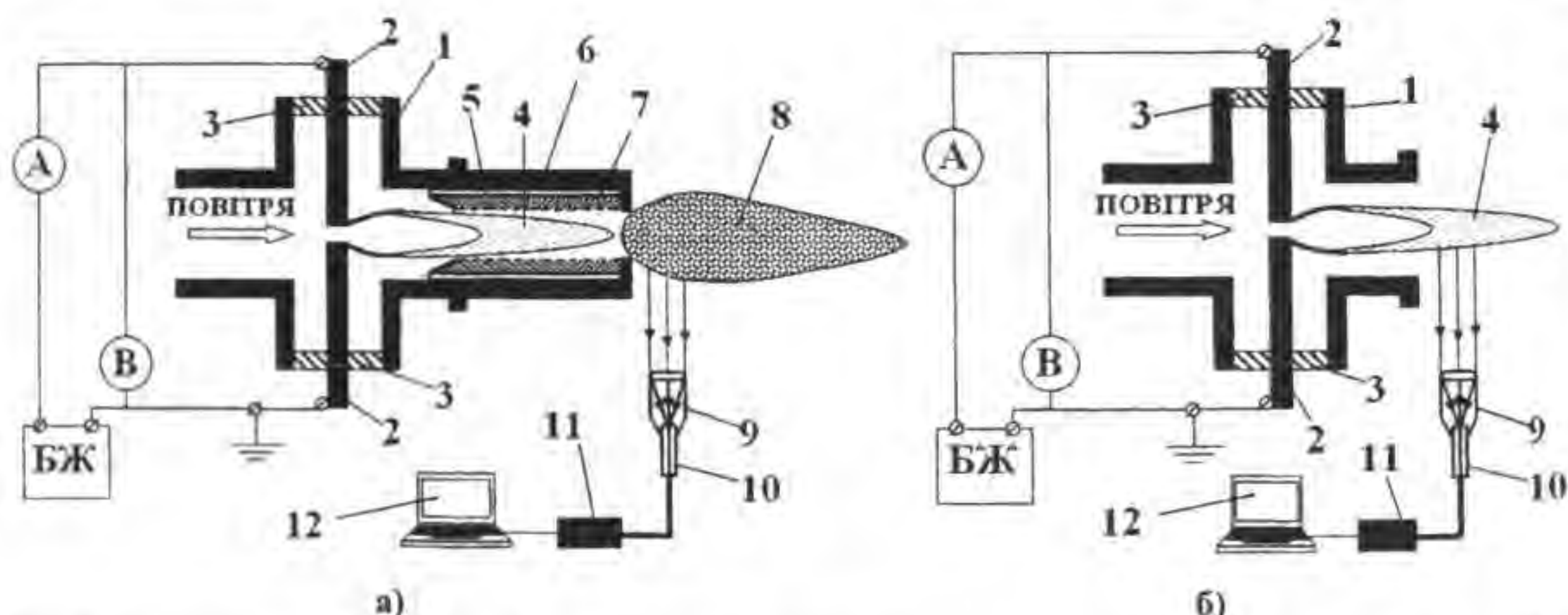


Рис. 1 Схема плазмово-динамічної системи: при наявності стеарину – а) та відсутності стеарину – б)

до сталевій циліндричній капсулі (6). Проміжок між сіткою (5) та сталевією капсулою (6) заповнений стеарином (7). Вага стеарину, що завантажувався в систему була 3 грами. При запалюванні розряду на виході із сталевієї капсулі (6) утворюється полум'я (8). Оптичні дослідження полум'я проводилися за допомогою оптичної системи, що зображена на рис. 1а. Вона складається з освітлювальної системи (9), світловода (10), спектрометра *S-150-2-3648 USB* (11) та комп'ютера (12). Спектрометр дозволяє вимірювати емісійні спектри в діапазоні довжин хвиль від 200 до 1100 нм. За допомогою комп'ютера відбувалася обробка даних, що надходять із спектрометра, та здійснювалось керування вимірюваннями. Полум'я по всій довжині мало однаковий колір, тому для дослідження вибиралася область стабільного горіння. Ця область знаходилася при виході із металевієї капсули. Освітлювальна система була орієнтована на точку, що знаходиться на осі полум'я на відстані 6 см від осі мідних електродів. Відстань від осі сталевієї циліндричній капсулі до освітлювальної системи складала 10 см, причому, вони були розташовані в перпендикулярних площинах.

Для дослідження плазмового факелу використовувалася схема яка зображена на рис. 1б. При цьому від системи, що зображена на рис. 1а, від'єднувалася сталевіа капсула (6), що містить стеарин і металеву сітку. Освітлювальна система була розташована на відстані 9 см від зрізу металевієї камери (1), перпендикулярно до її осі і орієнтована на точку осі плазмового факелу, що знаходилася на відстані 3 см від осі електродів (рис. 1б).

На рис. 2 зображена схема дослідження полум'я стеаринової свічки (1). Полум'я свічки має три характерні області, які мають різну температуру [2]. Найменшу температуру має область поблизу гніта (а), в якій відбувається випаровування стеарину. Вона є прозорою областю. Далі йде область (б) в якій відбувається руйнування парафінів і часткове їх згорання. Вона має оранжевий колір з темним відтінком. Температура в ній вища ніж у попередній області. Третя область (в), в якій відбувається повне згорання, має жовто-біле забарвлення. Вона має найвищу температуру.

Спектроскопічні дослідження проводилися за допомогою системи яка зображена на рис. 2. Вона складається з освітлювальної системи (2), світловода (3), спектрометра *S-150-2-3648 USB* (4) та комп'ютера (5). Освітлювальна система орієнтована на область полум'я з найбільшою

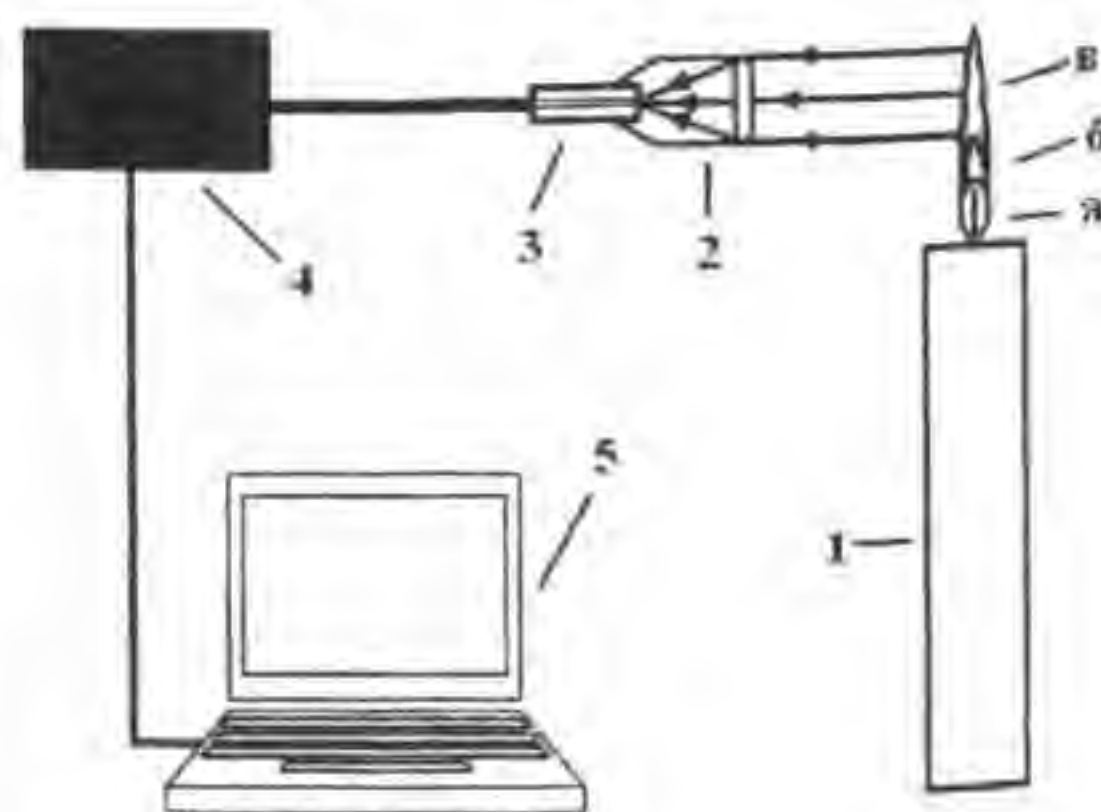


Рис. 2 Схема дослідження полум'я свічки

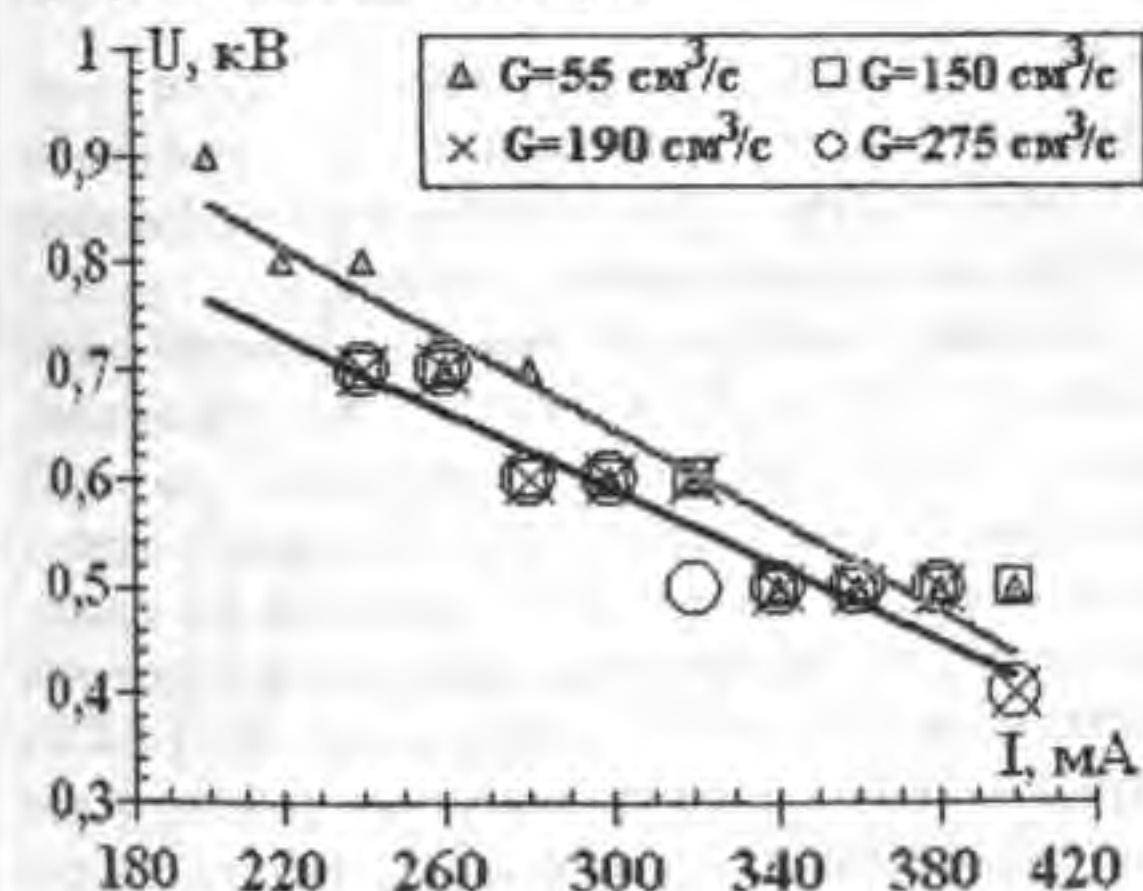


Рис. 3 Вольт-амперні характеристики розряду за наявності стеарину при різних потоках повітря

температурою, перпендикулярно до осі полум'я свічки. Відстань від джерела випромінювання до освітлювальної системи складає 10 см.

Результати та їх обговорення

Виміряно вольт-амперні характеристики розряду із стеарином при різних потоках повітря (рис. 3). Вольт-амперні характеристики мали спадаючий характер із зростанням струму. Даний розряд схожий до розряду дугового типу. Лінії трендів для мінімального потоку повітря $G = 55 \text{ cm}^3/\text{c}$ (сірий) та для максимального потоку повітря $G = 275 \text{ cm}^3/\text{c}$ (чорний) мають майже однаковий кут нахилу (рис. 3). Це можна пояснити тим, що розряд в діапазоні струмів від 200 до 400 мА мало залежить від величини потоку повітря.

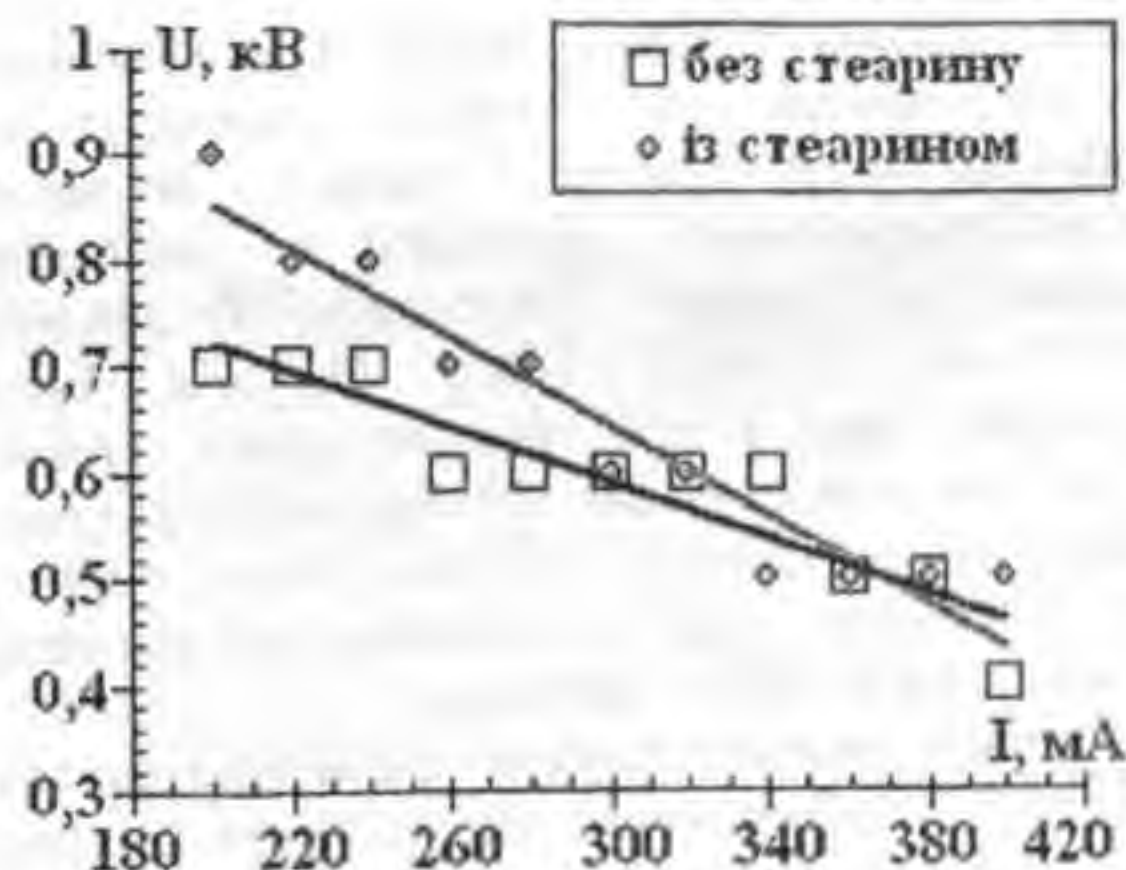


Рис. 4 Вольт-амперні характеристики розряду за наявності та відсутності стеарину. Потік повітря $G = 55 \text{ cm}^3/\text{c}$

На рис. 4 представлені вольт-амперні характеристики розряду за наявності та відсутності стеарину при потоці повітря $G = 55 \text{ cm}^3/\text{c}$. Вольт-амперна характеристика в діапазоні струмів від 200 до 400 мА має крутіший спад при наявності стеарину ніж при його відсутності. Це може бути пов'язано із тим, що пари стеарину призводять до утруднення горіння розряду. Мінімальний струм при якому відбувалося горіння розряду зростає із збільшенням потоку повітря. Максимальний струм залишався однаковим.

Емісійні спектри вимірювалися в режимі стабільного горіння при $G = 55 \text{ cm}^3/\text{c}$, $I = 300 \text{ mA}$, $U = 0,6 \text{ kV}$. Потужність яка вкладалась в розряд на підтримку плазми при даному режимі роботи складала $P = 180 \text{ Вт}$. Час згорання стеарину для

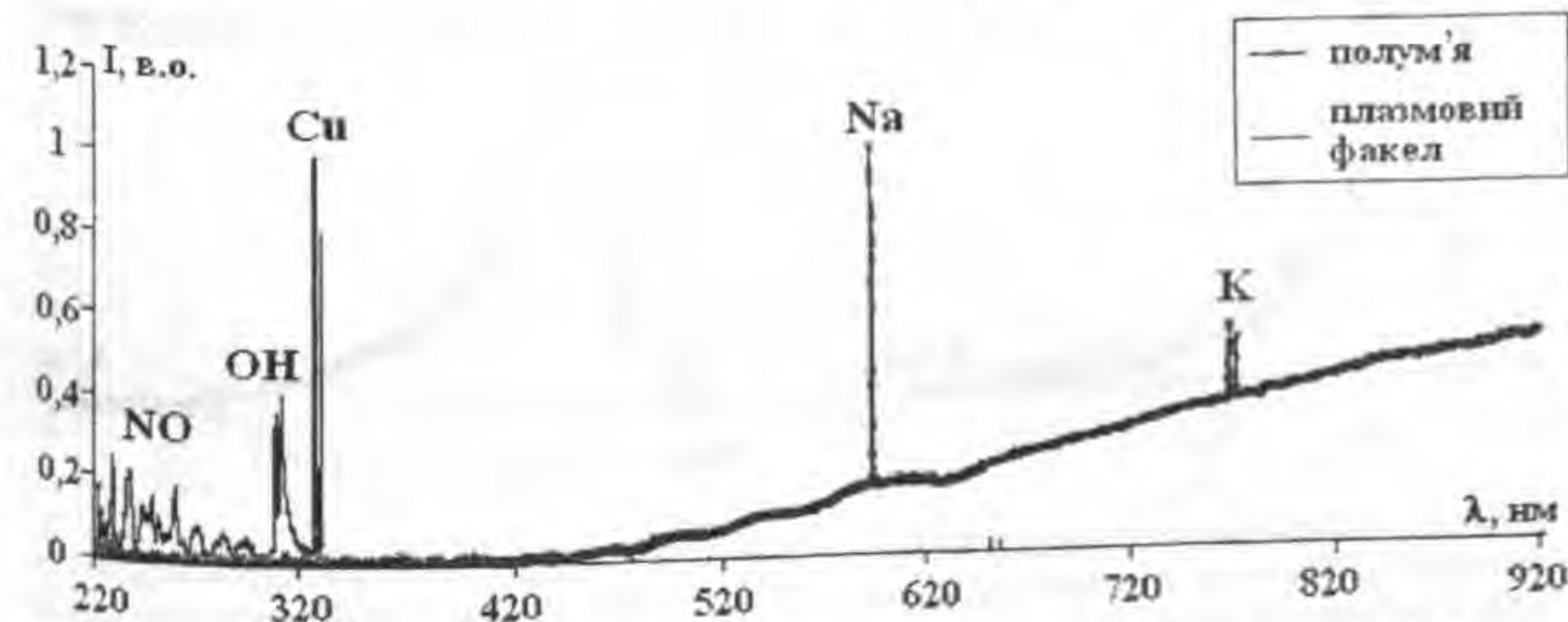


Рис. 5 Емісійні спектри плазмового факелу та полум'я при струмі 300 мА, напрузі 0,6 кВ, потоці повітря $55 \text{ cm}^3/\text{c}$

даного режиму складав 15 секунд. При підвищенні потоку повітря час згорання зменшувався. Співвідношення енергії повного згорання стеарину до енергії затраченої на генерацію плазми для режиму $G = 55 \text{ см}^3/\text{с}$, $I = 300 \text{ мА}$, $U = 0,6 \text{ кВ}$ складало 100/1. Полум'я було присутнє лише в тому випадку, коли горіння стимулювалося плазмою. При вимиканні розряду факел полум'я зникав, величина потоку повітря не змінювалась. При подальшому запалюванні розряду, факел знову з'являвся.

Типові емісійні спектри плазмового факелу та полум'я зображено на рис. 5. Обидва спектри нормовані на свій максимум: плазмового факелу на довжині хвилі $\lambda_{\text{H}} = 588,99 \text{ нм}$, полум'я на довжині хвилі $\lambda_{\text{H}} = 324,75 \text{ нм}$. Емісійний спектр плазмового факелу містить смуги монооксиду азоту NO та гідроксиду OH (A-X), лінії міді Cu (324,75 нм; 327,39 нм). Емісійний спектр полум'я є неперервним, зростаючим із збільшенням довжини хвилі, з накладанням резонансних ліній натрію Na (588,99 нм, 589,59 нм), калію K (766,49 нм, 769,89 нм) та смуги гідроксиду OH (A-X). Оскільки ми використовуємо промисловий стеарин, то там можуть бути присутні різні домішки, такі як K та Na. Емісійний спектр полум'я подібний до спектру випромінювання абсолютно чорного тіла.

Емісійний спектр полум'я свічки (рис. 9) також є неперервним, зростаючим із збільшенням довжини хвилі, з накладеними на нього резонансними лініями калію K (766,49 нм,

769,89 нм). Однак вони мають дуже малу інтенсивність у порівнянні із емісійними спектрами полум'я (рис. 9). Горіння відбувалося в повітрі при атмосферному тиску.

На рис. 6 зображено емісійні спектри смуги гідроксиду OH (A-X) в полум'ї та плазмовому факелі. Дані спектри нормовані на свій максимум. Хвіст смуги OH в полум'ї лежить нижче ніж хвіст смуги OH в плазмовому факелі. Це означає, що температура заселення збуджених обертальних рівнів T_r^* гідроксиду OH (A-X) в полум'ї менша ніж в плазмовому факелі. Адже саме T_r^* , найбільше впливає на вигляд хвоста смуги гідроксиду.

Для визначення температури заселення збуджених обертальних рівнів T_r^* виконувались розрахунки емісійних спектрів смуги гідроксиду OH (A-X) за допомогою програми SPECAR і порівнювалися з експериментально виміряними. Для порівняння використовувався хвіст смуги гідроксиду OH (A-X) в діапазоні довжин хвиль від 310 до 320 нм. Температури заселення збуджених коливальних T_v^* та електронних T_e^* рівнів задавалися рівними температурі заселення збуджених обертальних рівнів T_r^* . Емісійний спектр плазмового факелу та розрахований за допомогою програми SPECAR [6] спектр смуги гідроксиду OH (A-X) при $T_r^* = 2700 \text{ К}$ зображено на рис. 7. Отже температура заселення збуджених обертальних рівнів в плазмовому факелі рівна $T_r^* = 2700 \text{ К}$.

Емісійні спектри полум'я та розрахованої

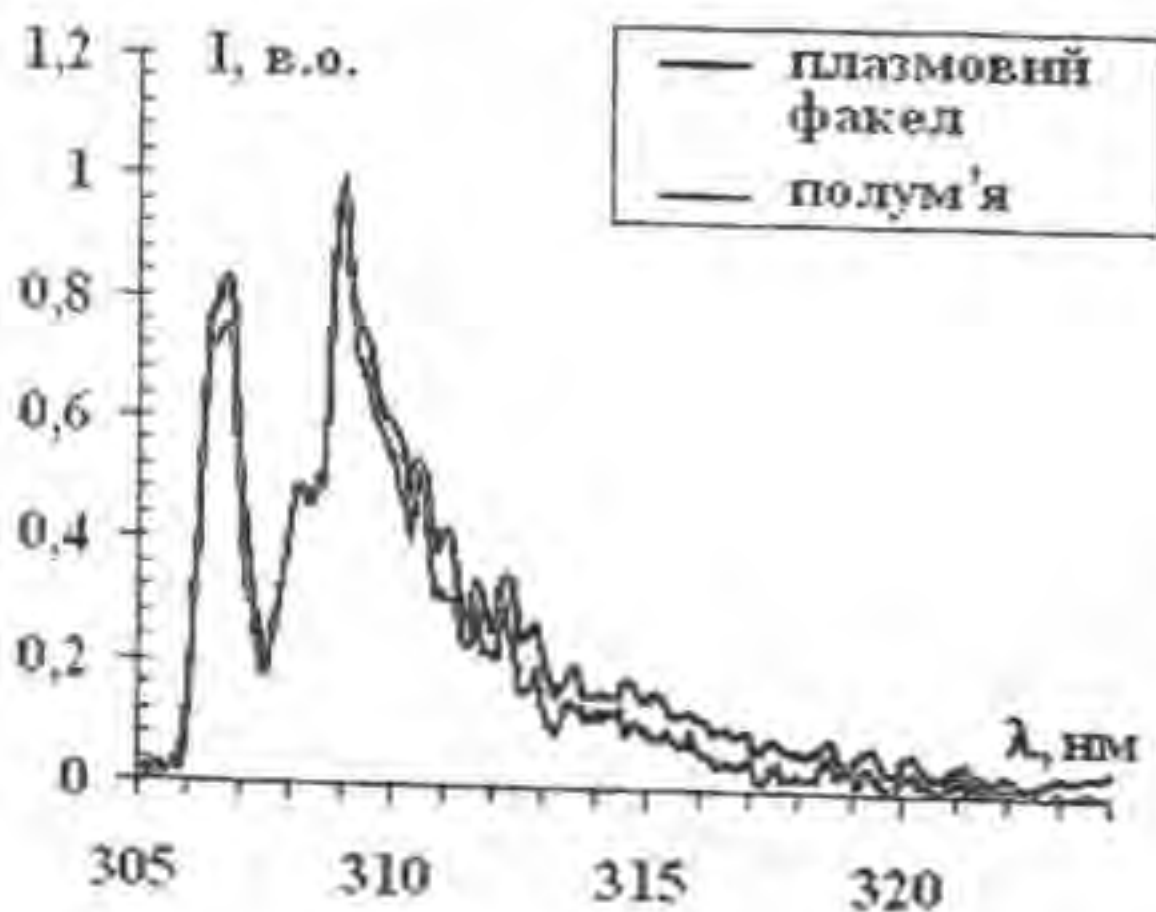


Рис. 6 Емісійні спектри смуги гідроксиду OH (A-X) в плазмовому факелі та в полум'ї, виміряні при потоці повітря $55 \text{ см}^3/\text{с}$, струмі 300 мА, напрузі 0,6 кВ

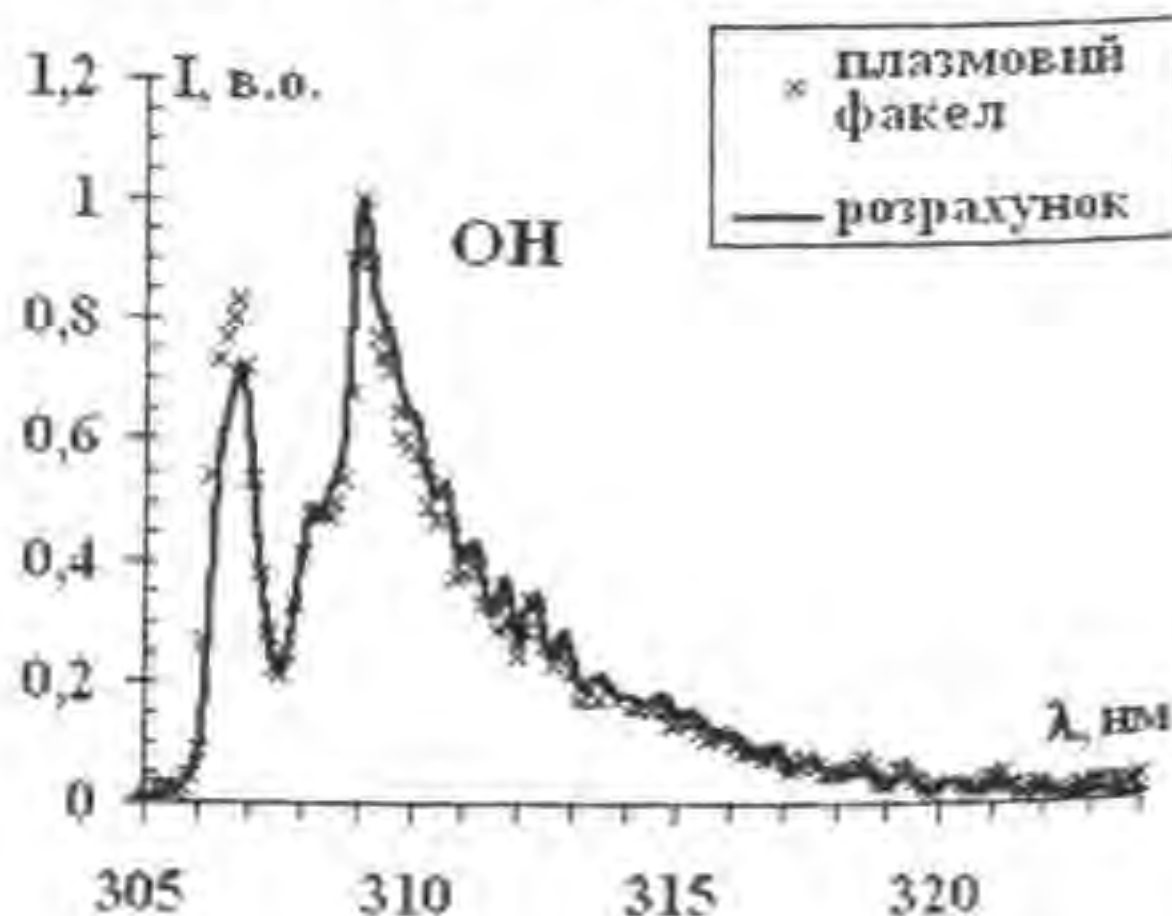


Рис. 7 Емісійні спектри смуги гідроксиду OH (A-X) в плазмовому факелі, виміряного при потоці повітря $55 \text{ см}^3/\text{с}$, струмі 300 мА, напрузі 0,6 кВ, та розрахований за допомогою програми SPECAR при $T_r^* = 2700 \text{ К}$

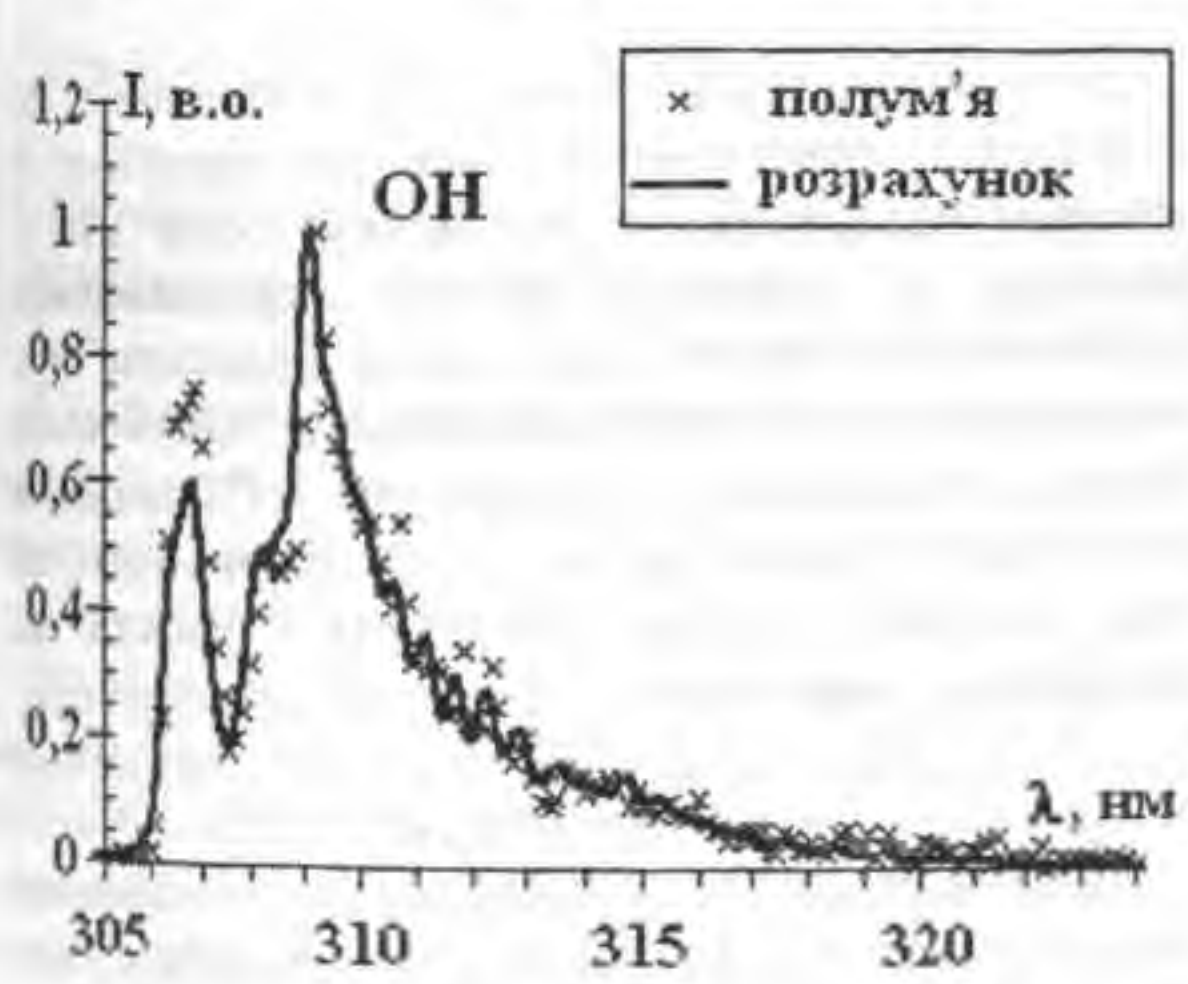
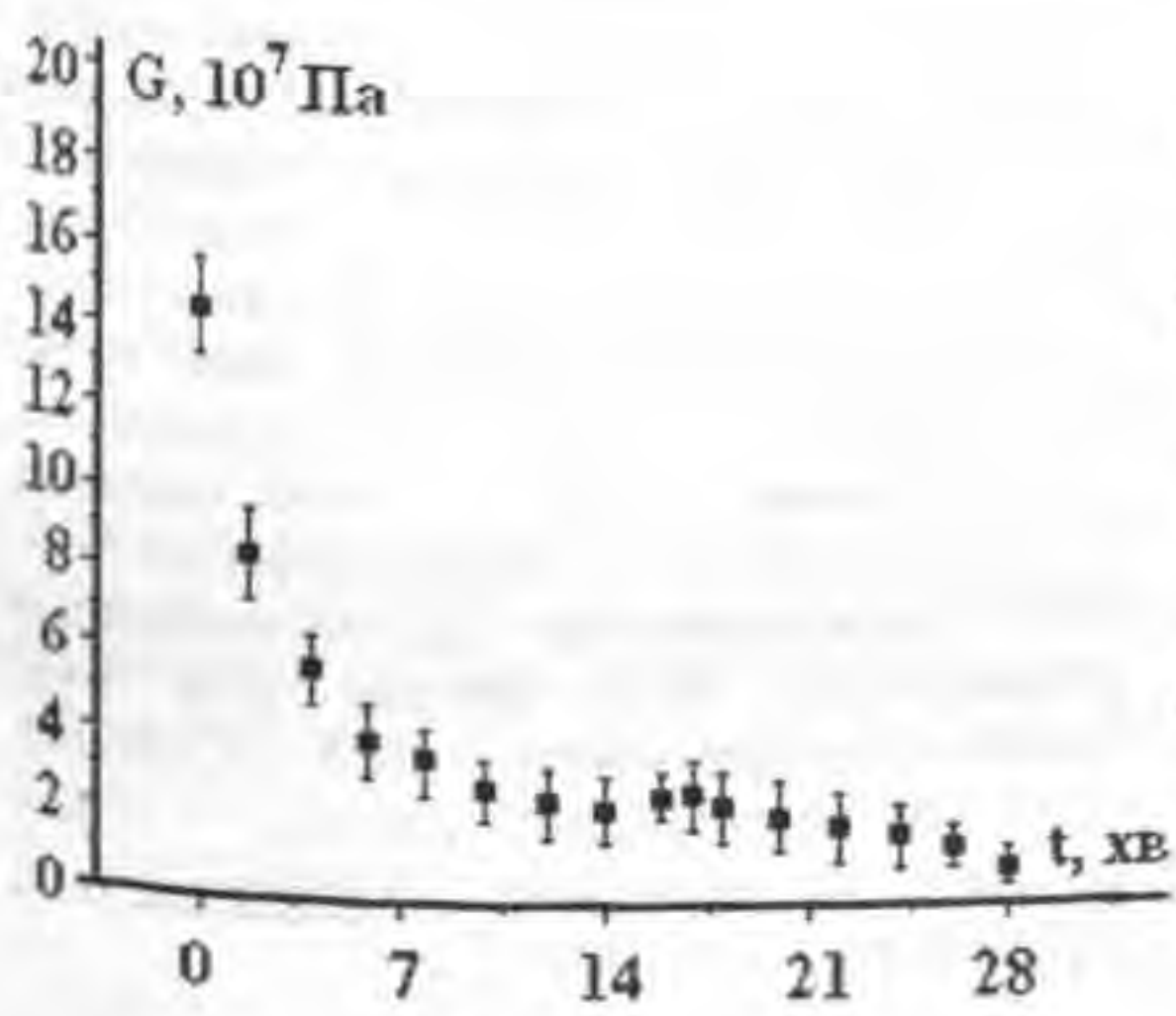


Рис. 8 Емісійні спектри полум'я, виміряного при потоці повітря 55 см³/с, струмі 300 мА, напрузі 0,6 кВ, та смуги ОН (А-Х), розрахованої за допомогою програми SPECAIR при $T_r^*=2300$ К

смуги ОН (А-Х) при $T_r^*=2300$ К зображені на рис. 8. Хвости обох спектрів досить добре співпадають. Отже температура заселення збуджених обертальних рівнів в полум'ї рівна $T_r^* = 2300$ К.

Виходячи з неперервного характеру емісійних спектрів полум'я та полум'я свічки, проводилось порівняння експериментальних спектрів з розрахованими спектрами випромінювання абсолютно чорного тіла. Розрахунки проводились за допомогою формули Планка (1):



а)

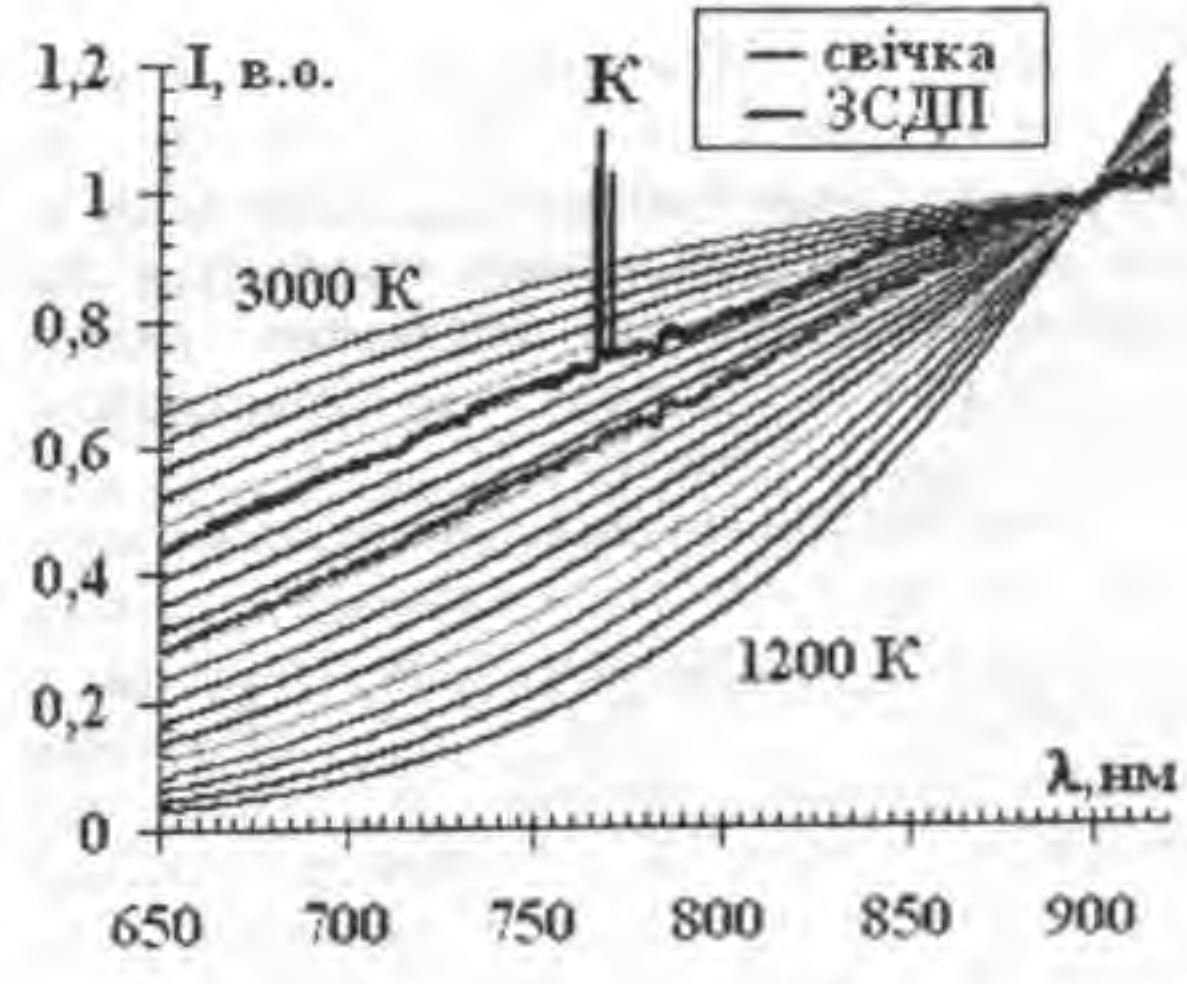
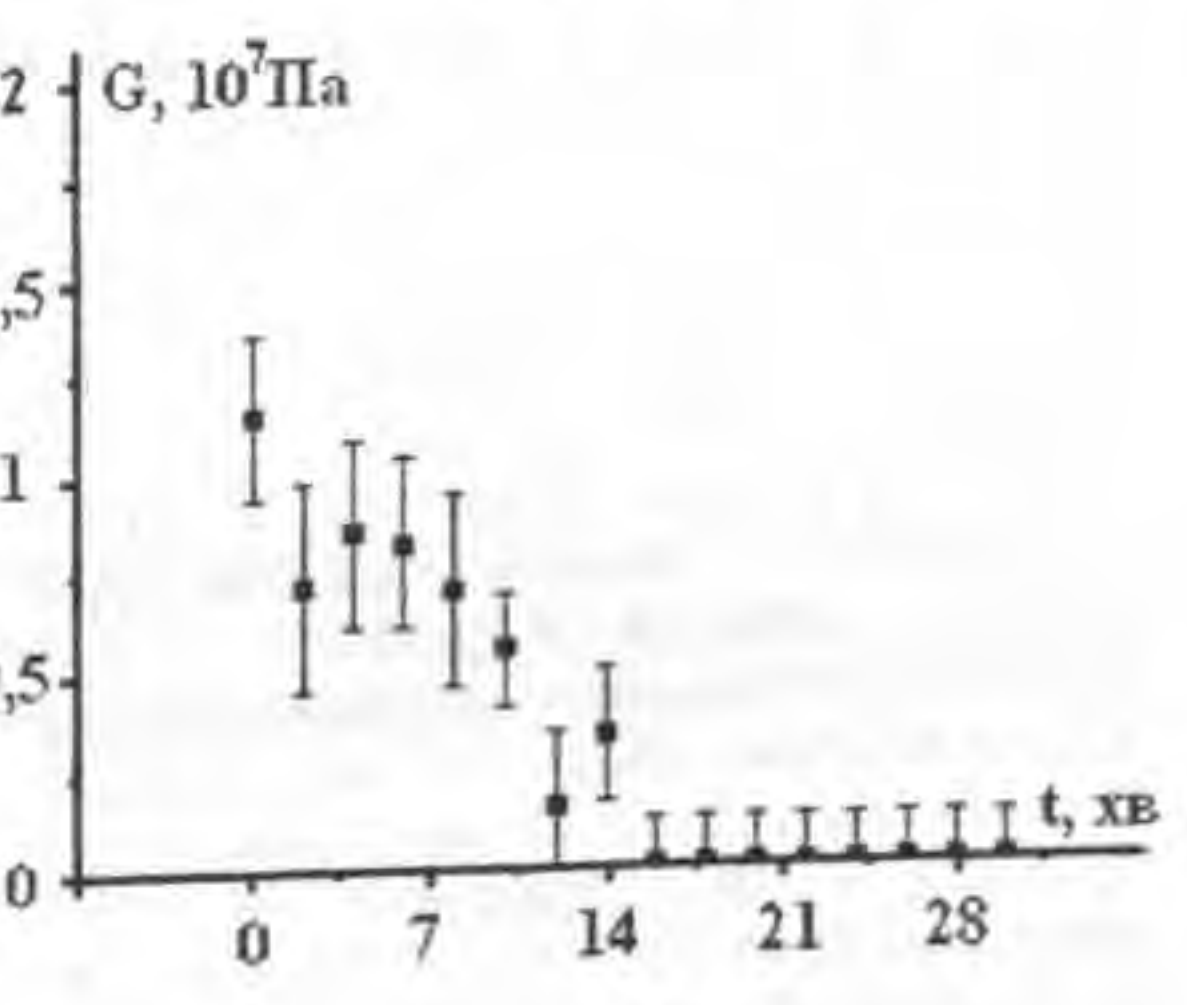


Рис. 9 Емісійні спектри полум'я свічки, полум'я ЗСДП при потоці повітря 55 см³/с, струмі 300 мА, напрузі 0,6 кВ, та розраховані спектри випромінювання абсолютно чорного тіла

$$I = \frac{hc^2}{\lambda^5} \left(\exp\left(\frac{hc}{kT\lambda}\right) - 1 \right)^{-1} \tag{1}$$

На рис. 9 зображено розраховану сітку спектрів випромінювання абсолютно чорного тіла з кроком 100 К. Розрахунки проводилися в діапазоні температур від 1200 К до 3000 К. На даному рисунку, також, зображені емісійні спектри полум'я свічки та полум'я при згоранні стеарину за допомогою плазми (ЗСДП). Емісійний спектр полум'я свічки співпадає з розрахованим для температури $T^* = 2100$ К. Емісійний спектр полум'я ЗСДП співпадає з розрахованим по формулі (1) для температури



б)

Рис. 10 Температурні залежності модуля зсуву докозана C₂₂H₄₆ – а) та октодекана C₁₈H₃₈ – б)

$T = 2500$ К. Температура полум'я $T_{\text{зсдп}} = 2500 \pm 100$ К на 400 К більша за температуру полум'я свічки $T_{\text{свчки}} = 2100 \pm 100$ К. Слід зазначити, що оберտальна температура при згоранні стеарину за допомогою плазми $T_r^* = 2300 \pm 500$ К близька до температури полум'я $T_{\text{зсдп}} = 2500 \pm 100$ К.

Також експериментально визначались модулі зсуву докозана і октодекана в процесі плавлення [4]. Кювета з парафіном була поміщена в термокамеру. Для діагностики використовувалась система із крутильним маятником. Сигнал з крутильного маятника надходив до АЦП і далі передавався на комп'ютер. Часові залежності модуля зсуву докозана $\text{C}_{22}\text{H}_{46}$ і октодекана $\text{C}_{18}\text{H}_{38}$ в процесі плавлення зображені на рисунку 10. Проводилися візуальні спостереження стану парафіну в кюветі. В результаті цих спостережень було встановлено, що докозан і октодекан через 28 хвилин з моменту початку плавлення мають структуру розплаву.

Як видно з рис. 10а модуль зсуву розплаву докозана значно відрізняється від нуля. Також з рис. 10б видно, що модуль зсуву октодекана рівний нулю. Таким чином розплав *n*-парафіну, молекула якого містить 22 атомних вуглецевих групи являється в'язко-пружним тілом. На відміну від докозана, розплав октодекана, молекула якого містить 18 атомних вуглецевих груп є в'язкою рідиною [1].

Відповідно приходимо до висновку, що збільшення ефективності горіння парафінів із довгими ланцюгами є наслідком того, що їх розплави є в'язко-пружними середовищами. Існування такого зв'язку можна пояснити тим, що для в'язко-пружного середовища відбувається більш інтенсивне розпилення. Підвищення інтенсивності цього процесу є наслідком того, що шар розплаву втрачає механічну стійкість під дією потоку окисника.

Висновки

Дослідження процесу спалення стимульованого плазмою поперечної дуги суміші стеарин-повітря виявили, що:

1. Пари стеарину утруднюють горіння розряду.
2. Для режиму стійкого горіння емісійний спектр полум'я є неперервним і відповідає спектру випромінювання абсолютно чорного тіла.
3. Температура заселення збуджених обертальних рівнів гідроксиду $\text{OH}(A-X)$ ($T_r^* = 2300$ К) близька до температури полум'я ($T_{\text{зсдп}} = 2500 \pm 100$ К, що виміряна за допомогою розрахованих кривих за формулою Планка).
4. Температура заселення збуджених обертальних рівнів гідроксиду $\text{OH}(A-X)$ в полум'ї ($T_r^* = 2300$ К) менша ніж в плазмовому факелі ($T_r^* = 2700$ К).

Список використаних джерел

1. Ландау Л.Д., Лифшиц Е.М. Гидродинамика / Л.Д. Ландау, Е.М. Лифшиц. - Москва: Наука, 1988. - 736 с.
2. Тидеман Б.Г., Сциборский Д.Б. Химия горения: [изд. 3-е, перераб. и доп.] / Б.Г. Тидеман, Д.Б. Сцибрицкий. - Л.: Гостранс издат, 1935. - 356 с.
3. Dornheim M.A. Ideal hybrid fuel is wax / M.A. Dornheim // Aviation week and space technology. - 2003. - 3. - P. 52-54.
4. Wunderlich B. Reversible Crystallization and the Rigid-Amorphous Phase in Semicrystalline Macromolecules / B. Wunderlich // Prog. Polym. Sci. - 2003. - № 28. - P. 383-450.
5. Недыбалюк О.А., Черняк В.Я., Ольшевский С.В., Лиситченко Т.Е., Булавин Л.А., Забашта Ю.Ф., Актан Е.Ю., Лендел В.В., Свечникова О.С., Орловская С.Г., Каримова Ф.Ф., Шкоропато М.С. Сжигание парафина при помощи плазмы / О.А. Недыбалюк, В.Я. Черняк, С.В. Ольшевский, Т.Е. Лиситченко, Л.А. Булавин, Ю.Ф. Забашта, Е.Ю. Актан, В.В. Лендел, О.С. Свечникова, С.Г. Орловская, Ф.Ф. Каримова, М.С. Шкоропато // Энергоэффективность-2010: международная научно-техническая конференция, 19-20 October 2010, Kyiv, Ukraine: abstracts. - Kyiv. - 2010. - P. 188-190.
6. SPECAR [електронний ресурс]. - Режим доступу до програми: <http://www.specair-radiation.net>

Надійшла до редколегії 25.10.2010

УДК 535.345.67

*Первак Ю.О., д.ф.-м.н., проф., Федоров В.В.,
аспірант

Спектральні властивості багатошарових структур з трьома півхвильовими резонаторами

Досліджено спектральні властивості чвертьхвильових інтерференційних фільтрів з трьома півхвильовими прошарками. Фільтри складаються з шарів оксиду кремнію (SiO_2) та оксиду титану (TiO_2). Визначено параметри багатошарових структур з контрастними смугами прозорості, які розщеплюються на три окремі підсмуги.

Ключові слова: багатошарова тонкоплівкова структура, смуговий інтерференційний фільтр

*E-mail: yupervak@univ.kiev.ua

Статтю представив д.ф.-м.н. Скришевський В.А.

При вирішенні багатьох проблем створення оптичних телекомунікаційних систем виникає задача частотної селекції. Для її розв'язання використовують різноманітні фільтри, пошук структури яких є типовою оберненою спектральною задачею [1-5]. При цьому особливо складною виявляється задача пошуку структур з високою контрастністю селекції. Найчастіше контрастні смугові фільтри створюють на базі чверть хвильових стоп з певною кількістю півхвильових прошарків. Вони отримали назву багато півхвильових фільтрів [6-9].

Всі можливі структури багато півхвильових фільтрів можна описати виразом [6-9]

$$S w(q)^m v, \quad (1)$$

де S позначає підкладку, на якій розташовано багатошарову структуру, яка умовно може бути представлена як система з трьох груп шарів w , q і v . При цьому група q повторюється m разів. Усе різноманіття півхвильових структур вкладається в наступні чотири підгрупи.

Перша:

$$w = (AB)^{l+1}, q = (BA)^{k+1} B, v = (BA)^{l+1}, \quad (2)$$

$$\begin{cases} s = m+1 & \text{при } l \geq 0, \\ s = m-1 & \text{при } l = -1. \end{cases}$$

Yu.O. Pervak, Dr. Sci., V.V. Fedorov, post grad. stud.

The synthesis of multilayer structures with high time and spatial dispersion

Spectral properties of quarter-wave interference filters with three half-wave layers were investigated. Filters consist of layers of silicon oxide (SiO_2) and titan oxide (TiO_2). It was found parameters of multilayer structures with the contrasting transparency bands, which fission on three separate sub-bands.

Key words: multilayer thin-film structure, band pass interference filter

Друга:

$$w = (BA)^l B, q = (BA)^{k+1} B, v = (BA)^l B, \quad (3)$$

$$s = m+1$$

Третя:

$$w = (AB)^{l+1}, q = (BA)^{k+1} (AB)^{k+1}, v = (BA)^{l+1}, \quad (4)$$

$$\begin{cases} s = 2m+1 & \text{при } l \geq 0, \\ s = 2m-1 & \text{при } l = -1. \end{cases}$$

Четверта:

$$w = (BA)^l B, q = (BA)^{k+1} (AB)^{k+1}, v = (BA)^l B, \quad (5)$$

$$s = 2m+1$$

де A і B – чвертьхвильові шари відносно довжини хвилі λ_0 з показниками заломлення n_A і n_B ; l і k – цілі числа, причому $k \geq 0$ і $l \geq -1$ для (2), (4), $l \geq 0$ для (3), (5); s – число півхвильових шарів в системі, S – підкладка з показником заломлення n_m .

Для того, щоб в аналітичному вигляді було можливо описати основні властивості таких систем було введемо декілька нових понять. Перше – функція середнього геометричного від обвідних сімейства кривих коефіцієнта пропускання – $T_{\text{серед}}(\delta)$. Друге – середня прозорість смуги

$D = (T_{\text{geom}})_{\delta=\pi/2}$. Тут δ позначає фазову товщину.

На довжині хвилі λ_0 фазова товщина шару $\delta = \pi/2$. Було отримано наближенні аналітичні вирази, що зв'язують введені поняття і півширину фільтру з параметрами шарів та структурою фільтру.

Аналіз отриманих виразів показав, що при фіксованому значенні n_H/n_L більшим значенням відносної півширини $\Delta\lambda_h/\lambda_0$ відповідають менші значення k , і навпаки. Для потрібних півхвильових систем вибір значень k обмежений діапазоном 2÷8 і тому для $\Delta\lambda_h/\lambda_0$ теж отримується обмежений діапазон. Цей діапазон для потрібних півхвильових систем дорівнює $\Delta\lambda_h/\lambda_0 \approx 0,3 \cdot 10^{-3} \div 0,2$. В свою чергу обмежений вибір $m=1, m=2$ для потрібних півхвильових систем при низьких значеннях n_H/n_L , які відповідають великим значенням $\Delta\lambda_h/\lambda_0$ при фіксованому k , визначають величини загального числа шарів в системі N , недостатньо високі для отримання низького рівня фону і високої контрастності. Таким чином, обмежений набір значень параметрів s, m, l, k або вводить відповідні обмеження на діапазон значень ширини смуги, при задаванні яких можуть бути отримані достатньо контрастні фільтри на класах потрібних півхвильових систем, або роблять взагалі неможливими конструкції потрібних півхвильових систем.

Однак такий підхід до пошуку параметрів структури з бажаними спектральними характеристиками не дозволяє визначити структури, смуга прозорості яких розщеплюється на повністю розділені підсмуги, кількість яких не перевищує число півхвильових шарів в структурі. Щоб визначити ці структури було проведено моделювання спектральних властивостей структур (1)-(5).

Усі розрахунки було проведено для нормального падіння випромінювання за допомогою матричного методу [15]. Метод дозволяє при заданих параметрах багатошарової структури (N – число шарів, n_r – показник заломлення, k_r – коефіцієнт екстинкції, d_r – товщина r -го шару, n_m і k_m – оптичні сталі підкладки, n_0 і k_0 – оптичні сталі зовнішнього середовища, θ_0 – кут падіння) легко розрахувати спектр пропускання $T(\lambda)$:

$$T = \frac{4\eta_0 \operatorname{Re}(\eta_m)}{(\eta_0 B + C)(\eta_0 B + C)^*} \quad (6)$$

де B і C матричні елементи характеристичної матриці структури

$$\begin{bmatrix} B \\ C \end{bmatrix} = \left(\prod_{r=1}^q \begin{bmatrix} \cos \delta_r & (i \sin \delta_r)/\eta_r \\ i \eta_r \sin \delta_r & \cos \delta_r \end{bmatrix} \right) \begin{bmatrix} 1 \\ \eta_m \end{bmatrix} \quad (7)$$

де

$\delta_r = (2\pi N_r d_r \cos \theta_r)/\lambda$, $\eta_{r,m} = \chi_{\text{vac}} N_{r,m} \cos \theta_{r,m}$ для ТЕ хвиль, $\eta_{r,m} = \chi_{\text{vac}} N_{r,m} / \cos \theta_{r,m}$ для ТМ хвиль, $\chi_{\text{vac}} = 2,6544 \cdot 10^{-3}$ сименс, λ – довжина хвилі, $N_r = n_r - ik_r$ – комплексний показник заломлення r -го шару, η_0 – провідність зовнішнього середовища, $c = 3 \cdot 10^8$ м/с – швидкість світла. Значення кутів θ , визначається законом Снеліуса

$$N_0 \sin \theta_0 = N_r \sin \theta_r = N_m \sin \theta_m \quad (8)$$

При моделюванні спектральних характеристик була врахована дисперсія вибраних плівкових матеріалів: оксидів кремнію (SiO_2) і титану (TiO_2). Для опису дисперсії матеріалів використовували формулу Коші. Показники заломлення SiO_2 визначали за формулою $n(\text{SiO}_2) = 1,46 + (3,35 \cdot 10^{-3}/\lambda^2) + (1,41 \cdot 10^{-5}/\lambda^4)$, а Nb_2O_5 за формулою $n(\text{TiO}_2) = 2,23 + (2,34 \cdot 10^{-3}/\lambda^2) + (7,69 \cdot 10^{-3}/\lambda^4)$, де λ – довжина хвилі в мкм. Врахована також дисперсія показника заломлення підкладки за формулою $n_s = 1,50 + (5,33 \cdot 10^{-3}/\lambda^2) + (4,92 \cdot 10^{-4}/\lambda^4)$. Поглинання в шарах при розрахунку характеристик не враховувалось.

Потрібно відзначити, що структури (2) і (3) відрізняються від структур (4) і (5) тим, що в них усі півхвильові прошарки складаються з одного і того ж матеріалу. В структурах (4) і (5) півхвильові прошарки утворюються різними матеріалами. Ця відмінність певним чином проявляється в спектральних властивостях цих структур.

На рис.1 показано спектральне пропускання структур типу (2) для випадку, коли параметр $l = -1$. В цьому випадку три півхвильових шари в структурі будуть спостерігатися, якщо параметр $m = 4$. В залежності від того, шаром з якого матеріалу починається структура, можливі два типи структур: $S[(\text{HL})^{k+1}\text{H}]^4$ і $S[(\text{LH})^{k+1}\text{L}]^4$. Тут S підкладка, а H і L чверть хвильові шари із матеріалів з високим (TiO_2) і низьким (SiO_2) показниками заломлення, відповідно. Значення параметру k показані на рисунку. Спектри 1 – 6 відповідають структурам $S[(\text{HL})^{k+1}\text{H}]^4$, а спектри 7 – 9 структурам $S[(\text{LH})^{k+1}\text{L}]^4$. В структурах $S[(\text{HL})^{k+1}\text{H}]^4$ півхвильові шари створюються двома сусідніми шарами H , а в структурах $S[(\text{LH})^{k+1}\text{L}]^4$ двома су-

сідніми шарами L. При однакових значеннях параметру k розщеплення смуг прозорості з центральною смугою на довжині хвилі $\lambda_0 = 550$ нм більш повним є для структур $S[(HL)^{k+1}H]^4$. Максимальне пропускання центральної смуги прозорості в усіх спектрах однакове і дорівнює 95,76% на $\lambda_0 = 550$ нм. Положення короткохвильової (λ_1) і довгохвильової (λ_2) компонент а також величини розщеплення $\Delta\lambda_1$ і $\Delta\lambda_2$ наведено в таблиці 1.

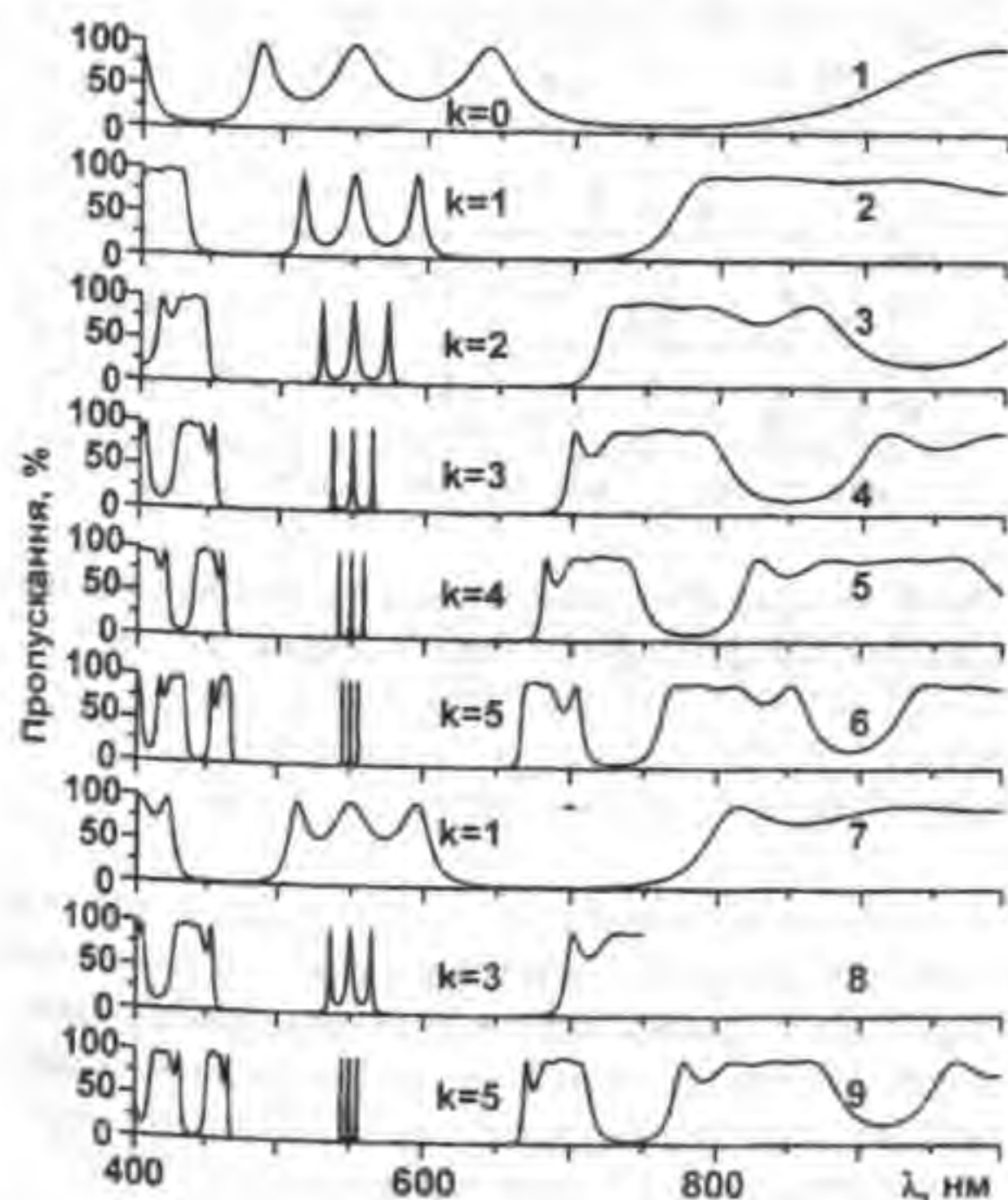


Рис.1. Спектральні залежності пропускання структур $S[(HL)^{k+1}H]^4$ (1 – 6) і $S[(LH)^{k+1}L]^4$ (7 – 9).

Таблиця 1. Положення смуг прозорості.

Структура	λ_1 , нм	λ_2 , нм	$\Delta\lambda_1$, нм	$\Delta\lambda_2$, нм
$S[(HL)H]^4$	486	641	64	91
$S[(HL)^2H]^4$	515	593	35	43
$S[(HL)^3H]^4$	529	574	21	24
$S[(HL)^4H]^4$	536,9	564,1	13,1	14,1
$S[(HL)^5H]^4$	541,8	558,6	8,2	8,6
$S[(HL)^6H]^4$	544,8	555,3	5,2	5,3
$S[(LH)^2L]^4$	513	594	37	44
$S[(LH)^3L]^4$	536,5	564,6	13,5	14,6
$S[(LH)^4L]^4$	544,7	555,5	5,3	5,5

Для структур (2) у випадку $l > -1$ три півхвильові шари будуть існувати при $m = 2$. При $l = 0$ загальна формула структури має вигляд $SLH[(HL)^{k+1}H]^2HL$ або $SHL[(LH)^{k+1}L]^2LH$. Один півхвильовий шар розташований в центрі структури,

а два інші через один чверть хвильовий шар межують з обмежуючими середовищами. Ці два півхвильові резонатори майже не впливають на вигляд спектру пропускання, який має вигляд, що характерний для простого фільтра Фабрі-Перо (рис. 2, криві 1-2). Додавання по боках структури додаткових двох чверть хвильових шарів приводить до неглибокого розщеплення смуги на три компоненти (рис.2, криві 3-4), але при збільшенні параметру k воно зникає (рис.2, крива 5) і спектр має вигляд, характерний для простого фільтра Фабрі-Перо. Більш чітке розщеплення можна отримати при збільшенні параметру l (рис. 2, криві 6 і 7). Але, очевидно, цей шлях менш ефективний, ніж використання параметру $l = -1$.

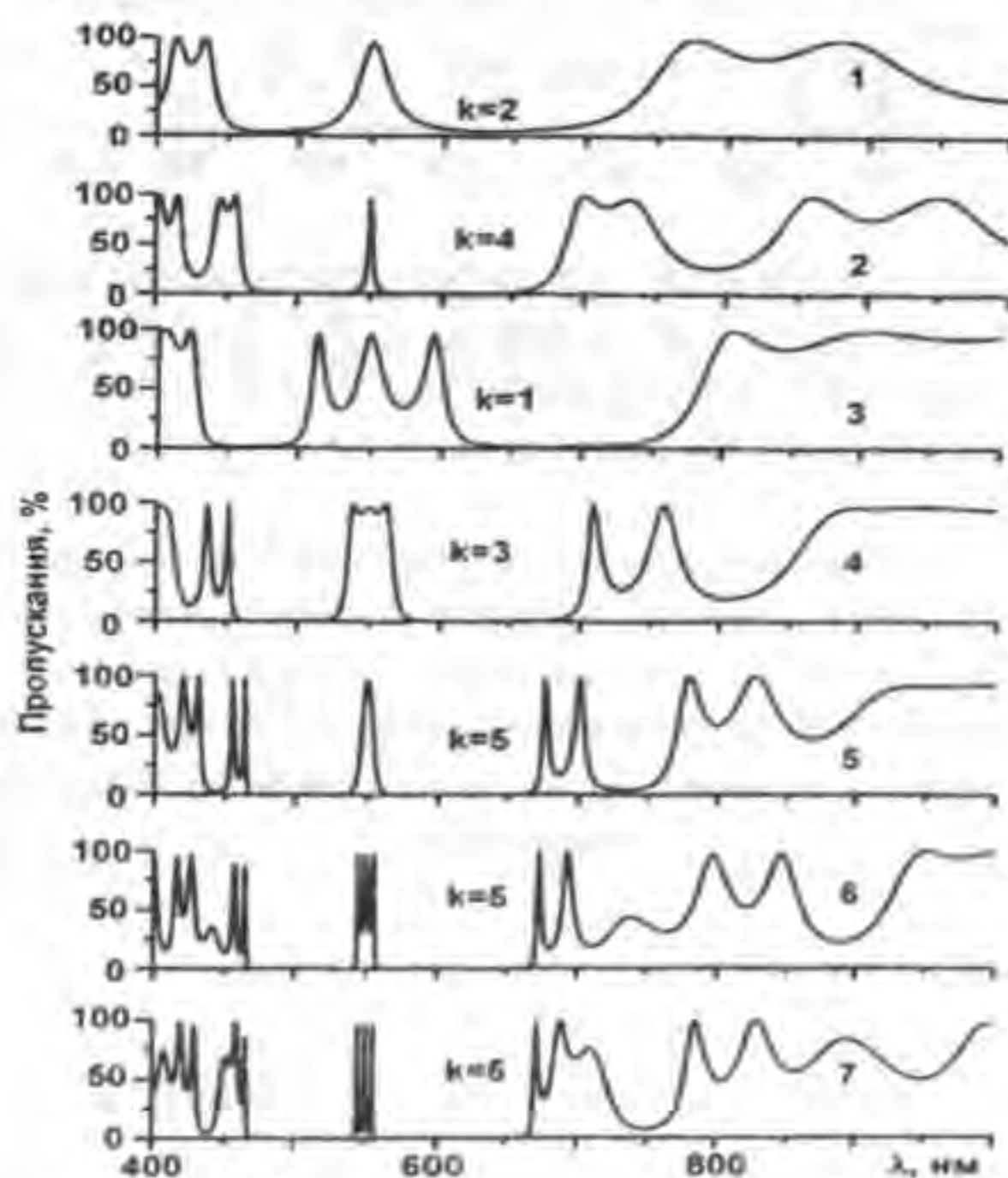


Рис.2. Спектральні залежності пропускання структур $SLH[(HL)^{k+1}H]^2HL$ (криві 1 – 2) та $S[(HL)^k[(HL)^{k+1}H]^2(LH)^l]^4$: $l = 1$ (криві 3 – 5); $l = 3$ (крива 6); $l = 4$ (крива 7).

Структури (3) містять три півхвильові шари, якщо $m = 2$. Можливі два типи структур:

$$S(HL)^lH[(HL)^{k+1}H]^2(HL)^lH \text{ і } S(LH)^lL[(LH)^{k+1}L]^2(LH)^lL.$$

При $l = 0$ і $l = 1$ ці структури можна вважати простим фільтром Фабрі-Перо, оскільки півхвильові шари на початку і на кінці структури практично не впливають на характеристики смуги прозорості на довжині хвилі $\lambda_0 = 550$ нм.

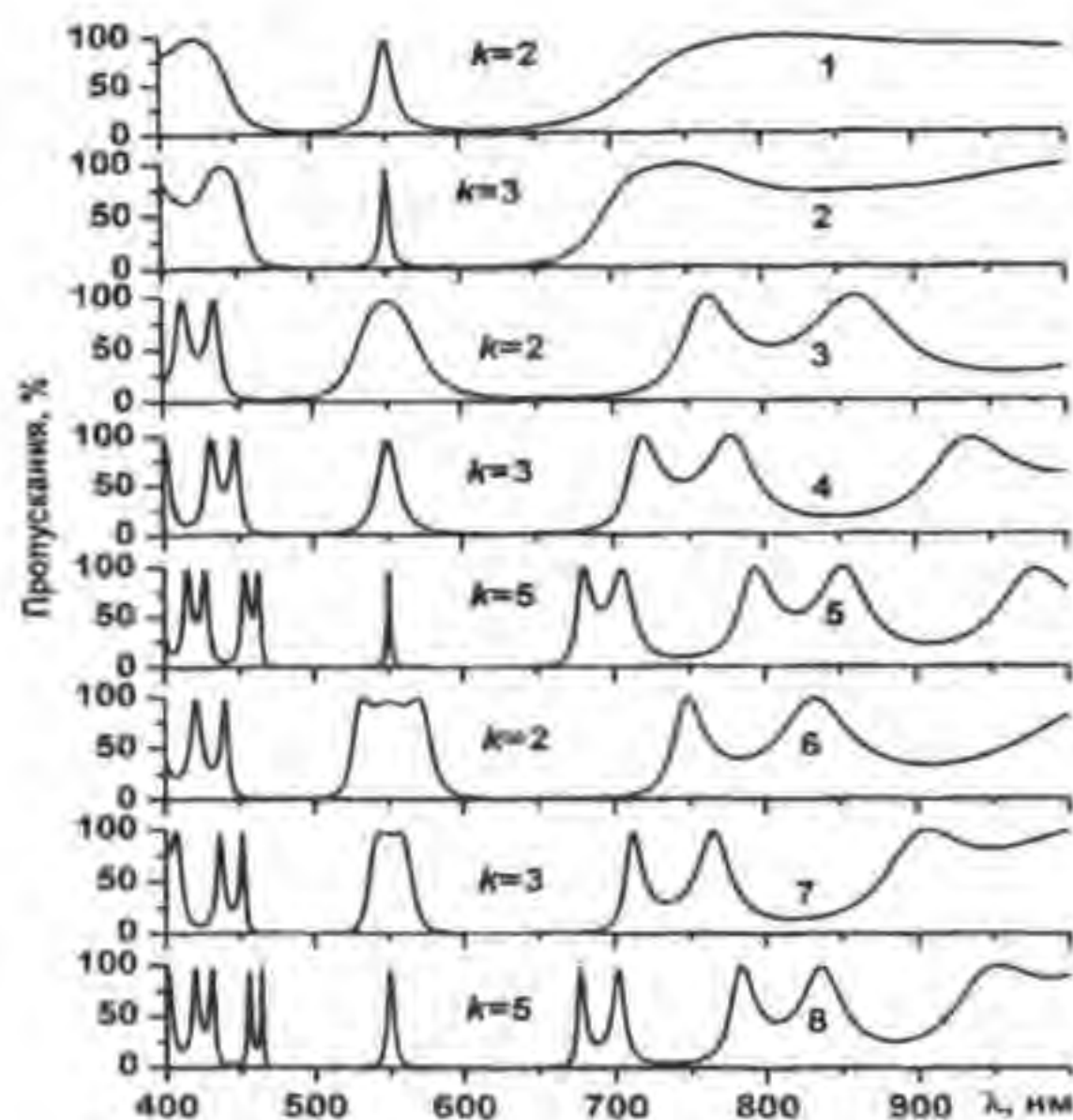


Рис.3. Спектральні залежності пропускання структур $SL[(LH)^{k+1}L]^2L$ (криві 1 – 2), $SLHL[(LH)^{k+1}L]^2LHL$ (криві 3 – 5) та $SHLH[(HL)^{k+1}H]^2HLH$ (криві 6 – 8).

Збільшення кількості чвертьхвильових шарів ліворуч і праворуч крайніх півхвильових шарів приводить до розщеплення центральної смуги прозорості на довжині хвилі $\lambda_0 = 550$ нм (рис.4), однак воно менш чітке, ніж в випадку структур (2) при значеннях $l = -1$.

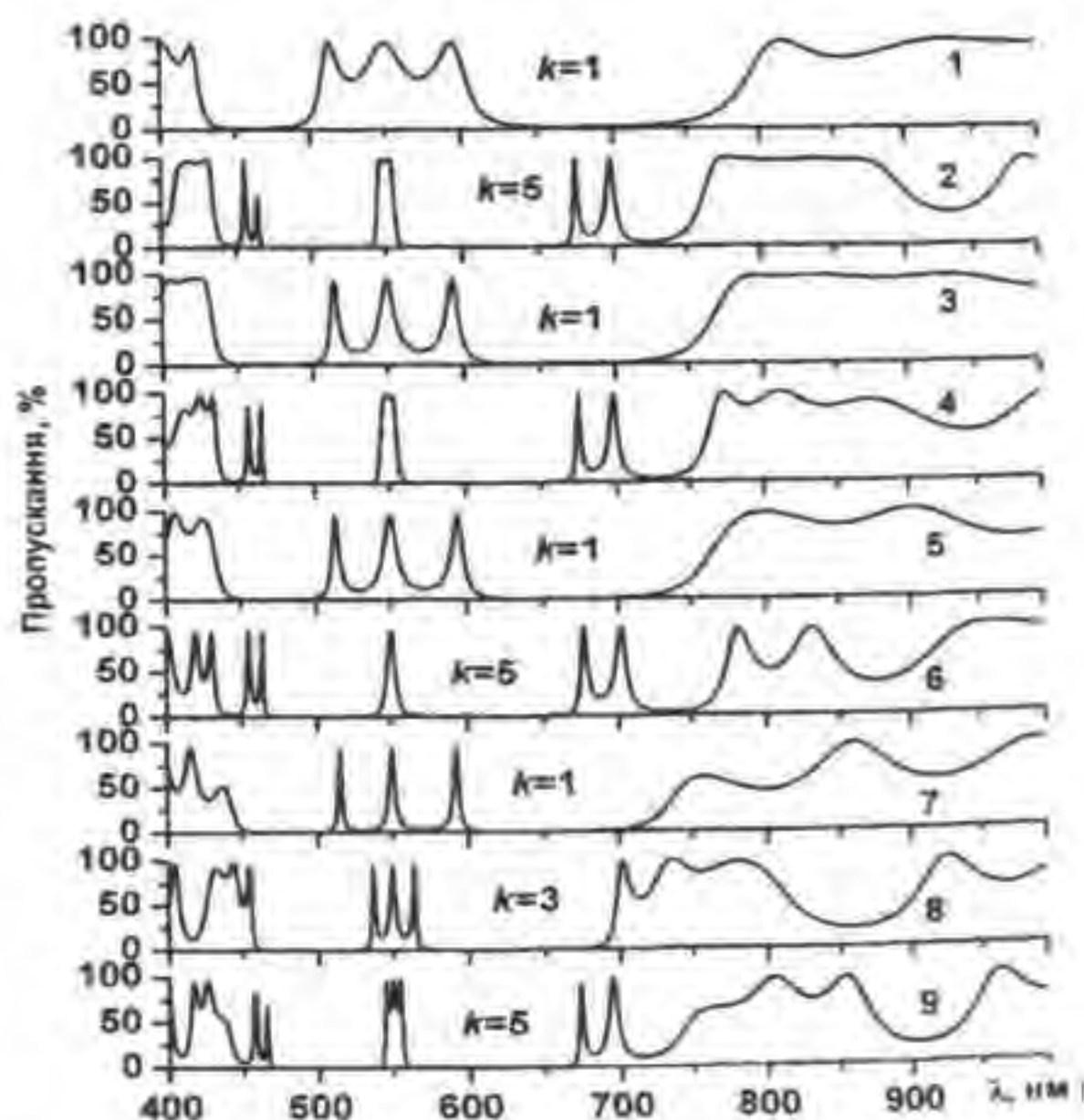


Рис.4. Спектральні залежності пропускання структур $S(LH)^2L[(LH)^{k+1}L]^2(LH)^2L$ (криві 1 – 2), $S(HL)^2H[(HL)^{k+1}H]^2(HL)^2H$ (криві 3 – 4), $S(LH)^3L[(LH)^{k+1}L]^2(LH)^3L$ (криві 5 – 6), $S(HL)^4H[(HL)^{k+1}H]^2(HL)^4H$ (криві 7 – 9).

Таким чином, в роботі було досліджено два типи чвертьхвильових багатошарових діелектричних структур з трьома півхвильовими прошарками і показано, що найбільш чітке розщеплення резонансної смуги прозорості на три окремі смуги отримується в структурах $S[(HL)^{k+1}H]^4$ та $S[(LH)^{k+1}L]^4$.

Список використаних джерел

1. Furman Sh. Basics of Optics of Multilayer Systems / Sh. Furman, A.V. Tikhonravov // Edition Frontieres, Gif-sur-Yvette. – 1992. – 104p.
2. MacLeod H.A. Thin-Film Optical Filters / H.A. MacLeod // Philadelphia: Inst. Phys. Publish., Pa., 2001. – P. 645.
3. Первак Ю.О. Оптика багатошарових покриттів / Ю.О. Первак // ВПЦ «Київський університет». – 2005. – 250 с.
4. Шпак А.П. Фотонні кристали. Фізика та застосування / А.П. Шпак, В.Ю. Первак, Ю.А. Куніцький, Ю.О. Первак // К.: Академперіодика. – 2006. – 120 с.
5. Первак В.Ю. Фізика фотонних кристалів / В.Ю. Первак, А.П. Шпак, Ю.О. Первак, Л.Ю. Куніцька // К.: Академперіодика. – 2007. – 304 с.
6. Несмелов Е.А. К теории отрезающих интерференционных фильтров / Е.А. Несмелов, Г.П. Конюхов // Оптика и спектроскопия. – 1971. – 31. – Вып. 1. – С. 133-137.
7. Марков Ю.Н. К теории полосовых интерференционных фильтров / Ю.Н. Марков, Е.А. Несмелов, А.С. Никитин, И.С. Гайнутдинов // Оптика и спектроскопия. – 1977. – 43. – Вып. 5. – С. 984-989.
8. Марков Ю.Н. К вопросу оптимизации полосовых интерференционных фильтров / Ю.Н. Марков, Е.А. Несмелов, А.С. Никитин, А.С. Гайнутдинов // ЖПС. – 1980. – 33. – №3. – С. 536-540.
9. Марков Ю.Н. Синтез контрастных полосовых интерференционных фильтров. Сложнопериодические системы / Ю.Н. Марков // Оптика и спектроскопия. – 1983. – 54. – Вып. 1. – С. 173-178.

Надійшла до редколегії 04.02.11

СУЧАСНА ФІЗИКА

УДК 532.536

Альошин О.Д., докт.ф.м.н., проф.

Структурні параметри надкритичного флюїду

У роботі методом молекулярного розсіяння світла в просторово неоднорідних системах поблизу критичної точки в полі гравітації Землі досліджені структурні характеристики надкритичного флюїду: величини радіуса кореляції, кількості флуктуацій параметра порядку в одному молі речовини, загального об'єму цих флуктуацій, густини речовини всередині флуктуацій параметра порядку; оцінено енергію міжмолекулярної взаємодії всередині флуктуацій параметра порядку. На основі одержаних даних зроблено висновок, що унікальні властивості надкритичного флюїду, які проявляються при його практичному застосуванні, є наслідком великої швидкості молекул речовини, які розлітаються при розпаді флуктуацій параметра порядку.

Ключові слова: надкритичний флюїд, критична точка, гравітаційний ефект, флуктуація

* E-mail: alekhin@univ.kiev.ua

Проблема фазових переходів другого роду та критичних явищ у рідинах є однією з актуальних і фундаментальних задач фізики конденсованого стану речовини. Відповідно до флуктуаційної теорії фазових переходів ФТФП [1] при наближенні до критичної точки (КТ) необмежено зростає основний параметр речовини – радіус кореляції R_c системи, що приводить до необмеженого зростання стисливості β_T ; теплоємності C_v ; теплопровідності k речовини; прямуванню до нуля коефіцієнта дифузії D ; температуропровідності χ . Ці унікальні властивості речовини в критичній області (надкритического флюїду - НКФ) - знаходять своє всебічне практичне використання у новітніх сучасних технологіях [2-5], наведених нижче.

1. Явище надкритической екстракції з використанням CO_2 дозволяє найбільш повно та швидко виділяти з рослинних і тваринних продуктів кофеїн, ефірні масла, радіопротекторні, біологічно активні речовини і багато інш.

Alekhin O.D., Dr. Sci., Prof.,

Structure parameters of supercritical fluids

The structure characteristics of supercritical fluids have been investigated for spatially inhomogeneous systems under gravity near the critical point by the molecular scattering method. It have been obtained in paper the correlation length, the number of fluctuations of order parameter in one mole of substance, the whole volume of these fluctuations, the density of substance inside a fluctuation of order parameter; it has been estimated the energy of intermolecular interaction inside a fluctuation of order parameter. It was made a conclusion on the base of obtained data that unique properties of supercritical fluid, which are manifested at the practice using, are due to the high velocity of flying apart of the molecules at the decay of the fluctuations of order parameter.

Key words: supercritical fluid, critical point, effect of gravity, fluctuation

2. CO_2 у надкритическом стані успішно використовується з метою підвищення видобутку нафти і її очищення від важких фракцій і металів, у процесах нафтохімії та багатьох інших галузях.
3. Унікальна розчинююча здатність НК- H_2O ефективно використовується при очищенні навколишнього середовища від забруднюючих її промислових і органічних відходів, для знищення хімічної зброї й бойових отруйних речовин.
4. Актуальною проблемою сучасної енергетики є розробка новітніх технологій з метою одержання та виробництва нових видів біопалив з використанням НК- H_2O .
5. НК- H_2O інтенсивно використовується для одержання нанокристалічних часток із заданими властивостями. Зокрема, з аморфного вуглецю синтезуються вуглецеві нанотрубки.
6. Окремо слід вказати на величезну роль НК- H_2O у природі [5]. Формування земної кори, вулканічна діяльність, перенос і

концентрування рудоутворюючих компонентів у родовищах і багато інше - все це наслідок активності надкритичного гідротермального флюїду.

На жаль, до теперішнього часу немає точної відповіді на питання, що є основним чинником таких унікальних властивостей НКФ при його практичному використанні. У той же час, розуміння того, що відбувається у надкритичних середовищах, надзвичайно важливо для усвідомленого вибору параметрів технологічних процесів і інтерпретації глобальних природних явищ.

Тому, мета даної роботи - пов'язати унікальні властивості НКФ при їхньому практичному використанні із структурними характеристиками та параметрами рівняння стану речовини у критичній області.

Згідно ФТФП [1] НКФ за своєю структурою є флуктуаційно-дисперсною системою - газом флуктуацій параметра порядку розміром R_c . Флуктуаційна частина термодинамічного потенціалу такої системи поблизу КТ визначається співвідношенням, подібним до рівняння енергії ідеального газу.

$$F_{\Phi}^* = \frac{F_{\Phi}}{P_c V_c} = \frac{N_{\Phi} k_B T_c}{P_c V_c} = \frac{N_{\Phi}}{N_A Z_c} = C_0 R_c^{-3} =$$

$$= C_0 [t^{\nu} \Phi(\Delta\mu / t^{\xi/\nu})]^{-1}; \quad (1)$$

$$C_0 = \frac{\rho_c}{\rho_{\Phi}} \cdot \frac{V_c}{4/3 \pi} (Z_c N_A)^{-1}$$

Тут $\Phi(\Delta\mu / t^{\xi/\nu})$ - масштабна функція ФТФП [1]; $\Delta\mu = (\mu - \mu_c) / \mu_c$; $t = (T - T_c) / T_c$; ξ , ν - критичні показники ФТФП [1]; $Z_c = P_c V_c / (RT_c)$; μ_c , T_c , P_c , ρ_c , V_c - критичні значення хімічного потенціалу, температури, тиску, густини та молярного об'єму речовини; N_A - число Авогадро.

Виходячи з (1), основними структурними параметрами НКФ є: N_{Φ} - число флуктуацій параметра порядку в одному молі речовини, R_c - радіус кореляції системи, ρ_{Φ} - густина речовини усередині флуктуації параметра порядку.

Для визначення цих структурних характеристик НКФ були використані експериментальні дані висотної та температурної залежності інтенсивності розсіяного світла

$I(z, t) \sim d\rho^*/d\mu^*$ неоднорідних систем поблизу КТ у полі гравітації Землі $h = \rho_c g z / P_c$ [7-9] (Тут $\rho^* = \rho / \rho_c$, $\mu^* = \mu / \mu_c$).

На основі цих експериментальних даних $I(z, t)$ величина радіуса кореляції $R_c(z, t)$ речовини визначається співвідношенням

$$R_c(z, t) = \frac{1}{q} \left[\left(\frac{I_c}{I(z, t)} \right)^{1 + \frac{\eta}{2}} - 1 \right]^{-\frac{1}{2}} \quad (2)$$

де $q = 4\pi/\lambda \cdot \sin(\theta/2)$, I_c - інтенсивність розсіяного світла в критичній точці; $\eta = 0,06$ - критичний показник [1].

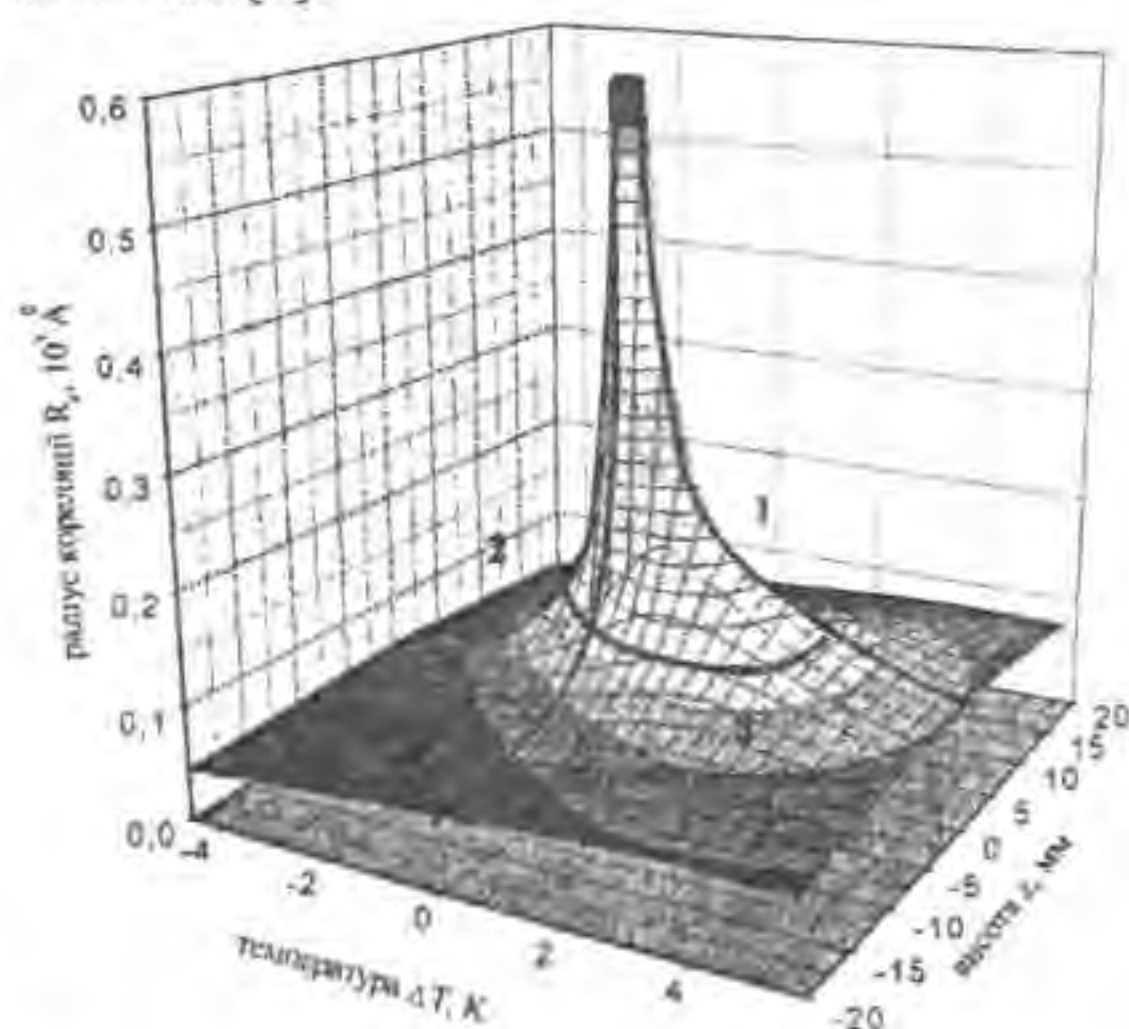


Рис. 1.

На Рис. 1, показано висотні та температурні залежності радіуса кореляції речовини, розраховані за формулою (2).

Ці залежності радіуса кореляції $R_c(h, t)$ (Рис. 1.) уздовж напрямків: критичної ізохори - 1; межі поділу фаз - 2; критичні ізотерми - 3, описуються відповідно співвідношеннями

$$R_c = r_1 \cdot t^{-\nu}; R_c = r_2 |t|^{-\nu}; R_c = r_3 \cdot h^{-\xi}, \quad (2)$$

де $\nu = 0,636$, $\xi = 0,405$ - критичні показники [1,6]; амплітуди: $r_1 \approx 3,8 \text{ Å}$, $r_2 \approx 2 \text{ Å}$, $r_3 \approx 1 \text{ Å}$.

В області висот $z = (5 \div 10) \text{ см}$ ($h = (0 \div 10^{-4})$) та температур $t = (10^{-2} \div 10^{-4})$ радіус кореляції змінюється в межах величин $R_c = 1 \div 100 \text{ нм}$.

Виходячи із цих даних $R_c(h, t)$, знаходимо флуктуаційну частину термодинамічного

потенціалу $F_{\phi}^*(t, h) = N_{\phi}(t, h) \cdot k_B T_K / P_K V_K =$
 $C_0 R_c^{-3}(t, h)$, ($C_0 \approx 10^{-22} \text{ см}^3$) [8,9], (Рис.2).

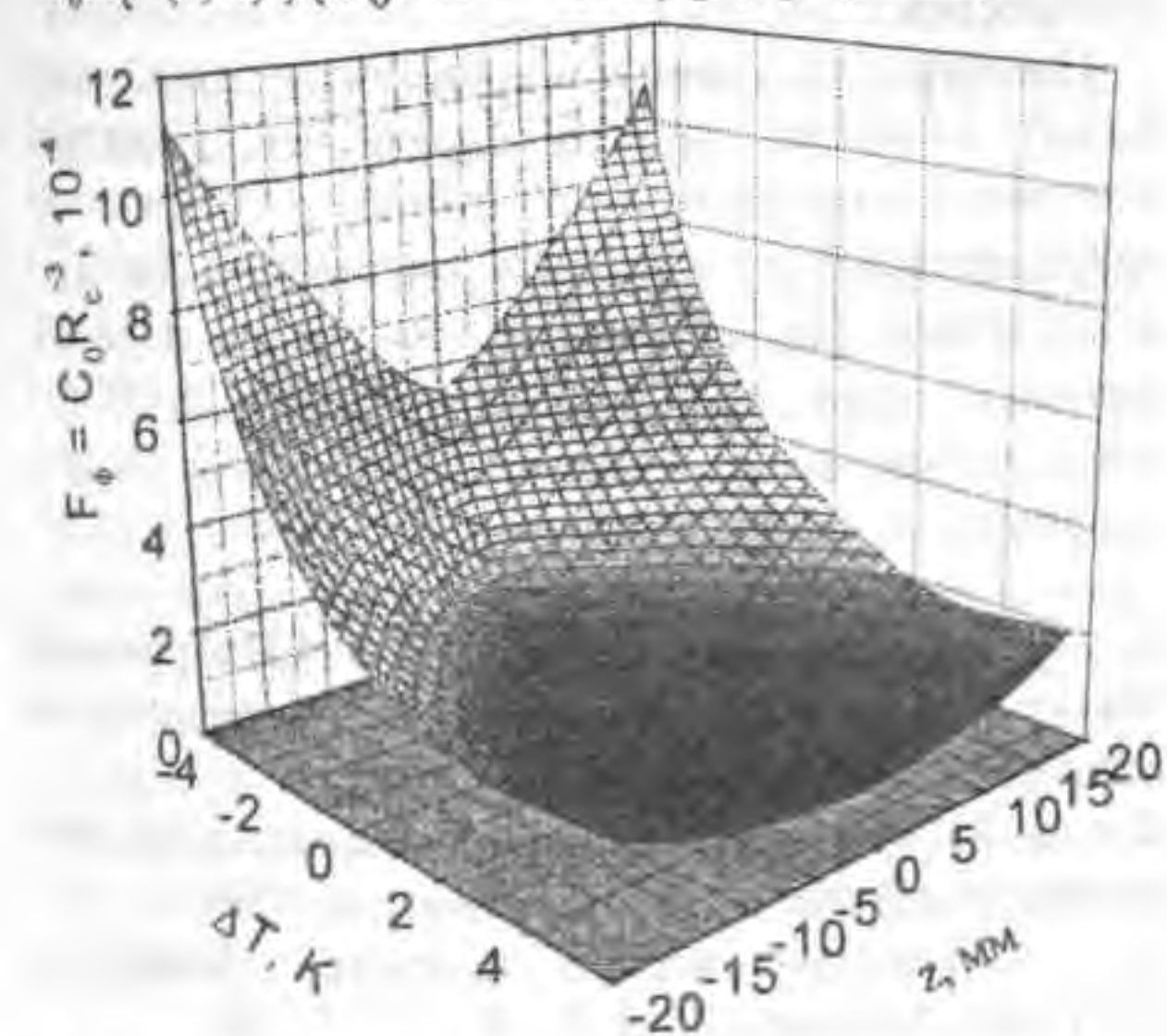


Рис. 2.

На основі виду цієї поверхні $F_{\phi}^*(t, h)$ (Рис. 2.) можна знайти всі термічні та калоричні рівняння стану речовини (НКФ): густини $\Delta\rho = dF_{\phi}^*/d\mu$; стисливості $\beta_T = d^2F_{\phi}^*/d\mu^2$; ентропії $\Delta s = dF_{\phi}^*/dt$; теплоємності $C_v = d^2F_{\phi}^*/dt^2$; коефіцієнта об'ємного розширення $\alpha_T = d^2F_{\phi}^*/d\mu dt$.

Використовуючи вигляд поверхні F_{ϕ}^* (Рис. 2) визначимо ще одну структурну характеристику НКФ – число флуктуацій параметра порядку в одному молі речовини $N_{\phi}(h, t) = \frac{P_K V_K}{k_B T_K} C_0 R_c^{-3}(h, t) = N_A Z_K F_{\phi}^*(h, t)$. Ці

результати представлені на Рис. 3.

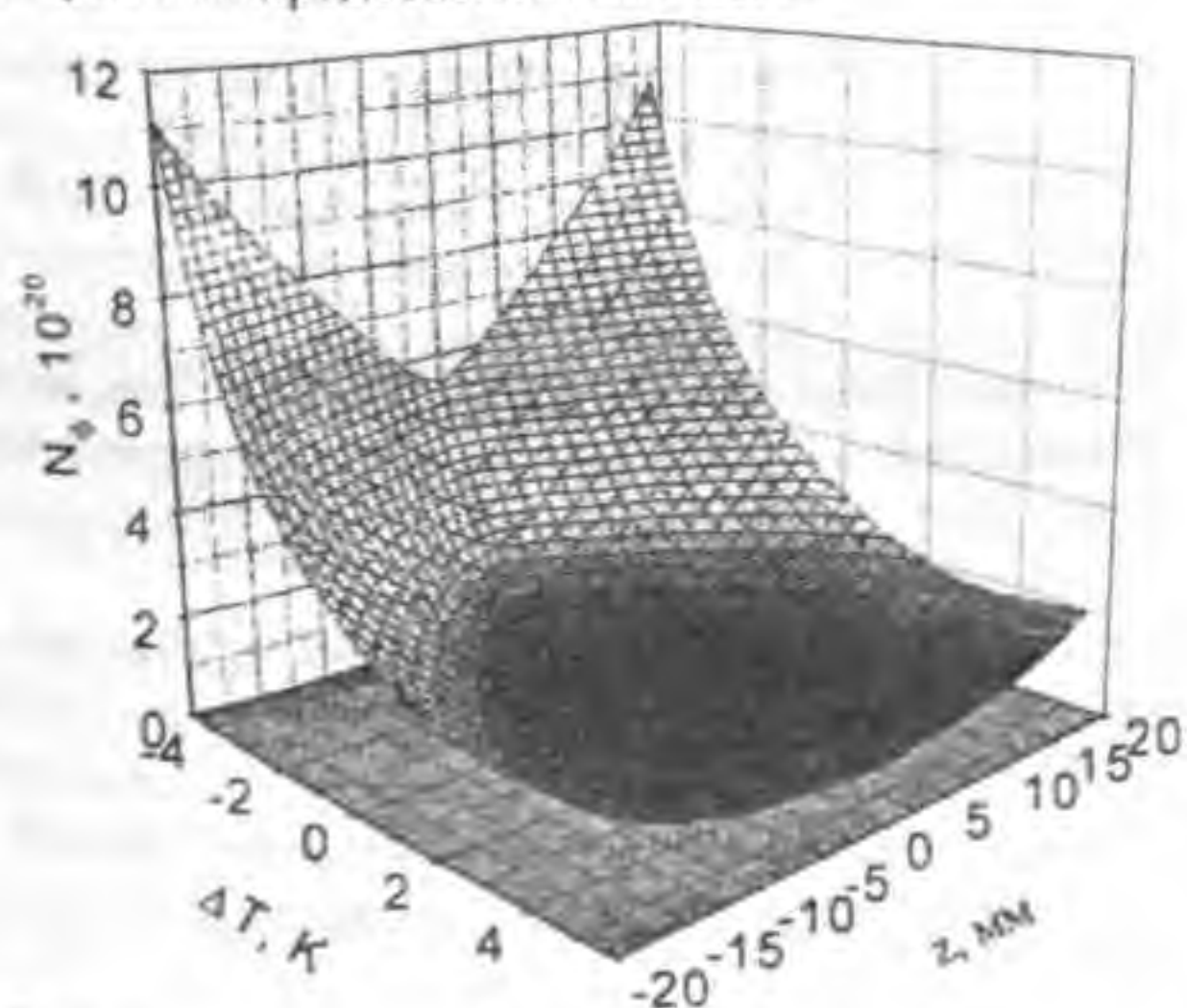


Рис. 3.

Дана характеристика НКФ уздовж граничних термодинамічних напрямків критичної ізохори, межі поділу фаз, критичної ізотерми представляється відповідно у вигляді:

$$\begin{aligned} N_{\phi} &= 2,9 \cdot 10^{23} \cdot t^{3v}; & N_{\phi} &= 2 \cdot 10^{24} \cdot t^{3v}; \\ N_{\phi} &= 2 \cdot 10^{25} \cdot h^{3\xi} \end{aligned} \quad (3)$$

Використовуючи результати $R_c(h, t)$ і $N_{\phi}(h, t)$ знаходимо величини флуктуаційного об'єму $v_{\phi} = \frac{4}{3} \pi R_c^3$, повний об'єм, що займають усі флуктуації параметра порядку

$$V_{\phi} = v_{\phi} \cdot N_{\phi} = \frac{4}{3} \pi R_c^3 \cdot N_{\phi}, \text{ а також густину речовини } \bar{\rho}_{\phi} = \frac{V_K}{V_{\phi}} \rho_K \geq 3\rho_K \text{ усередині}$$

флуктуаційного об'єму v_{ϕ} . Тоді схематичний розподіл густини речовини $\bar{\rho}_{\phi}(r)$ усередині флуктуаційного об'єму v_{ϕ} можна представити у вигляді (Рис. 4).

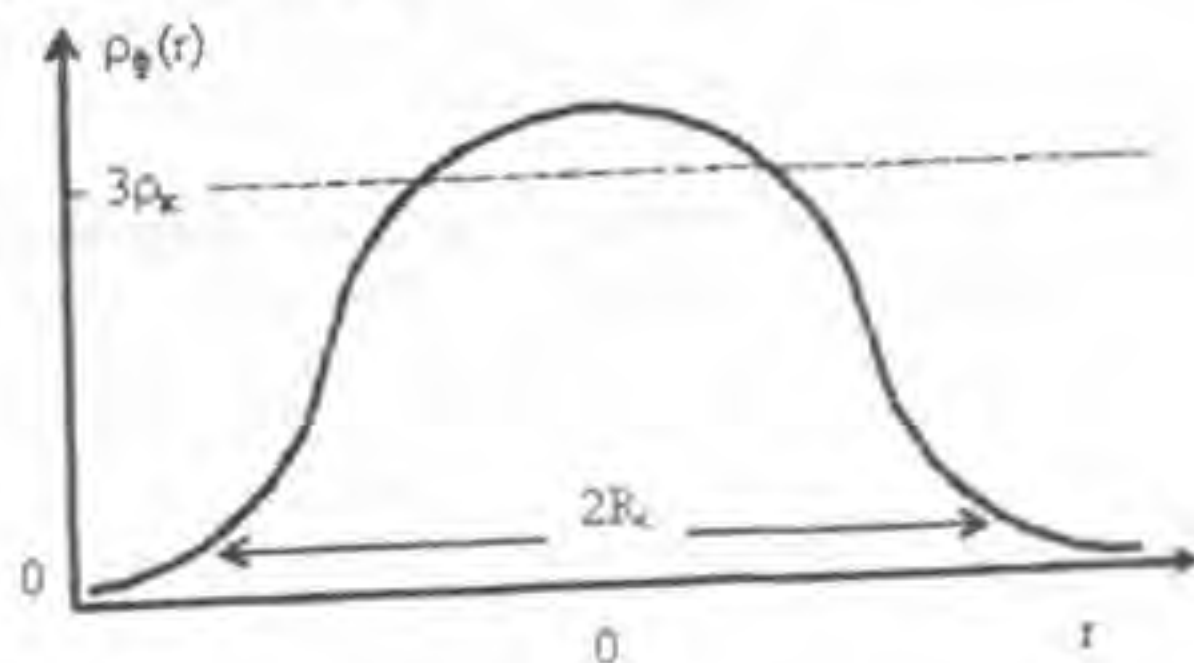


Рис. 4.

Отриманий результат $\bar{\rho}_{\phi} \geq 3\rho_K$ (Рис. 4.) свідчить про те, що середня густина речовини усередині флуктуаційного об'єму v_{ϕ} значно перевищує критичну густину речовини ρ_K та досягає величини густини рідини ρ_K віддалік від КТ. На підставі цього результату в [10, 11] зроблений висновок, що при таких значеннях густин $\rho_{\phi} \geq 3\rho_K$ усередині флуктуаційних утворень v_{ϕ} енергія міжмолекулярної взаємодії досить велика ($\epsilon_l \gg k_B T_K$). Виходячи із цієї сильної нерівності можна стверджувати, що після певного часу життя τ_{ϕ} флуктуація руйнується, а молекули речовини розлітаються з великими

швидкостями $v \gg (2k_B T_K/m)^{1/2}$, збираючись у чергову флуктуацію параметра порядку в іншому місці (Рис.5.).

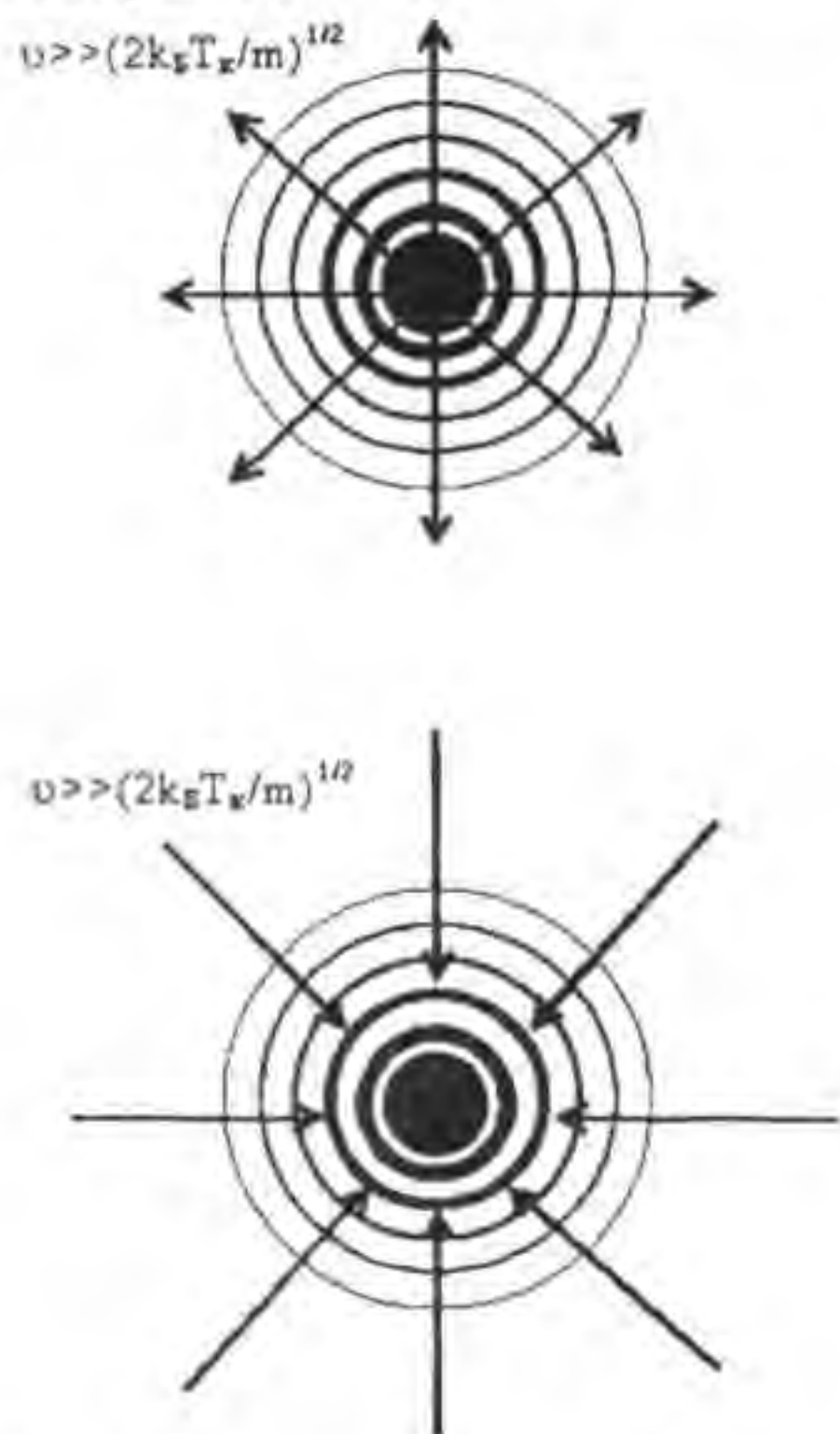


Рис. 5.

На основі такої моделі утворення і руйнування флуктуацій параметра порядку можна припустити, що саме великі швидкості руху молекул речовини є основною причиною прояву унікальних властивостей НКФ при його практичному використанні. До них необхідно віднести: різноманітні види СК-екстракції, застосування НК- CO_2 з метою підвищення видобутку нафти з нафтових шарів, використання НК- CO_2 для очищення електронних плат від забруднення, унікальна здатність НК-води розчиняти органічні та неорганічні [4] сполуки, їх використання для очищення навколишнього середовища й знищення різних видів хімічної зброї, використання НК- H_2O для одержання нанокристалічних часток із заданими властивостями та багато іншого [2,3].

Таким чином, представлені результати експериментальних досліджень висотних та температурних залежностей інтенсивності розсіяного світла в просторово неоднорідній речовині в полі гравітації Землі дозволили визначити ряд структурних характеристик НКФ:

R_c , N_ϕ , v_ϕ , ρ_ϕ , $\epsilon_1 \gg k_B T_K$, які відповідають за унікальні властивості НКФ при їх практичному використанні.

Представлені вище експериментальні дані $R_c(h,t)$ показали, що на даний час унікальні властивості надкритичного флюїду [2-5] реально використовуються в області параметрів $h \geq 10^{-5}$ і $t \geq 10^{-2}$ (лінія (4) на рис. 1). Як видно з цього рисунка, лише в області параметрів $h < 10^{-5}$ і $t < 10^{-2}$ унікальні властивості НКФ [2-5] будуть проявлятися найбільшим чином.

1. А.З.Паташинский, В.Л.Покровский Флуктуационная теория фазовых переходов. 2-е изд./ - М.: Наука, 1982. - 382 с.
2. М. McHung, V. Krukonis. Supercritical fluid extraction / Butterworth - Heinemann. 1994.
3. Сверхкритические флюиды. Теория и практика. Москва, 2008, Т., № 2. С.1 -101.
4. Сверхкритические флюиды. Теория и практика. Москва, 2010, Т.5, № 1. С. 5-102.
5. Ю.Э. Горбатый, Г.В. Бондаренко, Сверхкритическое состояние воды // Сверхкритические флюиды: Теория и Практика, 2007, Т.2, № 2, 5-19.
6. Alekhin A.D., Critical indexes for systems of different space dimensionality./ A.D. Alekhin // Journal of Molecular Liquids. - 2005. - 120 - P. 43-45.
7. Алехин А.Д. Критические явления жидкость-пар в гравитационном поле: Дис... д-ра физ.-мат.наук: 01.04.14./ Киев, 1994. 296 с.
8. JML 2006 Alekhin A.D. Gravity effect influence on internal field of liquid near the critical point // Journal of Molecular Liquids - 2006 - 127 - P. 62-64
9. Альохін О.Д., Дорош А.К., Рудніков Е.Г., Критическое состояние вещества в поле гравитации Земли. / Киев. Политехника. - 2008. - 404 с.
10. А.Д. Алехин Структура и свойства сверхкритического флюида - основы новейших технологий // ХХІV-ая научная конференция стран СНГ "Дисперсные системы", 2010, Одесса, Украина, С. 23-24.
11. Алехин А.Д. Исследования структуры и свойств сверхкритического флюида // Труды международной конференции „фазовые переходы, критические и нелинейные явления в конденсированных средах”, Махачкала 2010, С. 136-139.

Надійшла до редколегії 30.01.2011

УДК 532

Бардік В.Ю., к.ф.-м.н.
Слинчак О.Л., асп.

Особливості розповсюдження звуку у розбавлених водних розчинах спиртів в околі їх особливих точок

Аналізуються особливості розповсюдження звуку у водному розчині γ -піколіну в околі його особливої точки. Враховується, що специфіка температурної залежності звуку обумовлена процесами кластеризації у розчині. Показано, що високочастотна швидкість звуку, яка формує піки Мандельштама-Бріллюена у молекулярному розсіюванні світла, має мінімуми при певних температурах та концентраціях розчинів. Має місце повна узгодженість теоретичних розрахунків та експериментальних даних.

Розбавлені водні розчини неелектролітів є предметом численних досліджень різноманітними методами, що пояснюється наявністю багатьох особливостей їхніх фізичних властивостей. Не є виключенням і водні розчини спиртів та метилпіридинів.

Суттєву інформацію про динаміку процесів, що протікають поблизу особливих точок, дають дослідження розповсюдження ультра- та гіперзвуку. Дані про швидкість затухання гіперзвуку зазвичай отримують із спектрів розсіювання Мандельштама-Бріллюена. Ці спектри виникають внаслідок розсіювання світла на теплових флуктуаціях тиску. Розраховується швидкість звуку (або температурний коефіцієнт затухання), як правило, за напівшириною спектральних ліній Мандельштама-Бріллюена.

В [1] за спектрами Мандельштама-Бріллюена проводилось вивчення температурних залежностей швидкості гіперзвуку та ультразвуку, а також коефіцієнта поглинання гіперз-

Bardik V.Yu.,
Slynchak O.L., PhD

Specificity of the sound spreading in diluted aqueous solutions of alcohols near their peculiar points

Specificity of the sound spreading in aqueous solution of γ -picolin near its peculiar point is analyzed. It is taken into account that temperature dependence of the sound velocity is caused the clusterization effects in solutions. It is shown, that the high frequency sound velocity, which forms the Mandelstam-Brillouin peaks in molecular light scattering, has minimum at definitive temperature and concentration. The full agreement between theoretical calculations and experimental data takes place.

вуку у розчині β -піколін-вода при концентрації $x \sim 0.07$.

Швидкість гіперзвуку визначалась із зміщення компонент Мандельштама-Бріллюена з точністю $\pm 1\%$. Коефіцієнт поглинання гіперзвуку визначався із ширини компонент Мандельштама-Бріллюена (точність складала $\pm 5\%$).

Експериментально була досліджена швидкість розповсюдження гіперзвуку в інтервалі температур від 40 до 78 °C. При температурах від 40 до 55 °C відбувається монотонне зменшення швидкості ультразвуку та швидкості гіперзвуку з температурою, причому величина

$\frac{\partial c}{\partial T} = -0.8$. В інтервалі температур від 62 до 72 °C швидкості ультразвуку і гіперзвуку практич-

но не залежать від температури ($\frac{\partial c}{\partial T} \approx 0$). В

цьому ж інтервалі температур спостерігається максимум поглинання гіперзвуку і максимум

Представлено: акад. НАН України,
д.ф.-м.н., проф. Л.А. Булавін

інтенсивності центральної компоненти. Максимум поглинання гіперзвуку знаходиться при тих же температурах ($50 \div 70^\circ\text{C}$), що й максимум інтенсивності центральної компоненти.

В інтервалі температур від 72 до 78°C температурна похідна швидкості звуку $\frac{\partial c}{\partial T} = -3.6$, що в декілька разів перевищує значення похідної швидкості гіперзвуку в інтервалі від 40 до 60°C . Цей результат свідчить про деяку перебудову структури розчину при переході через область "особливої точки". Додамо, що у водному розчині β -піколіну при температурах від 40 до 60°C згідно з [2], спостерігається додатна дисперсія звуку $\sim 2.2\%$, а при наступному збільшенні температури дисперсія швидкості звуку відсутня.

В інтервалі температур від 32 до 42°C швидкість розповсюдження ультразвуку в розчині більша ніж швидкість розповсюдження ультразвуку в будь-якій з чистих компонент. Згідно з [3], це означає, що стисливість розчину менша ніж стисливість води.

Крім того, проведене дослідження вказує на те, що хоча критична точка розшарування і не виникає, та все-таки поведінка інтенсивності розсіювання світла, швидкості звуку вказують на наближеність розчину до недосяжної критичної точки розшарування.

Дослідження ультраакустичних параметрів водних розчинів метилпіридинів показали, що ізотерми адіабатичної стисливості цих розчинів перетинаються в області концентрацій $x \sim 0.04$. Подібна температурна незалежність стисливості розчинів при певних концентраціях пояснюється збільшенням "жорсткості" структури розчину при потраплянні молекул метил-

піридину у сітку водневих зв'язків, утворювану молекулами води.

Вивчення гіперакустичних параметрів водного розчину β -піколіну [4], виявило зменшення швидкості гіперзвуку при концентрації $x \sim 0.06$. Така поведінка швидкості звуку може бути пояснена утворенням кластерів у розчині вода- β -піколін.

Бартоном [5] були виконані дослідження швидкості і поглинання ультразвуку у водних розчинах третичного бутилового спирту. З'ясувалось, що для вимірюваних величин місце максимум поглинання ультразвуку при концентрації $x \sim 0.045$. В околі зазначеної концентрації спостерігається дисперсія швидкості звуку близько 7% .

Температурний коефіцієнт швидкості звуку складає $\frac{\partial c}{\partial T} = +3.8$ [6]. Максимуми поглинання ультразвуку, як і у випадку піколіну, відповідають тим самим концентраціям, що й максимуми розсіювання світла.

Стисливість, що є термодинамічною величиною, проявляє аномальні властивості в околі "особливої точки". Адіабатичну стисливість можна розрахувати за даними по швидкості звуку. Згідно з [7] адіабатична стисливість, обчислена для водних розчинів етанолу, ізопропанолу, н.-пропанолу та трет-бутанолу, має мінімуми, розташовані в околі відповідних концентрацій $x \sim 0.095; 0.075; 0.05; 0.045$, які знаходяться практично на тих же концентраціях, що й максимуми теплоємності та максимуми інтенсивності молекулярного розсіювання світла в цих розчинах.

Згідно з [8], стисливість водних розчинів неелектролітів визначається, в першу чергу,

ближньою гідратацією молекул розчиненої речовини, тобто взаємодією молекул розчиненої речовини з найближчими до нього молекулами розчинника. Насправді, сили цієї взаємодії є короткодійними, і тому неможливо очікувати помітного впливу молекул неелектроліту на структуру води на відстанях, що перевищують декілька радіусів молекул води.

В роботі [4] було показано, що адіабатична стисливість у водних розчинах γ -піколіну має мінімум в околі концентрацій $x \sim (0.04 - 0.06)$, причому положення мінімуму зміщується в сторону менших концентрацій при підвищенні температури розчину. Зокрема, було виявлено зменшення швидкості гіперзвуку при концентрації $x \sim 0.06$, що свідчить про можливість утворення кластерів у розчині.

З термодинамічної теорії відомо, що швидкість звуку може бути обчислена за формулою:

$$c^2 = \left(\frac{\partial p}{\partial \rho} \right)_T. \quad (1)$$

Оскільки аналізується поведінка швидкості звуку у водно-спиртовому розчині, то тиск є сумою тиску води та осмотичного тиску, обумовленого молекулами розчиненої речовини:

$$p = p_w + \Delta p_{osm}. \quad (2)$$

Таким чином, осмотична складова швидкості звуку дорівнює:

$$\Delta c_{osm}^2 = \left(\frac{\partial \Delta p_{osm}}{\partial \rho} \right)_T = \left(\frac{\partial \Delta p_{osm}}{\partial v} \right)_T \cdot \left(\frac{\partial v}{\partial \rho} \right)_T. \quad (3)$$

де $\left(\frac{\partial v}{\partial \rho} \right)_T = -\frac{v^2}{m_w}$. Тут v – об'єм, який приходить на одну молекулу води.

Осмотичний тиск Δp_{osm} моделюємо рівнянням Ван-дер-Ваальса [9]:

$$\begin{aligned} \Delta p_{osm} &= \frac{T x n}{1 - \frac{x n}{n_0}} - a x^2 n^2 = \frac{T x \rho}{m_w (1 - \frac{x \rho}{\rho_0})} - \frac{a x^2}{m_w^2} \rho^2 = \\ &= \frac{T x \rho}{m_w (1 - \frac{x \rho}{\rho_0})} - \frac{a x^2}{m_w^2} \rho^2 \end{aligned} \quad (4)$$

де T – температура розчину, x – концентрація розчину, ρ та ρ_0 – густина розчину та кластерів у розчині відповідно, m_w – маса молекули води, a – константа рівняння Ван-дер-Ваальса.

Згідно [9], величина $\left(\frac{\partial p_{osm}}{\partial v} \right)_T$ має ви-

гляд:

$$\left(\frac{\partial p_{osm}}{\partial v} \right)_T = -k_B T \frac{n_w x^2}{z_p} \cdot A, \quad (5)$$

де k_B – стала Больцмана, z_p – кількість молекул γ -піколіну у кластері, n_w – густина молекул води,

$$A = \left(\frac{\partial \tilde{p}}{\partial x} \right)_T = \frac{\tau_s(x)}{(1 - \tilde{b}x)^2}, \quad (6)$$

де $\tau_s(x) = \frac{T - T_s(x)}{T_s(x)}$, $T_s(x)$ – концентраційна

залежність положення псевдоспінодалі. Положення псевдоспінодалі у розчині вода- γ -піколін розраховано в роботі [10]. Константи $\tilde{a}$ та $\tilde{b}$ з наближенням до перколяційного порогу ($x_p \sim (0.05 - 0.07)$) зростають, що говорить про збільшення перколяційного кластеру та взаємодії між кластерами.

В околі псевдоспінодалі величина $\left(\frac{\partial p_{osm}}{\partial v} \right)_T$ прямує до нуля, тому при наближенні до псевдоспінодалі та особливої точки очікується мінімум швидкості звуку. Це припущення повністю узгоджується з експериментальними

результатами та кластерними уявленнями про будову розчину.

Об'єднавши вирази (3) – (6), отримаємо співвідношення для осмотичної складової швидкості звуку у розчині вода-γ-піколін:

$$\Delta c_{\text{osm}}^2 = \frac{k_B T}{m_s} \cdot \frac{x^2}{z_p} \cdot \frac{r_s(x)}{(1 - \tilde{b}x)^2} \quad (7)$$

На Рис. наведена температурна залежність швидкості звуку у розчині вода-γ-піколін, обчислена за формулою: $c = c_w + \frac{1}{2} \frac{\Delta c_{\text{osm}}^2}{c_w}$, де c_w позначає високочастотну складову швидкості звуку у воді, яка відповідає положенню компонент Мандельштама-Бріллюена.

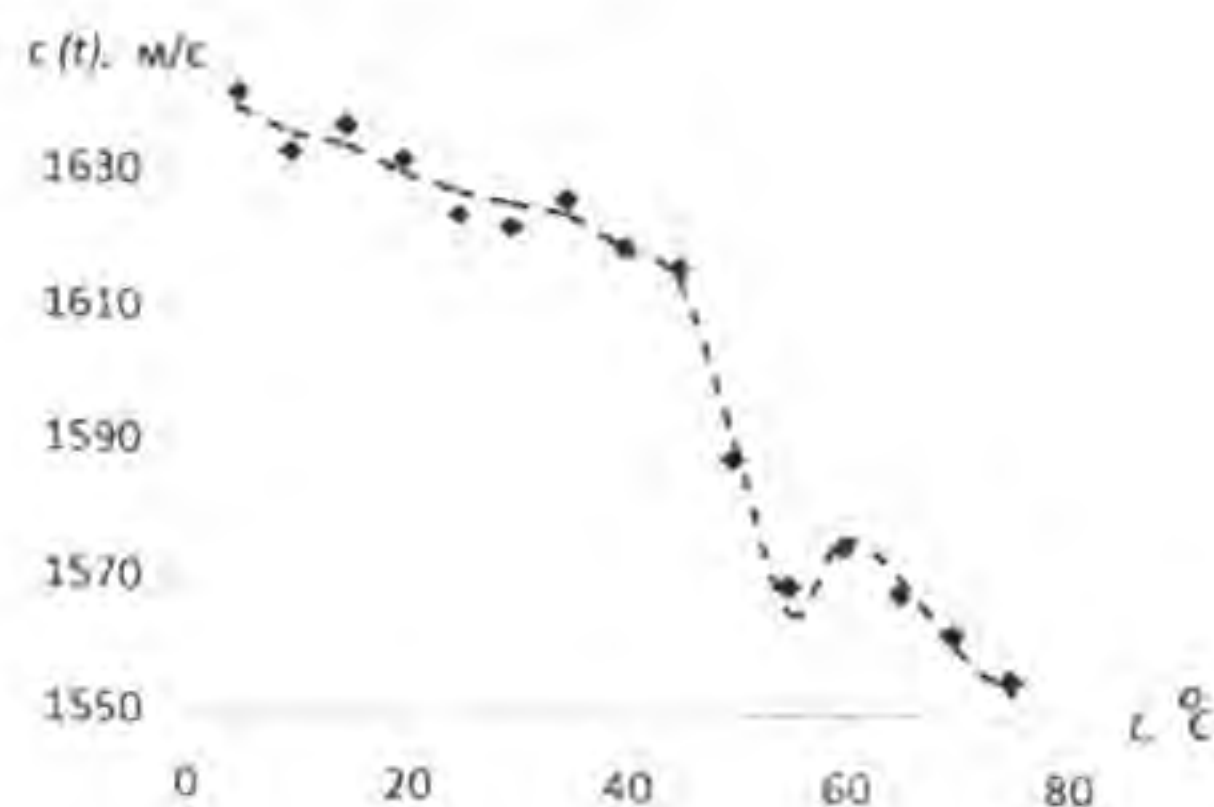


Рис. Температурна залежність швидкості звуку у розчині вода-γ-піколін:

♦ – експериментальні значення, пунктирна лінія – швидкість звуку, розрахована за формулою (7)

Як і очікувалось швидкість звуку має мінімум при температурі $t \sim 55$ °C, що свідчить на користь нашого припущення про кластеризацію розчину, і, відповідно, про наближення розчину до псевдоспінодалі. В околі псевдоспінодалі температурний коефіцієнт швидкості звуку

складає $\frac{\partial c}{\partial T} \sim -4.2$, що можна пояснити перебудовою структури розчину.

Як бачимо з рисунку, обчислена швидкість звуку у розчині вода-γ-піколін цілком узгоджується з експериментальними результатами.

Література

1. Кривохижа С.В., Сабиров Л.М. // Письма в ЖЭТФ. 1980. – Т.31, вып. 12. – С. 746-749.
2. Фабелинский И.Л. Молекулярное рассеяние света, М.:Наука, 1965. – 512 с.
3. Кривохижа С.В., Сабиров Л., Туракулов Я. // ЖЭТФ. 1980. – Т. 78. – С. 1579.
4. Сабиров Л.М., Семенов Д.И., Хайдаров Х.С. // Оптика и спектроскопия. 2007. – Т. 102, № 6. – С. 955-960.
5. Ch.T.Burton // J. Acoust. Soc. Amer., 20, 1948. – P. 186.
6. Арефьев И.М., Старунов В.С., Фабелинский И.Л. // Письма в ЖЭТФ, 1980. – Т.29, вып. 11. – С. 677-681.
7. Вукс М.Ф. Молекулярное рассеяние света в газах, жидкостях и растворах. Л.: Изд-во ЛГУ, 1977. – 320 с.
8. Михайлов И.Г., Соловьев В.А., Сырников Ю.П. Основы молекулярной акустики. –М.: Наука, 1964, 516 с.
9. Маломуж Н.П., Слинчак Е.Л. // ЖФХ. 2007. – Т. 81. № 11. С. 1-6.
10. Маломуж М.П., Слинчак О.Л. // УФЖ. – 2008. Т. 53. № 10. С. 967-971.

Надійшла до редколегії 17.01.2011

УДК 577.3

Войтешенко І.С., Ніколаєнко Т. Ю.,
Булавін Л. А. д.ф.-м.н., проф., Говорун Д.М.,
д.б.н., проф.

**Пружні властивості низькомолекулярних
сполук, що моделюють цукрово-
фосфатний кістяк 2-
дезоксирибополінуклеотидів**

Вперше оцінено релаксовані коефіцієнти жорсткості для кута спірального обертання та торсійних кутів біологічно важливих модельних конформерів кістяка полінуклеотидів у різних їхніх формах. Представлено енергії, необхідні для зміни кутів спірального обертання до нативних значень. Розраховано коливальні дисперсії усіх номенклатурних торсійних кутів кістяка полінуклеотидів.

Представлені значення можуть слугувати кількісною мірою структурної нежорсткості цукрово-фосфатного кістяка 2-дезоксирибополінуклеотидів.

Ключові слова: модельний біцукрово-фосфатний залишок, конформаційна нежорсткість, квантово-механічні дослідження, механічна пружність.

E-mail: isvoiteshenko@gmail.com

Власна жорсткість молекулярних спіралей ДНК зумовлена двома чинниками. Перший із них - вертикальні взаємодії у сусідніх Вотсон-Криківських парах основ, другий - це кулонівське відштовхування заряджених фосфатних груп спіралі. Відносний внесок цих типів взаємодії у загальну жорсткість спіралі є предметом багатьох досліджень, як теоретичних, так і експериментальних. Нині вважають, що значна частка від загальної жорсткості гомополімера припадає саме на пари основ, а фосфатно-фосфатні взаємодії є не вельми суттєвими. Поряд з тим побутує думка про те, що домінуюча роль належить саме фосфатно-фосфатним взаємодіям [1].

Водночас кількісні оцінки коефіцієнтів жорсткості макромолекули ДНК або її структурних фрагментів, які безпосередньо

I. S. Voiteshenko*, T. Y. Nikolaienko,
L. A. Bulavin, Dr.Sc., prof., D.M. Hovorun, Dr.Sc.,
prof.

**Elastic properties of low-molecular
compounds, which model sugar-phosphate
backbone of 2- deoxyribonucleotides.**

For the first time the relaxed force constants for spiral rotation angle and torsion angles have been estimated for biologically important model conformers of polynucleotide's backbone in its different forms. Energies necessary to changes the spiral rotation angles to their native values, are presented. Vibrational dispersions for all nomenclatural torsion angles of polynucleotide's backbone are calculated.

The presented values can serve as quantitative measure for structural flexibility of the sugar-phosphate backbone of 2- deoxyribonucleotides.

Key words: model bisugar-phosphate residue, conformational flexibility, quantum-mechanical investigations, mechanical elasticity.

визначають її пружні властивості, до цього часу є поодинокі.

Об'єкти та методи дослідження

Для кількісного опису конформаційної нежорсткості в біологічних молекулах доцільно оперувати поняттям коефіцієнтів жорсткості для тих чи інших їхніх механічних ступенів вільності [2]. Так для будь-якого важливого, з точки зору дослідника, структурного параметра (наприклад торсійного кута чи кута спірального обертання) відповідний коефіцієнт жорсткості визначається як друга похідна від приросту електронної енергії молекули при зміні цього параметра:

$$K_i = \frac{\partial^2 E}{\partial \tau^2} \quad (1)$$

Таблиця 1

Основні структурно-динамічні та пружні характеристики протонуваних біцукрово-фосфатних модельних залишків кістяка 2'-дезоксирибополінуклеотидів. Позначення (тут і нижче): K - релаксовані коефіцієнти крутильної жорсткості для кута спірального обертання θ та торсійних кутів α , β , γ , δ , ϵ та ζ відповідно (ккал/(моль·рад²)); ΔE_θ - значення енергії необхідне для зміни кута θ до його нативного значення (ккал/моль); ΔG - відносна вільна енергія Гіббса модельного конформера за нормальних умов (ккал/моль); σ - дисперсія відповідного торсійного кута (град), за температури $T = 298,15$ К.

Тип спіралі	B	B	B	A	A	B	B	P	A	B	P	B
Конформація цукру, (-endo)	C2'	C2'	C2'	C3'	C3'	C2'	C2'	C3'	C3'	C2'	C2'	C2'
ΔG	0,87	0,28	1,21	0,00	9,49	3,95	1,94	0,68	12,45	5,75	0,34	3,06
θ	72,4	73,5	73,2	122,7	37,7	74,3	73,1	167,4	63,5	82,9	68,9	69,2
K_θ	6,9	5,9	5,7	4,1	6,6	5,5	6,4	4,7	7,2	21,7	6,5	8,4
ΔE_θ	1,40	1,26	1,20	5,09	0,03	1,22	1,33	13,19	1,09	7,25	1,07	1,40
K_γ	22,8	24,6	25,7	18,6	21,8	23,0	25,1	24,3	28,7	21,0	31,7	29,9
K_β	11,5	11,2	12,2	19,4	7,7	10,8	11,7	27,7	3,3	23,2	21,4	11,2
K_α	9,8	12,4	12,6	1,7	12,5	12,5	13,3	19,5	5,9	8,2	7,0	11,3
K_ζ	14,9	15,4	15,3	10,5	13,1	14,5	15,7	7,4	5,2	11,2	8,2	16,3
K_ϵ	7,6	9,1	10,3	13,8	8,3	6,6	9,4	6,5	2,3	7,8	4,9	12,2
K_δ	15,6	7,7	7,9	17,6	35,9	12,0	7,9	23,0	10,3	12,2	7,6	11,3
σ_γ	9,3	9,0	8,8	10,3	9,5	9,3	8,9	9,0	8,3	9,7	7,9	8,2
σ_β	13,1	13,3	12,8	10,1	16,1	13,5	13,0	8,5	24,3	9,2	9,6	13,4
σ_α	14,4	12,8	11,6	31,6	12,9	12,8	12,4	10,3	18,3	15,7	17,0	13,4
σ_ζ	11,5	11,3	11,4	13,7	12,3	11,7	11,2	16,4	19,5	13,4	15,5	11,0
σ_ϵ	16,1	14,8	13,9	12,0	15,4	17,4	14,5	17,4	29,4	16,0	20,0	12,7
σ_δ	11,5	16,0	15,9	10,6	7,4	12,8	15,8	9,3	13,8	12,8	16,1	13,2

Згадана методика розроблена і детально описана в попередньому нашому дослідженні [3]. У представленій роботі за її допомогою та результатами квантово-механічних розрахунків нами визначено величини релаксованих коефіцієнтів жорсткості K для кута спірального обертання θ і торсійних кутів α , β , γ , δ , ϵ та ζ [1] в усіх модельних біцукрово-фосфатних залишках, що є твірними кістяка біологічно важливих конформацій ДНК. Також представлено значення енергій ΔE_θ , необхідних для зміни кутів спірального обертання до нативних величин, які спостерігаються у правозакрученій двоспіральній ДНК. Визначено також дисперсії σ [4], обумовлені тепловими коливаннями ядер для вищезгаданих торсійних кутів.

Об'єкти дослідження отримані вичерпним конформаційним аналізом біцукрово-фосфатних залишків в різних формах: негативно зарядженого (аніонного) і електронейтральних. При цьому з повної конформаційної множини виокремлено біологічно найважливіші та енергетично найбільш значимі конформери в аніонній [5], протонуваній [6] та нейтралізованій іоном натрію [7] випадках.

Результати та їхній аналіз

Як видно з Табл. 1 у нейтралізованих протоні модельних конформерах, структурна м'якість спостерігається здебільшого для структур, що продукують Р- подібні спіралі полінуклеотидних ланцюгів. Найм'якший конформер з представленої групи є енергетично найвигідніший А- подібний цукрово-фосфатний залишок з кутом спірального обертання 122,7°. Цю його властивість можна пояснити малим значенням коефіцієнта жорсткості для торсійного кута α , що складає лише 1,7 ккал/(моль·рад²). При цьому відповідне значення дисперсії має значення 31,6° за температури 298,15 К, що є максимальним з-поміж всіх кутів у досліджених модельних сполуках.

Усі протонувані форми модельних сполук, що продукують спіральні структури біологічно важливих В- подібних форм ДНК, мають значення коефіцієнтів жорсткості кута спірального обертання, що лежать у діапазоні від 5,5 до 8,4 ккал/(моль·рад²). При цьому величини енергій, які необхідно затратити для реалізації у них нативних значень кута спірального обертання складають 1,20-1,40 ккал/моль. Натрієві аналоги модельних сполук (Табл. 2), у свою чергу, мають відповідні значення коефіцієнтів жорсткості 7,0÷10,9 ккал/(моль·рад²), а енергії, необхідні для

Таблиця 2

Основні структурно-динамічні та пружні характеристики натрієвих біцукрово-фосфатних модельних залишків кістяка 2'-дезоксирибополінуклеотидів (позначення ті ж самі, що і в Табл. 1)

Тип спіралі	B	P	B	P	A	B	A	B	B	B	A	B
Конформація цукру, (-endo)	C2'	C2'	C2'	C3'	C3'	C2'	C3'	C2'	C2'	C2'	C3'	C2'
ΔG	5,80	0,72	8,95	1,02	10,48	13,14	15,01	8,47	9,70	6,66	0,00	7,78
θ	76,1	136,2	73,4	163,1	77,6	70,9	44,5	72,8	73,7	75,6	150,2	72,6
K_θ	7,0	5,6	9,9	12,4	5,0	5,6	5,5	9,3	10,9	7,2	22,7	10,5
ΔE_θ	1,72	8,50	2,12	32,36	1,57	1,04	0,13	1,91	2,37	1,73	48,27	2,16
K_γ	25,3	32,3	26,5	12,9	17,8	28,3	21,9	25,8	25,8	25,2	33,1	26,2
K_β	14,7	35,3	13,4	28,6	10,7	4,6	7,6	13,0	13,2	12,8	41,3	14,8
K_α	12,0	25,5	12,7	22,5	13,5	5,6	12,7	13,3	13,4	13,1	23,0	12,6
K_ζ	16,1	9,9	19,3	11,2	10,5	18,4	19,5	17,6	18,9	18,3	21,7	17,0
K_ϵ	12,3	7,3	11,6	25,4	9,7	19,9	9,1	12,6	12,6	12,9	32,1	12,9
K_δ	8,5	9,6	15,2	42,6	2,9	21,3	20,7	15,1	16,1	8,7	42,8	15,7
σ_γ	8,9	7,8	8,6	12,4	10,5	8,7	9,8	8,8	8,8	8,9	7,7	9,2
σ_β	11,6	7,5	12,3	8,3	13,6	20,7	16,1	12,4	12,3	12,4	6,9	11,7
σ_α	12,8	8,8	12,5	9,4	12,1	18,9	12,5	12,2	12,1	12,3	9,5	12,7
σ_ζ	11,1	14,1	10,1	13,3	13,7	10,4	10,1	10,6	10,2	10,4	9,5	11,0
σ_ϵ	12,7	16,4	13,0	8,8	14,3	10,0	14,8	12,5	12,5	12,4	7,9	12,5
σ_δ	15,3	14,4	11,4	6,8	26,0	9,6	9,8	11,5	11,1	15,1	6,8	11,4

реалізації їх нативного стану лежать у діапазоні $1,72 \div 2,37$ ккал/моль. Це дозволяє зробити висновок про те, що при нейтралізації фосфатних груп модельних конформерів іонами натрію нуклеотидний цукрово-фосфатний кістяк у B-формі є жорсткішим порівняно з випадком нейтралізації протонуванням.

Існують також конформери з досить великими значеннями коефіцієнта жорсткості у порівнянні з описаними вище. Так, для згаданих модельних протонованих сполук конформер з кутом спірального обертання $82,9^\circ$ та C2'-endo цукром, що продукує найменш енергетично вигідну B-подібну спіраль ДНК, має коефіцієнт жорсткості K_θ 21,7 ккал/(моль·рад²). Енергія, необхідна для набуття кута спірального обертання нативного значення складає 7,25 ккал/моль. Серед групи конформерів, що нейтралізовані іоном натрію, найбільша жорсткість 22,7 ккал/(моль·рад²) зафіксована для A-подібної модельної сполуки. Ця величина коефіцієнта крутильної жорсткості разом зі значною відмінністю кута спірального обертання $150,2^\circ$ вимагає найбільшої, серед усіх досліджених біцукрово-фосфатних залишків, енергії 48,27 ккал/моль для реалізації нативної A-форми ДНК.

Для енергетично найвигідніших сполук з C2'-endo та C3'-endo цукрами, що продукують P-подібні спіралі, нейтралізовані іонами натрію, як і у випадку протонованих біцукрово-фосфатних залишків, спостерігаються порівняно невеликі значення коефіцієнтів крутильної жорсткості: 5,6

та 12,4 ккал/(моль·рад²) відповідно. Схожа ситуація зафіксована і для аніонних модельних залишків (Табл. 3), де P-подібні модельні конформери є структурно м'якшими у порівнянні з аналогічними A-подібними формами.

Загалом, модельні біцукрово-фосфатні залишки з C3'-endo цукрами, що продукують A-подібні полінуклеотиди, але у жодному з випадків не мають повної відповідності всіх торсійних кутів до їхнього природного набору, характеризуються найширшим діапазоном значень коефіцієнтів жорсткості $4,1 \div 22,7$ ккал/(моль·рад²) у порівнянні з твірними B- та P-подібних форм.

Коефіцієнт релаксованої крутильної жорсткості для кута спірального обертання чи своєю природою є величиною, що має суттєво колективний характер і для біцукрово-фосфатного залишку залежить від шести торсійних кутів. Відповідні значення релаксованих коефіцієнтів крутильної жорсткості для всіх шести торсійних кутів наведено в Табл. 1 – 3. Варто зауважити, що їхні значення для кутів β , γ , δ та ϵ лежать у межах $1,5 \div 42,8$ ккал/(моль·рад²) і подібні до аналогічних величин для біологічно важливих конформерів молекули 1',2'-дидезоксирибози [4]. Схожа поведінка спостерігається і для дисперсій конформаційних ступенів вільності обох модельних сполук. Так, для 1',2'-дидезоксирибози ці величини становлять

Таблиця 3

Основні структурно-динамічні та пружні характеристики аніонних біцукрово-фосфатних модельних залишків кістяка 2'-дезоксирибополінуклеотидів (позначення ті ж самі, що і в Табл. 1)

Тип спіралі	P	P	A	A
Конформація цукру, (-endo)	C2'	C2'	C3'	C3'
ΔG	3,41	15,62	8,59	0,00
θ	75,5	54,0	45,6	72,2
K_{θ}	6,3	8,0	12,2	10,7
ΔE_{θ}	1,49	0,40	0,35	2,65
K_{γ}	20,7	18,7	22,1	5,4
K_{β}	13,5	14,9	6,8	1,5
K_{α}	3,9	7,2	6,7	5,6
K_{ζ}	19,2	9,9	11,4	13,5
K_{ϵ}	9,3	2,9	15,5	21,7
K_{δ}	9,9	14,5	33,0	23,1
σ_{γ}	9,8	10,3	9,5	19,2
σ_{β}	12,1	11,5	17,1	36,1
σ_{α}	22,4	16,5	11,8	18,8
σ_{ζ}	10,1	14,1	13,2	12,1
σ_{ϵ}	14,6	26,3	11,3	9,6
σ_{δ}	14,1	11,7	7,8	9,3

9,2÷30,2°, а для біцукрово-фосфатного залишку 6,8÷36,1°.

Висновки

Вперше показано, що найширший діапазон значень коефіцієнтів жорсткості для кута спірального обертання спостерігається для A-подібних модельних конформерів, на відміну від B- та P- подібних структур.

Вперше виявлено, що при нейтралізації фосфатних груп іонами натрію нуклеотидний цукрово-фосфатний кістяк B- подібних форм є жорсткішим порівняно з випадком, коли нейтралізація забезпечується протонуванням,

Список використаних джерел

1. Зенгер В. Принципы структурной организации нуклеиновых кислот. М.: Мир, 1987. – 584 с.
2. Булавін Л.А., Ніколаско Т.Ю., Говорун Д.М. Структурна нежорсткість молекули ортофосфорної кислоти: неемпіричне квантово-механічне дослідження // Доповіді НАН України. – 2009. – №10. – С. 80 – 87.
3. Ніколаско Т.Ю., Войтешенко І.С., Булавін Л.А., Говорун Д.М. Визначення коефіцієнтів жорсткості для колективних

ступенів вільності біологічних молекул // Вісник Київського університету серія фіз. мат. науки – 2010. - 2 – С. 279-283.

4. Булавін Л.А., Ніколаско Т.Ю., Говорун Д.М. Структурна мінливість біологічних молекул, спричинена їхніми нормальними коливаннями // Доповіді НАН України. – 2010. – №7. – С. 61 – 69.
5. Войтешенко І.С., Жураківський Р.О., Булавін Л.А., Говорун Д.М. Конформаційний аналіз низькомолекулярної моделі аніонного цукрово-фосфатного ланцюга ДНК // Доповіді НАН України. – 2010. – №11. – С.158- 166.
6. Войтешенко І.С., Жураківський Р.О., Булавін Л.А., Говорун Д.М. Повний конформаційний аналіз низькомолекулярної електронейтральної моделі цукрово-фосфатного ланцюга ДНК // Доповіді НАН України. – 2011 - №6 – (друкується).
7. Войтешенко І.С., Жураківський Р.О., Булавін Л.А., Говорун Д.М. Конформаційний аналіз низькомолекулярної моделі цукрово-фосфатного ланцюга натрієвої ДНК // Доповіді НАН України. – 2011 – №7 – (друкується).

УДК 518.4

Гречко Л.Г., д. ф.-м.н., проф.,
Породько Л. В., аспірант,
Лерман Л. Б., к.т.-н.н.,
Семчук О. Ю., к.ф.-м.н.

**Просторовий та часовий розподіл
температури на поверхні твердого тіла,
викликаний одномірною лазер
інтерференційною картинкою**

Робота присвячена вивченню нелінійних та нерівноважних явищ в конденсованих середовищах обумовлених дією зовнішнього електромагнітного поля та структурою поверхні. Розглянуто теплові ефекти, викликані впливом інтенсивного лазерного випромінювання, і проведено дослідження механізму утворення лазер-індукованих поверхневих структур та описано новий ефект – утворення поверхневих наноостровів міді при лазерному опроміненні поверхні термомагнітних напівпровідників типу Cu_2S .

Ключові слова: лазер-індуковані,
ефект – утворення, термомагнітні.

*E-mail: lilphys@mail.ru.

Introduction

Recently interest to research of any sort of the surface structures in solid state created by means of laser radiation and laser beams interference pattern has essentially increased [1,2]. Perspective direction becomes also laser ablation, which can be in addition used for formation of nano-sized particles of a solid electrolyte [3]. Pulse laser ablation of solid targets in the gas phase has been widely used for the preparation of various nanostructures materials such as nanoparticles, nanotubes and nanocomposites [4-6]. The study of laser induced heating and melting are of great importance for achieving high quality materials processing with lasers [7]. However until

L.G. Grechko, Dr. Sci., Prof.,
L.V. Porodko, PhD stud.,
L.B. Lerman, PhD,
O.Yu. Semchuk, PhD.

**Spatial and temporal distribution of temperature
on a surface of the solid state, caused by one
dimensional laserinterference pattern**

Considered the thermal effects caused by influence of the intensive laser radiation on a outer surface of solid state, it is carried out research of the mechanism of formation the laser-induced periodic structure and it is described new effect - formation a surface nanoislands of copper at a laser irradiation of a surface of thermomagnetic semiconductors of Cu_2S type.

Key Words: laser-induced, effect – formation,
thermomagnetic

recently poorly investigated there is a question on features of spatial and temporal distribution of temperature at laser heating a surface. Most full this question has been investigated in [8]. In this paper, the spatial and temporal temperature distribution in a material is calculated by the Green function method for different cases of laser radiation: pulsed laser and continuous laser wave.

Results and discussion

Let us consider process of heating of a surface of a solid state by one-dimensionally laser beam interference pattern, as it presented at the Fig.1.

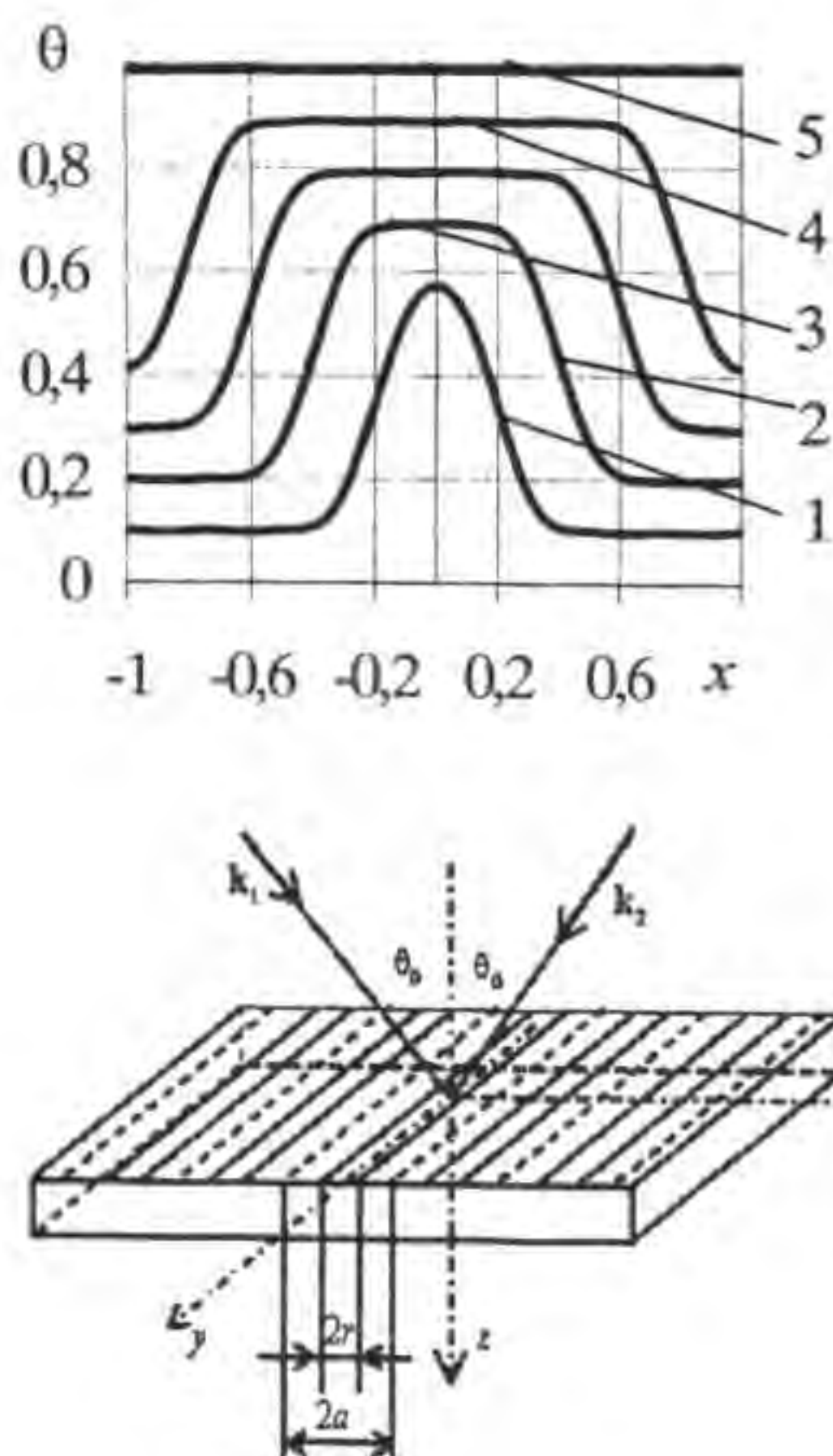


Fig.1. Picture of interferential dimensional on the solid surface created by coherent laser beams.

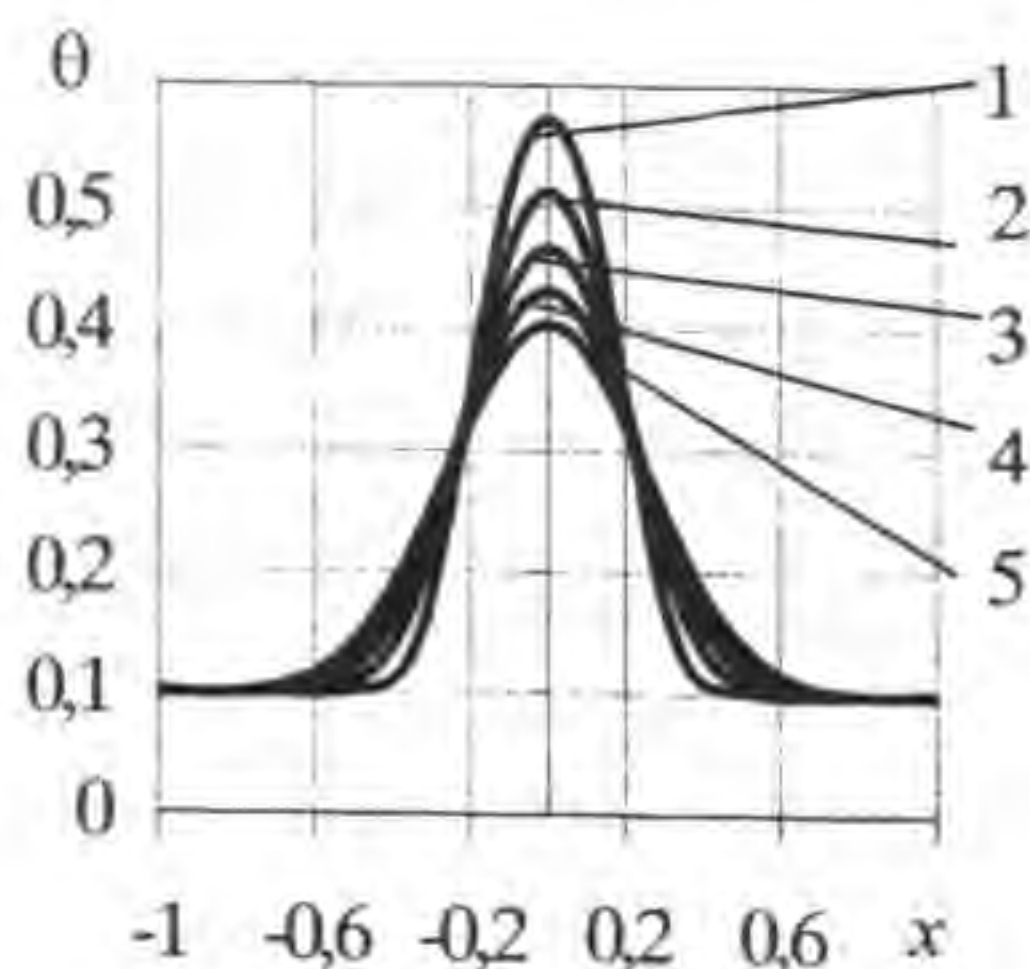


Fig.2. Distribution of dimensionless temperature θ on the surface of solids on average (a) diagonal ($x=y$) (b) cross sections one-dimensional periodic structure with some values for $t=0,005$ the width of the beam: 1 - $r/a=0.2$; 2 - $r/a=0.4$; 3 - $r/a=0.6$; 4 - $r/a=0.8$; 5 - $r/a=1.0$.

We shall assume that thermophysical characteristics of medium no depended on temperature and coordinates. In this case the heat diffusion equation

describing distribution of temperature T in a half-space $0 < z < \infty$ can be written as

$$\frac{\partial T}{\partial t} = \chi \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) + \alpha I(x, y, z, t), \quad (1)$$

where t is a time, x, y, z - Cartesian coordinates and x, y lies in the surface of a solid, χ is the thermal diffusivity, α is the absorption coefficient, I is the power density of laser beam.

We assume that a laser beam is presented of the Gaussian intensity distribution

$$I(x, y, z, t) = H(x, y) I_0 (1 - R) e^{-\alpha z} \exp \left\{ -\frac{(x^2 + y^2)}{r^2} \right\} f(t), \quad (2)$$

where I_0 is the maximum intensity of laser beam, R is the capacity reflection coefficient, r is the radius of one laser beam, $H(x, y) = 1$ if $x^2 + y^2 \leq r^2$, and $H(x, y) = 0$ in opposite case, $f(t)$ is the function, which describes the distribution of a laser impulse in time.

The distribution of dimensionless temperature on surfaces of a solid in the middle section of cell for various radiuses of a beam at $t = 0.005$ is presented in the Fig. 1a. The distribution of temperature velocity on surfaces of a solid in the middle section of cell when $r/a = 0.2$ and $f(t) = H(t)$ for various values of time in the Fig. 1b.

Developed by us now use the calculation method for analyzing the process of laser heating of the surface. Under the influence of laser radiation on matter, the light energy that is absorbed, eventually thermalizuyetsya, is converted into energy of chaotic motion of atoms and molecules. In other words, the optical effect leads to a heated substance. Significant enough to raise the temperature at absorption of powerful laser radiation causes phase transitions - goals often cascades of phase transitions such as ionization or gas, its evaporation from the surface, spraying, and after the end of the laser pulse - hardening and others. When the laser heating process description of substances are two essential conditions. First, because the penetration depth of light in optical medium heat sources are usually bulk, is distributed in the bulk environment, not localized, such as its boundaries, as happens in problems of conventional thermal heating. Second, the selection of optical energy is uniform on the interaction volume due to decreasing light intensity as its penetration depth in the environment. The latter leads to spatially inhomogeneous heating substances and processes is heat and mass transfer between different sites of the environment. The process of heat transmission is the transfer of thermal energy. For the two beams interference pattern intensity distribution is given by the relation:

$$I(x) = 2I_0 \{ \cos [(4\pi x / \lambda) \sin (\theta_0 / 2)] + 1 \}, \quad (3)$$

where I_0 - intensity laser light field, λ - wavelength radiation, which generates laser θ - climbing angle beams. Spatial frequency formed a picture defined wavelength laser that induces the picture, and angle of ascent beams - $d = \lambda / (2 \sin \theta_0 / 2)$.

We conducted a study of thermoelectric semiconductor surface before and after treatment laser radiation. The difference in reflection coefficient amounted agenda 2.5%. This may be due to the fact that on the surface of thermoelectric semiconductor processes due to laser ablation, is the formation of copper islands. Measurements by Auger spectroscopy showed that there is increasing concentration of copper increases as the intensity of the laser beam. On Fig.3 are the concentration dependence of copper atoms (in relative units), which is allocated on the surface of the semiconductor type Cu_2S termomahnitnoho the intensity of laser pulse.

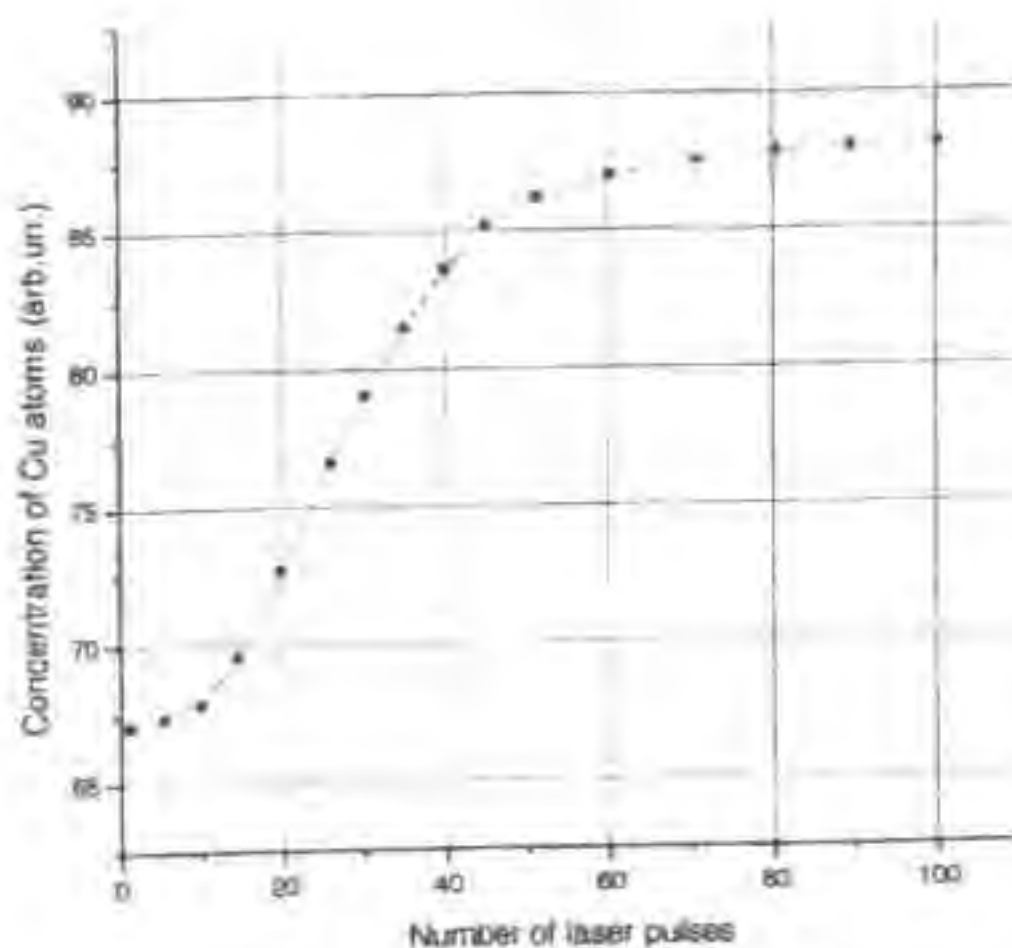


Fig.3. Dependence of concentration of copper atoms (in relative units) that allocated on the surface of the semiconductor type Cu_2S termomahnitnoho the intensity of laser pulse.

So, we have shown that the impact of the two beams on the surface termomahnitnoho semiconductor microstructures are formed. Their properties depend on: the wavelength of laser radiation and ablative characteristics of the semiconductor. Second, under the influence of laser beams is the formation of copper islands on the surface Cu_2S . In addition, thermal diffusion for the period of laser radiation does not affect the formation of interference laser structure during which the order of 5-10 microns. The method of calculation can be used to improve the method of laser lithography.

Conclusion

1. The process of heating solid surface several laser beams that create on its surface odnomirnu interference pattern. The equation that describes this process. For one-dimensional periodic structures with Gaussian intensity distribution beam the numerical solution of heat equation - found spatially periodic temperature distribution on the surface of solids. Found that the stationary distribution of surface temperature is set very quickly during the procedure $t = 0.2$.

2. Shown that the impact of two laser beams of large intensity on the surface of the semiconductor

termomahnitnoho formed spatially periodic microstructure having between 5-10 microns, thus found that thermal diffusion bonding does not significantly affect their formation. Their properties depend on several factors: the wavelength of laser radiation and ablative characteristics of the semiconductor. Also found that the impact of laser radiation on the surface of the semiconductor termomahnitnoho Cu_2S formed nanoostriivtsi bronze.

Developed in this section technique can be used in the development and improvement of laser lithography.

References

1. Aktag, Jor. Appl.Phys., 99 (2006) 093901.
2. O.Yu. Semchuk, V.N. Semioshko, L.G. Grechko. M. Willander and M- Karlsteen, Appl.Surf-Sci., 252 (2006) 4759.
3. K. Takada, R.Ma, M. Osada, N.Ohta, L.Zhang and T. Sasaki, Jor. Of Power Sources, 146 (2005) 703.
4. F. Mafune, J. Kohno, Y. Takeda, T. Kondow, and H. Sawade, J.Phys.Shem., B 104 (2000) 8333.
5. Y.H. Chem, and C.S. Yet, Chem Commun., 4 (2001) 371.
6. C.H. Liang, Y. Shimizu, T. Ssaski, and N. Koshizaki, J. Phys.Shem, B,107 (2003) 8645.
7. M.K. El-Adavi, and E.F. El-Shehawey, J. Appl. Phys., 60 (1986) 2250.
8. J.C. Conde, F. Lusquinos, P.Gonzalez, B. Leon, and M. Perez-Amor, Vacuum 64, (2002) 359.

Надійшла до редколегії 26.11.2010

УДК 537.32

Копань В.С., д.ф.-м. н., проф.,
Хуторянська Н.В. пров. інж.,
Копань Т.В., інж.

Термоелектричний адсорбційний ефект в нікелевому порошку

Досліджено в діапазоні 293-333 К термоелектрорушійну силу (ТЕРС) нікелевого порошку після адсорбції різних рідин. ТЕРС зменшується в ряді рідин: H_2O+HCl (50% об.), алкоголь, уайт-спірит. Абсолютна ТЕРС $E_{\pi}=-10$ мкВ/К зростає до $E_{\pi}=-27,6$ мкВ/К завдяки адсорбції рідин і тунелюванню електронів через зазори між частинками порошку. В магнітному полі 5 кЕ $E_{\pi}=-11,3$ мкВ/К. Після адсорбції 2% об. H_2O+HCl термоелектрична добротність нікелевого порошку збільшується семикратно.

Ключові слова: термоелектрична сила, нікелевий порошок, адсорбція, термоелектрична добротність.

* E-mail: @pulsary.com.ua

Вступ. В адсорбційних аналізаторах газів і рідин використовують плівки, мікродроти, волокна та інші об'єкти з великою питомою поверхнею [1]. Адсорбція газів і рідин змінює електроопір, наприклад. За його величиною визначають наявність домішок в середовищі, що аналізується. Порошки використовують для цих цілей у спеченому пористому стані. Насипні порошки мало придатні, бо їх електроопір змінюється не лише за рахунок адсорбції, а й від дії вібрацій (звуку, наприклад). Невідомі літературні джерела, в яких би повідомлялось про вплив адсорбції на термо-е.р.с. насипних порошків [2]. Дослідження цього явища цікаве з точки зору фізики і корисне для практичного використання в аналізаторах рідин на домішки, бо термо-е.р.с. чутлива до домішок, як і опір, але мало чутлива до вібрацій, бо останні впливають на обидві вітки термopари, внаслідок чого природи абсолютних термо-е.р.с. взаємно компенсуються.

Мета роботи – дослідження впливу адсорбції на термо-е.р.с. та термоелектричну добротність насипного нікелевого порошку.

Методика досліджень. Термо-е.р.с. вимірювали за допомогою потенціометра з

V.S. Kopan', Dr of Sci, Prof.,
N.V. Hutoryanska, eng.,
T.V. Kopan', eng.

The thermoelectric adsorption effect in Nickel powder

The thermoelectric powder (TEP) of Nickel powder after of liquids adsorption was measured between 293 K and 333 K. The TEP is decreasing in series of liquids: H_2O+HCl (50% vol.), alcohol, whitespirit. The absolute TEP $E_{\pi}=-10$ mkV/K arise to $E_{\pi}=-27,6$ owing to adsorption of liquids and tunneling of electrons across chinks between particles of powder. In field of magnet 5 kE $E_{\pi}=-11,3$ $\mu V/K$. After adsorption of 2% vol. of H_2O+HCl the thermoelectric quality of Ni-powder is seven times as large.

Key Words: thermoelectric power, adsorption, nickel, powder.

малим (20 Ом) входним опором та ціною поділки 10^{-8} В.



Рис.1. Форма частинок нікелевого порошку

Еталонна вітка термopари була виготовлена з нікелевої фольги, робоча – з нікелевого порошку, засипаного в пластмасову трубку. Холодні кінці термopари були при температурі $20\pm 0,1^{\circ}C$, гарячий «спай» – при $62\pm 0,05^{\circ}C$. Краплі адсорбату подавали в порошок

через отвір посередині довжини пластмасової трубки. Детально методика вимірювання пояснена в [4, 5, 7].

Нікелевий порошок (99,97%) був виготовлений розпилюванням напильником стержня (99,99%). Сильно деформований напильником порошок має багато гострих країв. Це сприяє тунелюванню електронів через зазори між частинками порошку. Для дослідів використовувався порошок (рис.1), просіяний через сито з діаметром комірок 40 мкм, густиною $7 \pm 0,5 \text{ г/см}^3$ і пористістю 0,21.

Експериментальні результати та їх обговорення.

Термо-е.р.с. термopари $E=7,5 \text{ мкВ/К}$ на початку дослідів (кругла перша точка на рис.2); після адсорбції 2% об. етанолу зменшується до $E=4$ (мінімум 1 на рис. 2), потім, внаслідок реадсорбції етанолу зростає до $E=8,2$. Початкові значення E (круглі точки) в циклі зростають зі збільшенням номера циклу.

В роботі [5] було показано, що в насипних порошках E генерується переважно за рахунок тунелювання зарядів через потенціальні бар'єри (зазори між частинками порошку) висотою U_2 і шириною ω , причому прозорість бар'єру

$$\frac{I_3}{I_1} = 16 \frac{E_k}{U_2} \exp(-2\gamma\omega), \quad (1)$$

$$\gamma = \frac{1}{\hbar} (2mU_2)^{1/2}, \quad (2)$$

де I_1 та I_3 – потоки електронів, що налітають на потенціальний бар'єр та тунелюють через нього, m – маса електрону, $E_k = kT = 0,024 \text{ eV}$ – кінетична енергія електрону (при $T=313 \text{ K}$), яку він має додатково біля енергії Фермі.

Зменшення E при адсорбції можна пояснити зменшенням U_2 та ω , коли зростає I_3/I_1 , тобто, коли в порошковій вітці потік електронів наближається до потоку в еталонній вітці. Поступове наростання початкових значень E в циклах адсорбції засвідчує про зростання U_2 та ω .

Оцінимо термоелектричну добротність порошку:

$$Z_{\Pi} = \frac{E_{\Pi}^2}{k_{\Pi} \rho_{\Pi}}, \quad (3)$$

де E_{Π} – абсолютна термо-е.р.с., ρ_{Π} і k_{Π} – питомий опір та теплопровідність порошку.

Порівняємо Z_{Π} з добротністю Z_{Φ} нікелевого еталону з фольги. Відповідні числа:

$E_{\Phi} = -17,6 \text{ мкВ/К}$ (відносно Pb),
 $\rho_{\Phi} = 6,8 \cdot 10^{-6} \text{ Ом*см}$, $k_{\Phi} = 0,93 \text{ Вт/град*см}$,
 $E_{\Pi} = -10 \text{ мкВ/К}$, $\rho_{\Pi} = 9 \cdot 10^{-6} \text{ Ом*см}$,
 $k_{\Pi} = 0,2 \text{ Вт/град*см}$. Тоді одержимо
 $Z_{\Pi} = 0,6 \cdot 10^{-4} \text{ K}^{-1}$, $Z_{\Phi} = 0,5 \cdot 10^{-4} \text{ K}^{-1}$, $Z_{\Pi}/Z_{\Phi} = 1,2$.

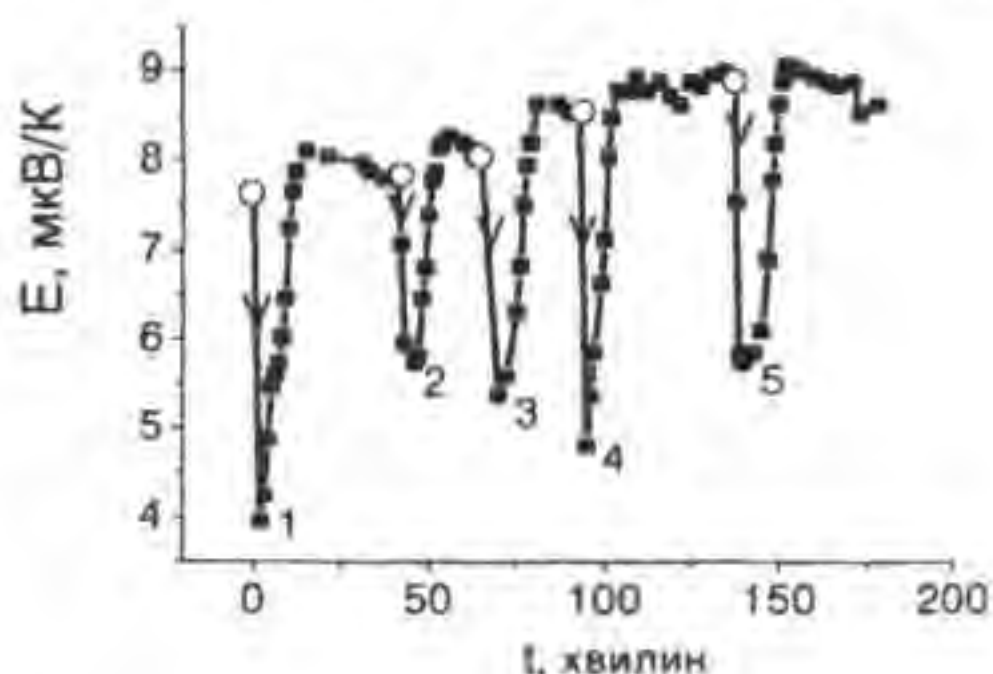


Рис.2. Залежність термо-е.р.с. від часу та циклів 1-5 адсорбції 2% об. етанолу.

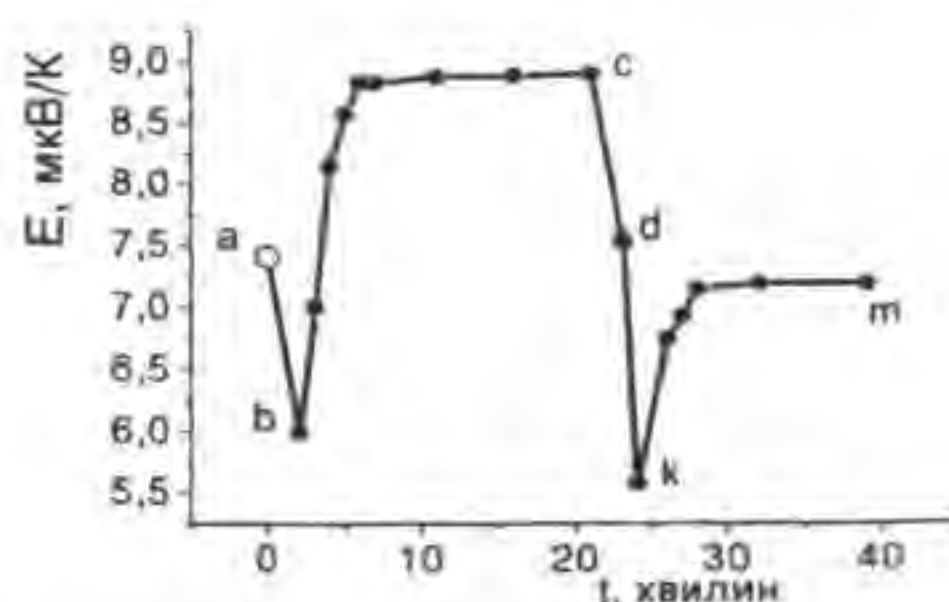


Рис.3. Залежність термо-е.р.с. E від часу адсорбції 2 % об. етанолу (abc і dkm) та впливу магнітного поля (cd).

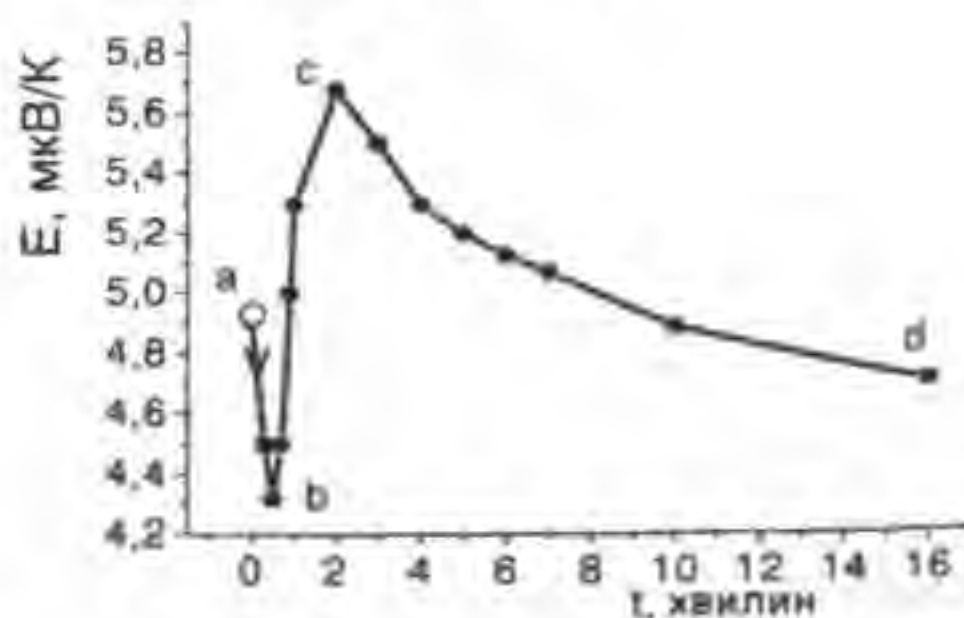


Рис.4. Зміна термо-е.р.с. при адсорбції (ab) і десорбції (bcd) порошком 2% об. уайт-спіриту.

В хороших напівпровідникових матеріалах $Z=3,4 \cdot 10^{-3}$, планується збільшити Z у 1,5-2 рази [6] за рахунок переходу на нанокристалічні матеріали.

Таким чином, нікелевий порошок на 20% має більшу добротність за добротність нікелевої фольги. Це обумовлено малою теплопровідністю k завдяки зменшенню проходження фононів через зазори між частинками. Спроба збільшити Z_T за рахунок заміни монолітних віток термопар на порошкові в напівпровідникових високодобротних сполуках поки не увінчалась успіхом [6]. Можливо тому, що вивчалися компактовані, а не насипні порошки.

Оскільки в (1) у чисельнику кінетична енергія електрону $E_e = kT$, то I_2/I_1 пропорційне температурі T . В нікелевій фользі густина потоків електронів I_2 та I_1 не залежить від T , бо немає потенціальних бар'єрів, крізь які відбувалось би тунелювання, бо густина електронів провідності не залежить від T . В напівпровідниках – експоненційна залежність концентрації носіїв електрики від T . Диференціальна термо-е.р.с. термопари Ni (моноліт) – Ni (порошок) $E = +(7-9)$ мкВ/К (рис.2) в 70-90 разів більша за $E = 0,1$ мкВ/К [7] термопари Ni (моноліт) – Ni (деформована фольга). Це можна пояснити малою проникливістю тунельних бар'єрів. Так, при густині електронів провідності в Ni $\sim 10^{23}$ см⁻³ для електронів з енергією близько енергії Фермі (саме вони генерують термо-е.р.с.) $U_F \sim 2,5$ еВ [9], коли $\omega = 0,4$ нм в системі метал-вакуумна щілина-метал. Тоді, згідно (1) і (2), $I_2/I_1 \approx 10^{-4}$, мала (10^{-4}) доля електронів генерує термо-е.р.с. шляхом тунелювання. В дійсності – доля ще менша, бо на поверхні частинок порошку є оксиди. Отже, насипні металеві порошки займають проміжне положення між монолітними металом і напівпровідником. Тому в них має бути підвищена добротність.

На рис.3 показано звичайну криву abc адсорбції етанолу. В точці c на порошкову вітку подіяли магнітним полем $H = 5$ кЕ. Термо-е.р.с. зменшилась на $\Delta E = 1,3$ мкВ/К. В нікелевій фользі термо-е.р.с. зростає при накладанні магнітного поля на $\Delta E = 0,2$ мкВ/К [7]. Отже, спадання відбувається за рахунок злипання частинок порошку під дією магнітостатичних сил, внаслідок чого звужується бар'єр ω . Це спадання переважає зростання E внаслідок деформації частинок порошку.

Магнітне поле опускає на величину $\Delta E = 1,3$ мкВ/К криву адсорбції dkm (рис.3) і збільшує адсорбційну частину dk на 30% у порівнянні з ab. Це означає, що d-електрони поверхневих атомів практично не діють на молекули етанолу. Адсорбція 2% об. уайт-спіриту (рис.4) менше впливає на термо-е.р.с.,

ніж адсорбція етанолу, бо уайт-спірит – хімічно нейтральний до нікелю ізолятор. Адсорбція порошком електроліту (рис.5) сильно змінює термо-е.р.с., бо відбуваються хімічні реакції. Тому маємо справу з хемосорбцією. Крім того, електроліти теж генерують термо-е.р.с. [3].

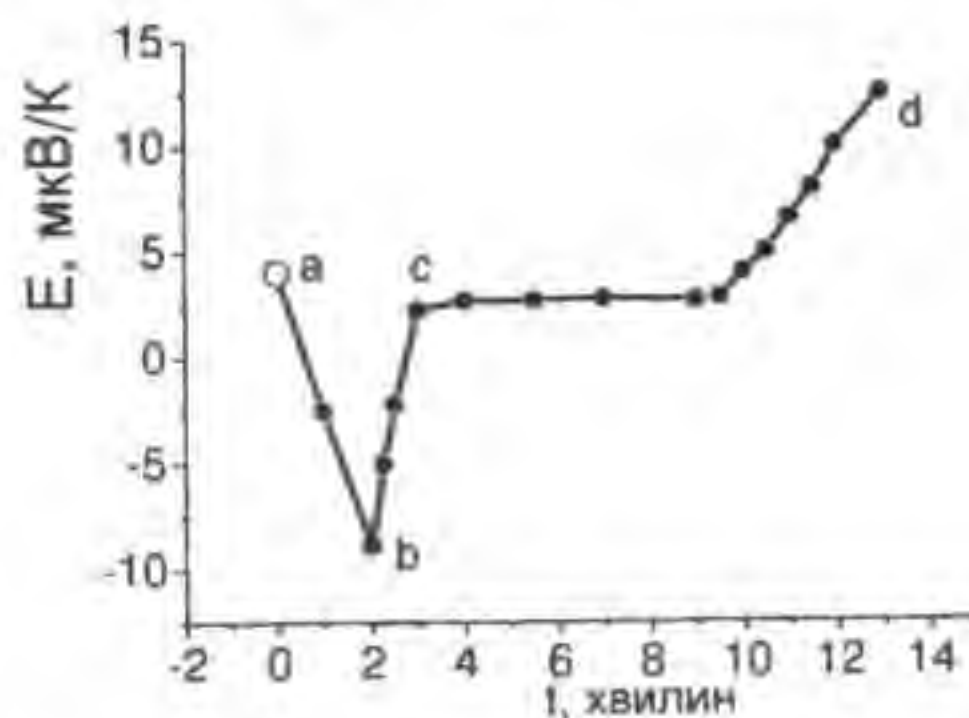


Рис.5. Зміна термо-е.р.с. порошку при адсорбції (ab) і десорбції (bcd) 2% об. HCl+H₂O (50% об.)

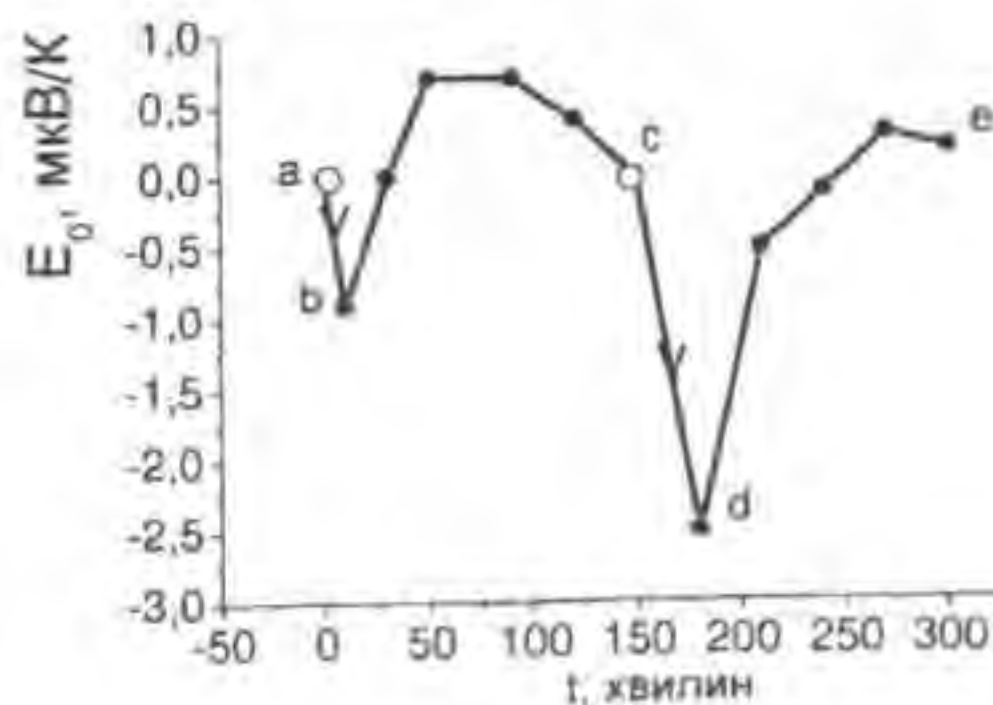


Рис.6. Часова залежність електрорушійної сили E_0 , що виникає в термопарі (за відсутності градієнту температури) при адсорбції 2% об. етанолу: ab і cd – адсорбція в двох циклах, bc і de – десорбція.

В роботі [8] показано, що магнітне поле значно впливає на електроопір композиції ПВХ – нанокристалічний нікель і на адсорбцію нею ацетону, толуолу і етанолу. Дія магнітного поля обумовлена змінами в областях тонких перемичок неперервних нескінченних кластерів частинок Ni [8].

Оцінимо, як впливає адсорбція, наприклад, HCl+H₂O на термоелектричну добротність нікелевого порошку. Термо-е.р.с. в точці b (рис.5) $E = -10$ мкВ/К відносно нікелевої фольги, абсолютна термо-е.р.с. якої становить

$E_{\text{ф}} = -17,6$ мкВ/К. Тому $E_{\text{п}} = -17,6 \cdot 10 = -176$. Інші параметри порошку в стані, що відповідає точці b (рис.5) такі: $\rho_{\text{п}} = 8 \cdot 10^{-6}$ Ом*см, $k_{\text{п}} = 0,3$ Вт/град*см. Тоді, згідно (3), $Z_{\text{п}} = 3,6 \cdot 10^{-4} \text{ К}^{-1}$, а $Z_{\text{п}}/Z_{\text{ф}} = 3,6/0,5 = 7,2$.

Отже, можна підібрати такі адсорбати, що збільшують абсолютну термо-е.р.с. порошку, зменшують електроопір і мало збільшують теплопровідність. Для нікелевого порошку, як видно з порівняння наведених вище параметрів, не виконується характерний для металів закон Відемана-Франца, згідно якого $k_{\text{п}} \cdot \rho_{\text{п}} = T \cdot \text{const}$. Наприклад, після адсорбції електроліту $k_{\text{п}} \cdot \rho_{\text{п}} = 2,4 \cdot 10^{-6}$, а до адсорбції $k_{\text{п}} \cdot \rho_{\text{п}} = 2,7 \cdot 10^{-6}$. Все це в цілому подає надії на перспективність використання в термоелектричних пристроях модифікованих адсорбатами насипних порошків. Вони мають перевагу перед композиціями типу електропровідник-діелектрик, бо в них адсорбція забезпечує рівномірний розподіл адсорбата по поверхні електропровідних частинок і об'ємі зразка. Наслідком цього є збільшення термоелектричної добротності.

При адсорбції, як відомо, виділяється теплота. Тому виникає е.р.с. $E_{\text{п}}$ навіть тоді, коли різниця температур на термопарі $\Delta T = 0$. Адсорбція етанолу, наприклад, спричинює появу $E_{\text{п}} = -(1,7 \pm 0,8)$ мкВ (рис.6). В наших дослідках на термопарі $\Delta T = T_{\text{гор}} - T_{\text{хол}} = 42^\circ \text{C}$, тому адсорбційний приріст термо-е.р.с. $\Delta E = -3,6 \cdot 42 = -151$ мкВ (рис.1). Отже, $E_{\text{п}}/\Delta E \approx 1\%$. При адсорбції електроліту $\text{H}_2\text{O} + \text{HCl}$ $E_{\text{п}} = 5$ мкВ, а $\Delta E = 17 \cdot 42 = 714$ мкВ (рис 5). $E_{\text{п}}/\Delta E \approx 0,7\%$. Отже, $E_{\text{п}}$ знаходиться в межах похибок вимірювання термоелектричного адсорбційного ефекту.

Коли концентрація електроліту в порошок $C \approx 2\%$ об., то виникає також е.р.с. електрохімічної природи $E_{\text{е}} \approx +3$ мкВ. При $C = 6-10\%$ об. електроліт заповнює наполовину зазори між частинками порошку (пористість 0,18-0,23), $E_{\text{е}}/\Delta E \approx (2-3)\%$. Величина $E_{\text{п}}$ не є значущим параметром, бо температура, до якої нагрівався порошок внаслідок адсорбції, була розподілена по гаусіані з центром біля отвору, через який в порошок подавався адсорбат. Тому $E_{\text{п}}$ є сумою двох е.р.с., що мають протилежні знаки.

Висновки.

1. Збільшення термоелектричної добротності нікелевого порошку на 20% у порівнянні з добротністю фольги обумовлено переважно тунелюванням електронів крізь щілини між частинками порошку та обмеженням руху фононів.

2. При малій (10^{-4}) проникливості потенціального бар'єру шириною $\omega = 0,4$ нм термо-е.р.с. генерує мала (10^{-4}) доля електронів провідності. Тому термо-е.р.с. термопари з порошковою робочою віткою в 70-90 разів перевершує термо-е.р.с. термопари з робочою віткою із деформованої фольги.

3. Зміна термо-е.р.с. наростає при адсорбції молекул по ряду: уайт-спірит, етанол, електроліт $\text{H}_2\text{O} + \text{HCl}$ (50% об.).

4. Термоелектрична добротність порошку, що адсорбував електроліт, у 7 разів перевершує добротність порошку до адсорбції.

Список використаних джерел

1. Адамсон А. Физическая химия поверхности (Москва: Мир: 1979).
2. Дурягіна З.А. Фізика та хімія поверхні (Львів: Львівська політехніка: 2009).
3. Котырло Г.К., Лобунец Ю.Н. Расчет и конструирование термоэлектрических генераторов и тепловых насосов// Київ, Наукова думка.- 1980.- 327.
4. Борисенко В.Д., Копань В.С., Бессонов В.Б. Термоэлектрическое устройство для контроля неоднородности материалов. АС СССР №750357. Бюллетень изобретений.- 1980.- №27.
5. Копань В.С., Хуторянська Н.В., Копань Ю.В. Термоелектричний адсорбційний ефект в металевих та вуглецевих порошках // Вісник Київського університету, Серія: фіз.-мат. науки. - 2010. - №3 - С.265-268.
6. Булат Л.П., Драбкін І.А., Пивоваров Г.І. Про термоелектричні властивості матеріалів з нанокристалічною структурою// Термоелектрика, 2008.- №4.- С.27-33.
7. Копань В.С., Хуторянська Н.В., Копань Ю.В. Вплив магнітного поля і механічних напруг на термо-е.р.с. нікелю // Вісник Київського університету, Сер.: фіз.-мат. науки. - 2005. - №1. - С. 332-338.
8. Семко Л.С., Кручек Е.И., Горбик П.П., Шевляков Ю.А. Влияние магнитного поля на сенсорные характеристики композиционных материалов на основе нанокристаллического никеля// Химия, физика и технология поверхности. - 2008.- вып.14.- С.398-406.
9. Ильченко Л.Г., Лобанов В.В. Теоретический расчет структурного потенциала в сверхтонкой вакуумной щели между двумя твердыми телами// Физико-химия наноматериалов и супрамолекулярных структур.- 2007.- №2.- С.53-81.

Надійшла до редколегії 17.01.2011.

УДК 532.6; 538.9

Королович В.Ф.^{*}, асп.
Пур Хосро П., студ.
Мороз К.О., к.ф.-м.н.
Меленевський Д.О., к.ф.-м.н.
Держипольський А.Г.

Водні системи з вуглецевими нанотрубками: молекулярна структура і токсичність

В даній роботі представлено огляд літератури з молекулярної структури води, що оточує одностінну вуглецеву нанотрубку. Аналіз молекулярної структури здійснювався за функціями радіального розподілу атомів кисню та водню, орієнтацією молекул води і повною енергією взаємодії в системі вода – вуглецева нанотрубка.

Коротко оглянуто дослідження *in vivo* та *in vitro* токсичності вуглецевих нанотрубок.

Ключові слова: водні системи, вуглецеві нанотрубки, молекулярна структура, токсичність.

^{*}E-mail: korolovich@gmail.com

Вступ

З терміном “нанотехнології” пов’язують теперішній розвиток і майбутнє науки та високих технологій, що впливатимуть на тривалість, комфортність і захищеність людського життя за різних умов. Так до 2012 р. 80% лікарських препаратів у США будуть вироблятися на основі нанотехнологій [1-3], а вже зараз впроваджуються розробки: багат шарових нанопокриттів для імплантатів прискорюючих проростання судин та зменшуючих вірогідність інфікування організму [4]; фільтрувальних наноматеріалів [5]; біосенсорів [6]; рентгенівських установок, принцип роботи яких заснований на застосуванні вуглецевих нанотрубок (CNT) [7].

Спільною рисою більшості нанотехнологій пов’язаних з вуглецевими нанотрубками (Рис.1) виникнення на етапах впровадження чи експлуатації - водних систем з вуглецевими нанотрубками (сучасна класифікація рідинних систем представлена академіком НАН України Л.А. Булавіним і професором І.І. Адаменко в роботі [8]). Дослідження термодинамічних

V.F. Korolovych^{*}, PhD
P. Pour Khosrow, student,
K.O. Moroz, PhD.
D.O. Melenevskyi, PhD.
A.G. Derzhypolskyi

Water system with carbon nanotubes: Molecular structure and toxicity

In this review article, we present summary literary data about molecular structure of water surrounding a carbon nanotubes.

Analysis of the molecular structure was carried out by radial distribution functions of oxygen and hydrogen. The orientation of water molecules and the total energy of interaction in water - carbon nanotubes.

Presented is the short study by *in vivo* and *in vitro* toxicity of carbon nanotubes.

Key Words: water system, carbon nanotubes, molecular structure, toxicity.

властивостей водних систем з CNT виявило ряд особливостей: гістерезис баричної залежності густини [9], зміщення максимуму ізотермічного модуля пружності [10] водної системи в низькотемпературну область у порівнянні з водою [11] та рідинною системою вода – фулерен C_{60} [12]. Особливості термодинамічних властивостей зумовлені поданням особливостей молекулярної структури вуглецевих нанотрубок і води.

Не дивлячись на зростаючу з кожним роком кількість публікацій, присвячених фізичним властивостям водних систем з вуглецевими нанотрубками, на сьогодні актуальне питання їх молекулярної структури.

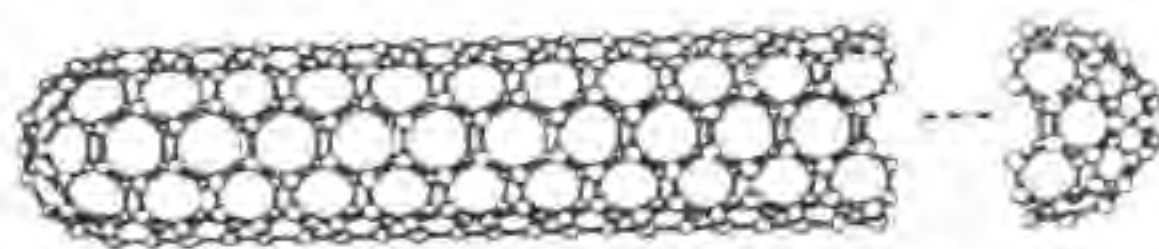


Рис.1. Модель одностінної вуглецевої нанотрубки

Представлено: акад. НАН України,
д.ф.-м.н., проф. Л.А. Булавін

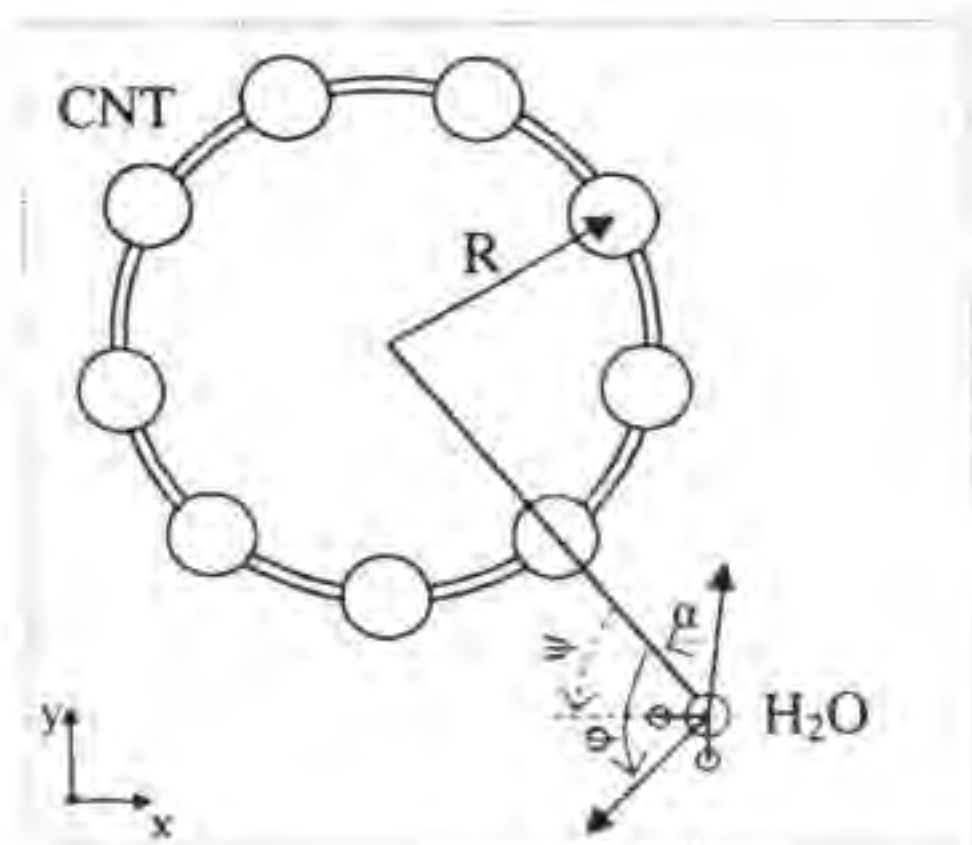


Рис.2. Схематичне зображення системи
вуглецева нанотрубка - молекула води
(за основою взятий рис. 2 [13])

зарядами на Н та О атомах води:

$$U = \frac{1}{3} \frac{q}{4\pi\epsilon_0} \sum_{\alpha, \beta} \Theta_{\alpha, \beta} \frac{3r_{\alpha}r_{\beta} - r^2\delta_{\alpha\beta}}{r^5} \quad (2)$$

де α і β пробігають через всі декартові координати: x, y, z , а r – відстань між зарядом О чи Н атома і квадруполем атома С; $\delta_{\alpha\beta}$ – δ функція; $\Theta_{\alpha\beta}$ – тензор квадрупольного моменту [31].

Структурні властивості шару води, що оточує вуглецеву нанотрубку аналізувалися (у порівнянні з аналогічними даними для вільної води, плоского прошарку води і шару води в контакті з водяною парою) по:

а) Функції радіального розподілу густини атомів кисню та водню

Функція радіального розподілу густини атомів – це гарно відома та найбільш застосовна структурна характеристика рідин та рідинних систем. Як відомо, на функціях радіального розподілу густини атомів кисню $g_{O-O}(r)$ і водню $g_{H-H}(r)$ чистої води спостерігається декілька максимумів, що відповідають радіусам координаційних сфер. Перший максимум відповідає першій координаційній сфері, яку ототожнюють з найближчою відстанню між атомами одного типу. Вона є найбільш чутливою до внесення домішок у воду і зміни термодинамічних параметрів (на Рис.3. – продемонстровано вплив температури на функцію радіального розподілу густини атомів кисню у воді за результатами МД моделювання

Табл.1

Параметри моделі взаємодії вода – вуглецева нанотрубка^[13]

$K_{wr} = 4637 \text{ kJ mol}^{-1} \text{ \AA}^{-2}$	$r_w = 1.0 \text{ \AA}$
$K_{w\theta} = 383 \text{ kJ mol}^{-1} \text{ rad}^2$	$\theta_w = 109,47^\circ$
$\epsilon_{ow} = 0,6502 \text{ kJ mol}^{-1}$	$\sigma_m = 3.166 \text{ \AA}$
$q_o = -0,82 e$	$q_H = -0,41 e$
$\epsilon_{r(r)} = 0,3126 \text{ kJ mol}^{-1}$	$\sigma_{r(r)} = 3.19 \text{ \AA}$
$\Theta_{x',y'} = -3,03 \cdot 10^{-40} \text{ C}^2$	

проф. І.І. Адаменко та асп. В.Ф. Королович). Зазначимо, вперше експериментальне розділення 3-ох радіальних функцій води: $g_{O-O}(r)$, $g_{O-H}(r)$, $g_{H-H}(r)$ було здійснене на основі 3-ох незалежних експериментів по розсіянню: електронів [32-33], нейтронів та рентгенівських променів [34-35], що дало можливість виявити нові особливості молекулярної структури, відсутні на теоретичних кривих.

Аналіз профілів радіального розподілу атомів О і Н води, за даними [13], показав, що додавання вуглецевих нанотрубок у воду призводить до зміщення 1-ї координаційної сфери атомів водню ближче до поверхні CNT. Атоми Н розташовуються на відстані від поверхні CNT $r^* = r - R < 2,1 \text{ \AA}$.

Врахування квадрупольного моменту, що полягає в притяганні додатнього заряду атома Н до поверхні CNT і відштовхуванні від'ємного заряду атома О, виявив незначний його вплив на

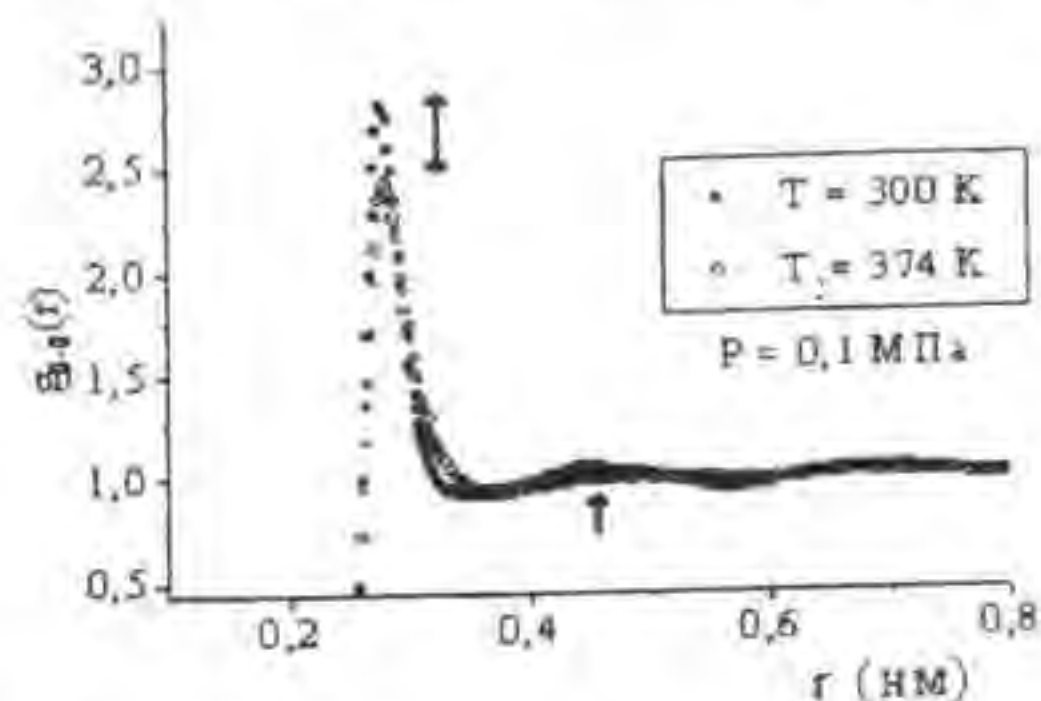


Рис.3. Функція радіального розподілу густини атомів кисню у воді при тиску 0,1 МПа і температурах 300 К і 374 К

густиною радіального розподілу атомів води (цей факт узгоджується з дослідженнями адсорбції води на графіті [36]).

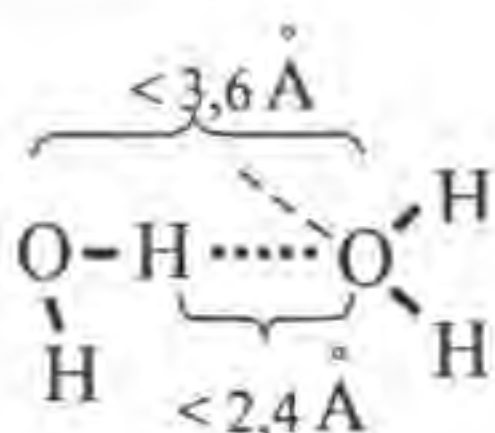


Рис.4. Геометричний критерій утворення водневого зв'язку між молекулами води

Отримані результати гарно узгоджуються з дослідженнями води в контакт з плоскою стінкою [37].

б) Орієнтація молекул води в системі з вуглецевими нанотрубками

Уявлення про орієнтацію молекул води отримані з дослідження орієнтації дипольного моменту води D_{H_2O} і орієнтації водневих зв'язків, НОН площини та середньої кількості водневих зв'язків, що припадають на одну молекулу води.

Критерієм утворення водневого зв'язку між двома молекулами H_2O є геометричний критерій [38] (слід зазначити, що детальний огляд критеріїв водневого зв'язку представлено в роботі [39]), а саме (Рис.4):

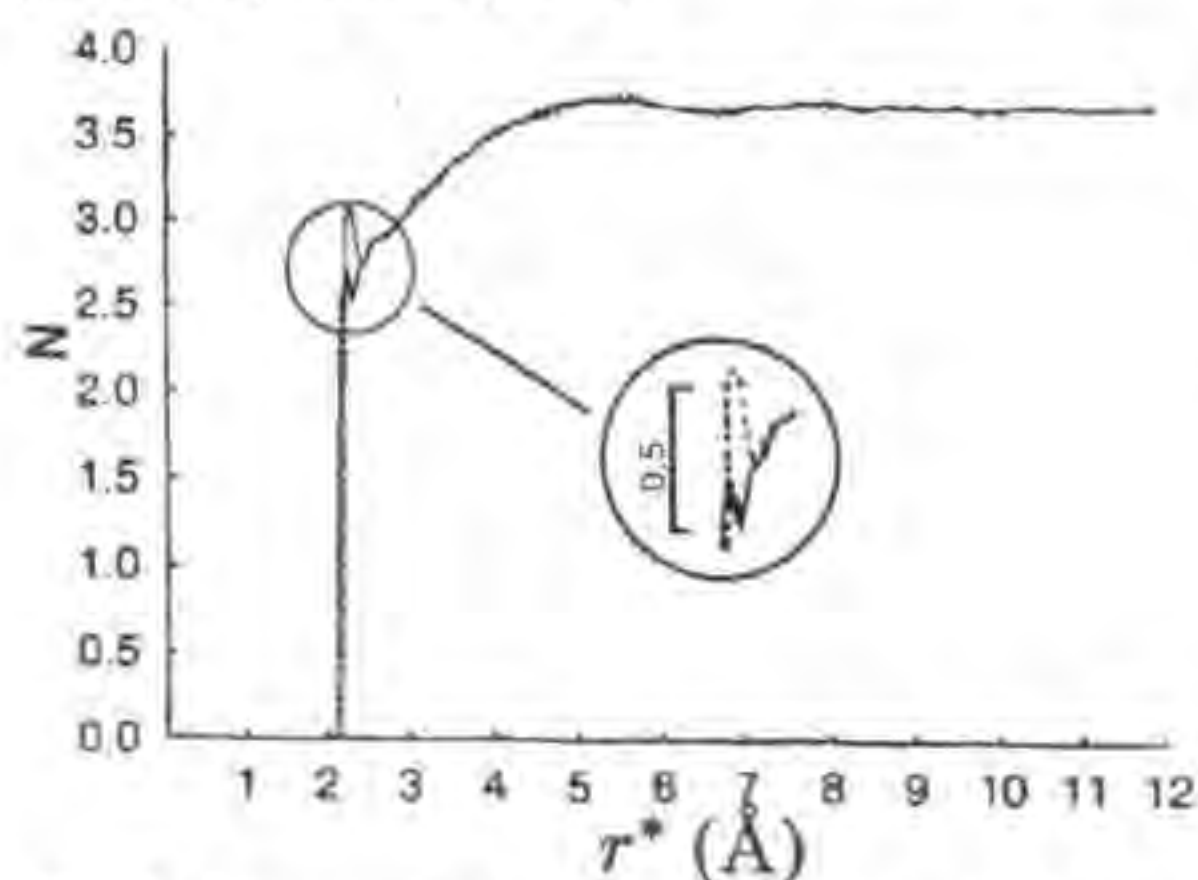


Рис.5. Залежність середньої кількості водневих зв'язків N , що припадає на одну молекулу води від відстані до поверхні вуглецевої нанотрубки [13]

- Відстань між атомами кисню О двох молекул води $< 3,6 \text{ Å}$
- Відстань між атомом кисню О молекули акцептора і атомом водню Н – молекули донора $< 2,4 \text{ Å}$

- Кут між О – О напрямком і зв'язком О – Н донора $< 30^\circ$

Дослідження [13] D_{H_2O} вказує, що його орієнтація змінюється в залежності від відстані до поверхні CNT. На відстанях $r^* = 2,85 \text{ Å} - 3,37 \text{ Å}$ кут φ , що визначає орієнтацію D_{H_2O} (Рис.2) рівний в більшості $94^\circ - 95^\circ$, тобто D_{H_2O} практично паралельний площині CNT. При $r^* = 4,90 \text{ Å}$ дипольний момент повертається під кутом 75° . Отже з збільшенням відстані від поверхні CNT положення дипольного моменту води стає все більш характерним для об'ємної води, остаточно набуваючи властивостей об'ємної води при $r^* \geq 9,00 \text{ Å}$. Ці результати узгоджуються з дослідженням поведінки дипольного моменту води між гідрофобними поверхнями [40] та [37].

Подібно до орієнтації D_{H_2O} , орієнтація водневих зв'язків та НОН площини молекули H_2O також залежить від відстані до поверхні CNT [13]. На відстані $r^* = 2,85 \text{ Å}$ кут ψ , що визначає орієнтацію водневих зв'язків рівний $92^\circ - 93^\circ$ (тобто водневі зв'язки практично паралельні поверхні CNT).

При r^* рівному: $3,37 \text{ Å}$; $4,34 \text{ Å}$; $4,79 \text{ Å}$; $5,22 \text{ Å}$ величина кута ψ відповідає значенням: 100° , 115° , 125° , 135° . Це узгоджується з результатами дослідження орієнтації площини НОН молекули води, яка з віддаленням від поверхні CNT обертається в напрямку перпендикулярному до поверхні CNT.

Дослідження структури водневого зв'язку показало, що середня кількість водневих зв'язків N , яка припадає на одну молекулу води зменшується від 3,75 в об'ємі до 2,89 біля поверхні CNT (Рис.5).

Максимум на Рис.5 при $r^* = 2,85 \text{ Å}$ зумовлений водою в об'ємі.

Аналіз отриманих даних також виявив малий вплив квадрупольного моменту на орієнтацію молекули води і підтвердив ефективність прийняття за радіус обрізання далеко діючого потенціалу $9,50 \text{ Å}$ (середньо квадратичне відхилення при аналогічних дослідженнях з радіусом обрізання $12,66 \text{ Å}$ між кривими радіального розподілу густини і

орієнтації дипольного моменту молекул води є менше 2% та 5% відповідно)

в) Повна енергія взаємодії в системі вода – вуглецева нанотрубка

В рідинній системі вода – вуглецева нанотрубка повна енергія цієї системи відрізняється від повної енергії води на величину ΔE_{Q-W} [13]:

$$\Delta E_{Q-W} = \Delta E_{CC} + E_{CW} + E_{WW}$$

де доданок ΔE_{CC} – це зміна енергії взаємодії між атомами вуглецю (C), зумовлена присутністю молекул води навколо CNT; ΔE_{CW} – енергія взаємодії атомів C нанотрубки з молекулами води, що оточують CNT; ΔE_{WW} – частково представляє енергію необхідну для створення порожнини у воді для CNT. Вона і обумовлена взаємодією між молекулами води та пов'язана з поверхневою енергією.

Для розглядуваної системи вода – CNT [13] внески ΔE_{CC} , ΔE_{CW} , ΔE_{WW} на одиницю довжини CNT дорівнюють:

$$\Delta E_{CC} = 0,59 \text{ кДж / мол}$$

$$\Delta E_{CW} = -17,33 \text{ кДж / мол}$$

$$\Delta E_{WW} = 45,40 \text{ кДж / мол}$$

Отже, найбільший внесок у ΔE_{Q-W} вносить доданок ΔE_{WW} , який відображає характеристики поверхні і поверхневої енергії пов'язаної з утворенням порожнини у воді. Його аналіз та порівняння поверхневої енергії, що відповідає плоскій порожнині 126,9 дин/см, циліндричній порожнині 124 дин/см (з радіусом 9,46 Å) та чистої води 119,66 дин/см [41-42] вказує, що поверхнева енергія суттєво не залежить від геометрії поверхні (дослідження проведені в роботі [43] показали, що лише при значно малих радіусах викривлення поверхні поверхнева енергія суттєво змінюється).

Токсичність вуглецевих нанотрубок

Застосування наноматеріалів у медицині і біології гостро поставило питання про їх вплив на організм людини і навколишнє середовище. Сформувався новий розділ токсикології, що досліджує токсичність і біосумісність наноматеріалів – нанотоксикологія.

Нанотоксикологія передбачає дослідження як на тваринах (*in vivo*), так і на культурах клітин (*in vitro*).

Починаючи з 2000 р. досліджується токсичність вуглецевих нанотрубок (CNT) та їх водорозчинних комплексів. Отримані результати взаємозаперечують один одного [44], що свідчить про необхідність подальшого дослідження токсичності CNT.

Данні одних досліджень вказують, що одностінні CNT проявляють більшу токсичність ніж багатостінні CNT при взаємодії з альвеолярними макрофагами (*in vitro*) [45]. А контакт одностінних CNT з клітинами людської шкіри викликає морфологічні зміни в них [46].

Данні інших досліджень свідчать про відсутність токсичності у CNT, а прояви токсичності пояснюють недостатньою очисткою CNT. У 40 добровольців та у 4 кроликів досліджували вплив CNT на шкіру та слизову оболонку очей [47]. Отримані результати свідчать про відсутність токсичності CNT.

Легенева токсичність вуглецевих нанотрубок визначається їх геометричними параметрами та часом впливу. *In vivo* дослідження часу впливу CNT на дихальні шляхи гвінейських свиней виявило: через 4 тижні [48] – відсутність запальних або функціональних змін у них; через 90 діб [49] – формування мультифокальних гранул навколо матеріалу і запальна реакція навколо респіраторних бронхіол. *In vitro* дослідження токсичності впливу CNT різного розміру присвячена стаття.

У цілому, токсичність вуглецевих нанотрубок продовжує викликати інтерес дослідників різних наукових областей, але на даний час питання токсичності вуглецевих нанотрубок потребує подальшого дослідження.

Висновки

Аналізуючи залежність структурних характеристик води, що оточує вуглецеву нанотрубку з відстанню від поверхні вуглецевої нанотрубки можна припустити, що додавання у воду вуглецевих нанотрубок призводить до руйнування сітки водневих зв'язків води навколо вуглецевої нанотрубки.

Основний вклад в зміну повної енергії води при додаванні вуглецевих нанотрубок вносить енергія необхідна для утворення порожнини у воді.

Список використаної літератури

1. Agoramoorthy G., Chakraborty C. «Re: introduction to nanotechnology: potential applications in physical medicine and rehabilitation». Am. J. Phys Med Rehabil. 2007. 86(12). p.1031-1032
2. Jain K.K. «Applications of nanobiotechnology in clinical Diagnostics». Clin. Chem., 2007. 53(11). p.2002-2009.
3. Gordon A.T., Lutz G.E., Boninger M.L., Cooper R.A. «Introduction to nanotechnology: potential applications in physical medicine and rehabilitation». Am. J. Phys. Med. Rehabil. 2007. 86(3). p. 225-241.
4. www.nanowerk.com «Biological nanocoating technology improves surgical implants». 15.12.2008
5. Лернер М. И. «Электровзрывные нанопрошки неорганических материалов: технология производства и области применения» Автореферат на соискание ученой степени доктора технических наук. РФ. Томск 2007
6. Balavoine F.; Schultz P.; Mallouh V.; Ebbesen T.W.; Mioskowski C. Angew. Chem. 1999, 38 (13/14), 1912-1915.
7. G. Cao et al. «A dynamic micro-CT scanner based on a carbon nanotube field emission x-ray source» Phys. Med. Biol. 2009. (54) 2323-2340
8. І.І.Адаменко, Л.А. Булавін «Фізика рідин та рідинних систем». К.-2006. ст. 659.
9. І.І. Адаменко, et al., "Spectral and thermodynamic properties of water system with SWCNTs", ICHMS-2009, p.454.
10. В.Ф. Королович, І.І. Адаменко та інш. "Особливості структури і міжмолекулярної взаємодії у водних системах з функціоналізованими вуглецевими нанотрубками" FMMN-2009, ст.
11. І.І. Адаменко, S.O. Zelinsky, V.F. Korolovych, Ukr. J. Phys. 52, 855-859 (2009).
12. І.І. Адаменко et al., "Equation of C_{60} Fullerene Aqueous Solution", Int. J. Thermophysics, V. 26, (2005).
13. J.H. Walther, R. Jaffe, T.Halicioglu and P.Koumoutsakos «Carbon nanotubes in Water: structural Characteristics and Energetics» J. Phys. Chem. B. 2001, 105, 9980-9987
14. Елецкий А.В. "Углеродные нанотрубки" УФН 1997, 167, 945
15. Wang Chung-Lei, Li Zhao-Xia, and et. "High density gas state at water /graphite interface studied by molecular dynamics simulation" Chine Phys B., 2008, 17(07). p. 2646-09
16. M.C. Gordillo, G. Nagy, J. Marti. " Structure of water nanoconfined between hydrophobic surfaces" The J. Chemical Physics. 2005. 123. 054707.
17. Teleman O., Jönsoon B., Engström S. / Mol.Phys.1987, 60 (1),193-203.
18. Саркисов Г.Н. "Структурные модели воды" УФН 2006. 176 (8). ст. 833-844.
19. Andrea T.A., Swope W. C., Andersen H. C. / J. Chem. Phys. 1984, 79(9), 4576-4584.
20. D. Frenkel, B. Smit "Understanding Molecular simulation" Academic press 2002, p.638
21. Berendsen H.J.C., Grigera J.R., Straatsma T.P. / J. Phys. Chem. 1987 (91). 6269
22. Wallqvist A., Teleman O. / Mol. Phys. 1991, 74(3), 515-533.
23. Mizan, T. I.; Savage, P. E.; Ziff, R. M. J. Phys. Chem. 1994, 98, 13067-13076
24. Mizan, T. I.; Savage, P. E.; Ziff, R. M. J. Comput. Chem. 1996, 17(15), 1757-1770.
25. Mizan, T. I.; Savage, P. E.; Ziff, R. M. J. Phys. Chem. 1996, 100, 403-408
26. Mizan, T. I.; Savage, P. E.; Ziff, R. M. J. Supercrit. Fluids 1997, 10, 119-125
27. Guo, Y.; Karasawa, N.; Goddard, W. A. III. Nature 1991, 351, 464-467
28. Tuzun, R. E.; Noid, D. W.; Sumpter, B. G.; Merkle, R. C. Nanotech. 1996, 7 (3), 241-248.
29. Rappe, A. K.; Casewit, C. J.; Colwell, K. S.; Goddard W. A., III; Skiff, W. M. J. Am. Chem. Soc. 1992, 114, 10024-10035.
30. Bojan, M. J.; Steele, W. A. Langmuir 1987, 3 (6), 1123-1127.
31. Hansen, F. Y.; Bruch, L. W. Phys. Rev. B 1995, 51 (4), 2515-2536
32. Kaatz U., Woerman D. Dielectric study of a binary aqueous mixtures with a lower critical point // J. Phys. Chem. – 1984. – 88, №2 – p.284 – 288.
33. Palikas G., Kalman E., Kovacs P. , Mol. Phys. – 1977. – 34, №2 – p.284 – 288
34. Narten A.H., Levy H.A. Liquid water: molecular correlation functions from X-ray diffraction// 1 bid. -56 - №11 – p.5681 – 5687.
35. Navasques G., Berry M.V. The statistical mechanics of wetting // J. Mol. Phys. – 1977. – 34, №3. – p.649 – 664
36. Bojan, M. J.; Steele, W. A. Langmuir 1987, 3 (6), 1123-1127.
37. Wallqvist, A., Chem. Phys. Lett. 1990, 165(5), 437-442
38. Gordillo, M. C.; Martí, J. Chem. Phys. Lett. 2000, 329, 341-345.
39. Н.Ф. Степанов. «Водородная связь: как ее понимать». Соровский обозревательный журнал. Том 7, №2, 2001
40. Lee, C. Y.; McCammon, J. A.; Rossky, P. J. J. Chem. Phys. 1984, 80 (9), 4448-4455.
41. Lide, D. R. Handbook of Chemistry and Physics, 78th ed.; CRC Press: New York, 1998.
42. Kroschwitz, J. I., Howe-Grant, M., Eds. Encyclopedia of Chemical Technology, 4th ed.; New York, 1998.
43. Bogdan, A. J. Chem. Phys. 1997, 106 (5), 1921-1929.
44. І.С. Чекман, О.В. Швець, О.О. Нагорна, «Карбонові нанотрубки: методи отримання та перспективи застосування в медицині», Укр. Мед. Часопис, 3(65) – V/VI, (2008)
45. Monteiro-Rivieri N.A. et al., Carbon V.44, p.1070- 1078, (2006)
46. Сивигров Д.Е. и др. "Изучение морфо-физиолог. параметров клеток in vitro в присутствии ОУНТ". Rusnanotech-2008, с. 316-317.
47. Huczko A., Lange A. Carbon nanotubes: experimental evidence for a null risk of skin irritation and allergy. Fullerene Sci. Tech, 9(2), p. 247, (2001)
48. Huczko A., Lange A., Calko E. et. al. Physiological testing of carbon nanotubes: are they asbestos-like?, Fullerene Sci. Tech, 9(2), p. 251-254, (2001)
49. Huczko A., Lange A., Bystrzejewski M. et al. Pulmonary toxicity of 1-D nanocarbon materials. Fuller Nanotub. Car. 13; 141-145, (2005).

Надійшла до редколегії 01.11.2010

Підп. до друку 22.02.2011.
Формат 70x100¹/₁₆. Папір офс.
Ум. друк. арк. 23,54. Наклад 350.
Зам. 47.

ДП «Інформаційно-аналітичне агентство»
вул. Еспланадна 4-6, м. Київ, 01601, Україна
тел/факс 287-03-79, 289-77-62
E-mail: iaa@dstati.kiev.ua

