

Наведено результати досліджень з аналізу, оцінки, керування й оптимізації динамічних систем, проблем еколого-економічного аналізу та чисельних методів моделювання процесів.

Для викладачів, наукових співробітників, аспірантів і студентів.

In this issue the results of researches in analysis, estimates, control and optimization of dynamical systems, problems of ecology-economic analysis and numeral methods of processes are presented.

For scientists, professors, aspirants and students.

<b>ВІДПОВІДАЛЬНИЙ РЕДАКТОР</b>	О. К. Закусило, д-р фіз.-мат. наук, проф., акад. НАПН України
<b>ЗАСТ. ВІДПОВ. РЕДАКТОРА</b>	А. В. Анісімов, д-р фіз.-мат. наук, проф., чл.-кор. НАН України
<b>ВІДПОВІДАЛЬНИЙ СЕКРЕТАР</b>	Д. Я. Хусаїнов, д-р фіз.-мат. наук, проф.
<b>РЕДАКЦІЙНА КОЛЕГІЯ</b>	В. В. Акименко, д-р техн. наук, проф., Д. Б. Буй, д-р фіз.-мат. наук, проф.; Ф. Г. Гаращенко, д-р техн. наук, проф.; В. А. Заславський, д-р техн. наук, проф.; Є. О. Лебедев, д-р фіз.-мат. наук, проф.; С. І. Ляшко, д-р фіз.-мат. наук, проф., чл.-кор. НАН України; О. Г. Наконечний, д-р фіз.-мат. наук, проф.; М. С. Нікітченко, д-р фіз.-мат. наук, проф.; Д. А. Номіровський, д-р фіз.-мат. наук, проф.; О. І. Провотар, д-р фіз.-мат. наук, проф.; В. Н. Редько, д-р фіз.-мат. наук, проф., акад. НАН України; В. О. Яценко, д-р техн. наук, проф.
<b>Адреса редколегії</b>	03127, Київ-127, просп. акад. Глушкова, 6, факультет кібернетики ☎ (38044) 259 01 49
<b>Затверджено</b>	Вченою радою факультету кібернетики 16.11.2015 (протокол № 4)
<b>Атестовано</b>	Постанова Президії ВАК України № 1-05/1 від 26.01.11
<b>Зареєстровано</b>	Міністерством юстиції України. Свідоцтво про державну реєстрацію КВ № 16271-4743Р від 31.12.09
<b>Засновник та видавець</b>	Київський національний університет імені Тараса Шевченка, ВПЦ "Київський університет". Свідоцтво внесено до державного реєстру ДК № 1103 ВІД 31.10.02
<b>Адреса видавця</b>	01601, Київ-601, 6-р Т. Шевченка, 14, кімн. 43 ☎ (38044) 239 31 72, 239 32 22; факс 239 31 28

---

## ЗМІСТ

---

<b>Апанасенко Д. В.</b> Про методи кластеризації множин складених нечітких чисел .....	5
<b>Грищенко О. Ю., Загородня Г. О., Оноцький В. В.</b> Двокроковий симетризований алгоритм для чисельного моделювання процесу розповсюдження випромінювання в деформуючому середовищі .....	8
<b>Івохін Є. В.</b> Використання простих чисел для інтервального визначення нечітких трикутних цілих чисел.....	15
<b>Кулян В. Р., Юнькова О. О.</b> Методи оптимального керування в задачах диверсифікації портфеля інвестицій .....	18
<b>Махно М. Ф.</b> Про гібридну модель динаміки процесу обробки сукупності завдань .....	22
<b>Нікітін А. В.</b> Умови стійкості стохастичних диференціальних рівнянь ІТО-Скоророда зі сталими коефіцієнтами у гільбертових просторах .....	27
<b>Нікітченко М. С., Шкільняк С. С.</b> Числення секвенційного типу для перевірки виконаності в логіках квазіарних предикатів.....	30
<b>Петрович В. М., Требіна Н. М., Двірничук К. В.</b> Про один підхід до розв'язання задачі математичного моделювання одновимірної динамічної системи з неповно визначеним крайовим станом .....	38
<b>Скобелєв В. Г.</b> Проблеми аналізу та синтезу великомасштабних мереж (огляд) .....	42
<b>Скуратовський Р. В.</b> Модернізований алгоритм Поліга-Хелмана, Шенкса.....	56
<b>Трохимчук Р. М., Козлов К. Є.</b> Використання модульних нейронних мереж для аналізу та прогнозування соціально-економічних та екологічних показників регіону.....	63

---

## СОДЕРЖАНИЕ

---

<b>Апанасенко Д. В.</b> О методах кластеризации множеств составных нечетких чисел .....	5
<b>Грищенко А. Е., Загородная А. А., Оноцкий В. В.</b> Двухшаговый симметризованный алгоритм для численного моделирования процесса распространения излучения в искажающей среде .....	8
<b>Ивохин Е. В.</b> Использование простых чисел для интервального определения нечетких треугольных целых чисел.....	15
<b>Кулян В. Р., Юнькова Е. А.</b> Методы оптимального управления в задачах диверсификации портфеля инвестиций .....	18
<b>Махно М. Ф.</b> О гибридной модели динамики процесса обработки совокупности заданий .....	22
<b>Никитин А. В.</b> Условия устойчивости стохастических дифференциальных уравнений ИТО-Скоророда с постоянными коэффициентами в гильбертовых пространствах .....	27
<b>Никитченко Н. С., Шкильняк С. С.</b> Исчисления секвенциального типа для проверки выполнимости в логиках квазиарных предикатов .....	30
<b>Петрович В. Н., Требина Н. Н., Двирничук К. В.</b> Об одном подходе решения задачи математического моделирования одномерной динамической системы с неполно определенным краевым граничным состоянием.....	38
<b>Скобелев В. Г.</b> Проблемы анализа и синтеза крупномасштабных сетей (обзор).....	42
<b>Скуратовский Р. В.</b> Модернизированный алгоритм Полига-Хэлмана, Шенкса.....	56
<b>Трохимчук Р. Н., Козлов К. Е.</b> Использование модульной нейронной сети для анализа и прогнозирования социально-экономических и экологических показателей региона.....	63

---

## CONTENTS

---

<b>Apanasenko D. V.</b> About clusterization methods for datasets of composite fuzzy numbers.....	5
<b>Gryshchenko O. Yu., Zagorodnia G. O., Onotskyi V. V.</b> Symmetrized two-step difference algorithm for the numerical modeling of the process of radiation propagation in deforming environment .....	8
<b>Ivokhin E. V.</b> A simple numbers using for interval determination of fuzzy triangle integers .....	15
<b>Kulian V. R., Iunkova O. O.</b> Methods of optimal control problem in the diversification of the portfolio .....	18
<b>Makhno M. F.</b> About hybrid model of dynamics of the jobs collection processing .....	22
<b>Nikitin A. V.</b> Stability conditions ITO-Skhorohod stochastic differential equations with constant coefficient in Hilbert space .....	27
<b>Nikitchenko M. S., Shkilniak S. S.</b> Sequent type calculi for checking satisfiability in logics of quasi-ary predicates .....	30
<b>Petrovich V. N., Trebina N. M., Dvirnichuk K. V.</b> About one approach to problem solving of one decision of task of the mathematical programming of unidimensional dynamic system with incomplete certain border state .....	38
<b>Skobelev V. G.</b> Problems of analysis and synthesis of large-scale networks (survei) .....	42
<b>Skuratovskii R. V.</b> Modern algorithm of Pohlig-Hellman and Shanks.....	56
<b>Trokhymchuk R. M., Kozlov K. E.</b> Application of modular neural networks for analysis and prediction of socio-economical and ecological indicators of the region .....	63

**ПРО МЕТОДИ КЛАСТЕРИЗАЦІЇ МНОЖИН  
СКЛАДЕНИХ НЕЧІТКИХ ЧИСЕЛ**

*В статті розглянуто та розв'язано задачі групування елементів нечіткої множини, яка задається сукупністю складених нечітких чисел трикутного вигляду. Запропоновано та проаналізовано декілька методів кластеризації. Проведено тестування роботи алгоритмів на конкретному прикладі.*

**Ключові слова:** кластеризація, невизначеність, нечітке трикутне число, методи групування даних.

Традиційна задача групування даних у відносно однорідні групи, що виникає в самих різних сферах науки, економіки, соціуму та ін., включає в себе проблему виділення кластерів та задачу знаходження еталонних вузлів. В реалізації методів кластеризації важливу роль грають показники належності даних до групи, способи обчислення відстаней між елементами, вагові коефіцієнти, що залежать від конкретної предметної області та спостережень в рамках цієї області. Методи мають різну обчислювальну складність та різний ступінь ефективності при вирішенні конкретних прикладних задач.

Для проведення процесу кластеризації існує багато підходів, більшість з яких є евристичними методами, що базуються на певних алгоритмах дій дослідника і не вимагають складних статистичних розрахунків. Однак, їх використання у випадку обробки інформації нечіткого характеру суттєво ускладнюється або взагалі стає неможливим через специфічне представлення нечіткості.

Розробка конструктивного алгоритму для кластеризації нечітких даних, представлених сукупністю нечітких чисел, включає в себе формалізацію способів пошуку кластерного центру множини даних та реалізацію процедури групування нечітких чисел в межах заданої кількості кластерів.

У літературі, присвяченій проблемам кластеризації [1–3], найчастіше розглядаються три алгоритми: C-means, пікового групування та різницевого групування. Ці методи представляють собою кардинально різні підходи до вирішення проблеми кластеризації. Як правило, перший з них потребує від дослідника точного знання кількості кластерів, на базі чого будується алгоритм. Другий метод вимагає багато часу для виконання, бо має багато нетривіальних умов для перевірки щодо належності елемента до кластеру та побудови самого кластеру. Третій – найбільш оптимальний, але для його застосування необхідне детальне знання специфічних залежностей в системі для встановлення значень важливих параметрів, без яких метод не дає бажаного результату.

Задачу кластеризації нечітких даних, представлених сукупністю нечітких множин, за допомогою алгоритму нечіткого групування C-means було розглянуто в роботі [4]. Проведемо дослідження вищезгаданих методів кластеризації для випадку нечітких множин спеціального вигляду.

**Означення 1.** [5] Нечіткою множиною  $\tilde{A}$  універсальної множини  $X$ , називається сукупність пар  $\tilde{A} = \{(\mu_{\tilde{A}}(x), x)\}$ , де  $\mu_{\tilde{A}} : X \rightarrow [0, 1]$  – відображення множини  $X$  в одиничний відрізок  $[0, 1]$  і називається функцією належності нечіткої множини  $A$ .

За умов подання інформації у вигляді нечітких множин в якості універсальної множини  $X$  традиційно розглядається довільна підмножина скінченновимірного простору  $R^N : X \subseteq R^N, x = (x_1, \dots, x_N) \in R^N$ . У випадку  $X \subseteq R^1$  нечітка множина  $\tilde{A}$  містить сукупність пар, що складаються з двох скалярних значень  $x \in R^1$  та  $\mu_{\tilde{A}}(x)$ .

Нехай задано сукупність нечітких множин  $\tilde{A}_1, \dots, \tilde{A}_m$ , що визначені у відповідних універсальних множинах  $X_1, \dots, X_m$ , де  $X_i \subseteq R^1, i = \overline{1, m}$ . Тоді можна говорити, що задано множину нечітких чисел. Будемо вважати, що кожне з них має трикутний вигляд, тобто  $\tilde{A}_i = \{(a_i, b_i, c_i)\}$ ,  $a_i \leq b_i \leq c_i, i = \overline{1, m}$ , з лінійними функціями належності  $\mu_{\tilde{A}_i}(x), i = \overline{1, m}$ , вигляду:

$$\mu_{\tilde{A}_i}(x) = \frac{x - a_i}{b_i - a_i}, x \in [a_i, b_i]; \quad \mu_{\tilde{A}_i}(x) = \frac{c_i - x}{c_i - b_i}, x \in [b_i, c_i]; \quad \mu_{\tilde{A}_i}(x) = 0, x \notin [a_i, c_i], i = \overline{1, m}.$$

Розглянемо універсальну множину  $X$  у вигляді декартового добутку  $X = \prod_{i=1}^m X_i$ . Сформуємо множину

$$\tilde{A}^m = \{(x^1, \mu_{\tilde{A}_1}(x^1)), (x^2, \mu_{\tilde{A}_2}(x^2)), \dots, (x^m, \mu_{\tilde{A}_m}(x^m))\}, \tag{1}$$

де  $x^i \in X_i, i = \overline{1, m}$ .

**Означення 2.** Множину  $\tilde{A}^m$  назвемо складеним нечітким числом в універсальній множині  $\prod_{i=1}^m X_i$ .

Скінченний набір складених нечітких чисел  $\tilde{A}^m$  у множині  $X = \prod_{i=1}^m X_i$  позначимо  $K(\tilde{A}^m), |K(\tilde{A}^m)| = k$ .

**Означення 3.** Множину

$$L_0^m = \{x^1 \in X_1, x^2 \in X_2, \dots, x^m \in X_m : \mu_{\tilde{A}_1}(x^1) > 0, \mu_{\tilde{A}_2}(x^2) > 0, \dots, \mu_{\tilde{A}_m}(x^m) > 0\}, \tag{2}$$

$$L_0^m \subseteq X_1 \times X_2 \times \dots \times X_m,$$

будемо називати носієм складеного нечіткого числа  $\tilde{A}^m$ .

Означення 4. Множину

$$L^m(\alpha) = \{x^1 \in X_1, x^2 \in X_2, \dots, x^m \in X_m : \mu_{\tilde{A}_1}(x^1) \geq \alpha, \mu_{\tilde{A}_2}(x^2) \geq \alpha, \dots, \mu_{\tilde{A}_m}(x^m) \geq \alpha\} \quad (3)$$

назвемо множиною рівня  $\alpha \in (0, 1]$  складеного нечіткого числа  $\tilde{A}^m$ .

Розглянемо два довільних складених нечітких числа

$$\begin{aligned} \tilde{U}^m &= \{(x^1, \mu_{\tilde{A}_1}^U(x^1)), (x^2, \mu_{\tilde{A}_2}^U(x^2)), \dots, (x^m, \mu_{\tilde{A}_m}^U(x^m))\}, \\ \tilde{V}^m &= \{(y^1, \mu_{\tilde{A}_1}^V(y^1)), (y^2, \mu_{\tilde{A}_2}^V(y^2)), \dots, (y^m, \mu_{\tilde{A}_m}^V(y^m))\}. \end{aligned}$$

Обчислимо величину  $\gamma = \min_{i=1, m} \min \{\mu_{\tilde{A}_i}^U(x^i), \mu_{\tilde{A}_i}^V(y^i)\}$ , яка є мінімальним значенням серед значень мір належності окремих елементів обох множин  $\tilde{U}^m, \tilde{V}^m$ . Це значення дозволяє побудувати дві множини рівня  $\gamma$  у вигляді звичайних множин  $L_{\tilde{U}}^m(\gamma), L_{\tilde{V}}^m(\gamma)$ , що визначають точки універсальної множини  $X = \prod_{i=1}^m X_i$ , між якими може бути обчислена евклідова відстань  $d = \|L_{\tilde{U}}^m(\gamma) - L_{\tilde{V}}^m(\gamma)\|$ .

На основі введеного поняття відстані між складеними нечіткими числами та способу їх порівняння можна сформулювати алгоритми нечіткого групування даних, поданих у вигляді сукупності складених нечітких чисел  $\{\tilde{A}^{m(j)}, j = \overline{1, p}\}$  з  $K(\tilde{A}^m)$ . Будемо вважати, що цю сукупність можна згрупувати в  $k$  кластерів.

На початку процесу задається величина  $\gamma \in (0, 1]$ , яка буде визначати рівень нечіткості даних, що розглядаються. Припустимо, що складені нечіткі числа  $\tilde{A}^{m(j)}, j = \overline{1, q}, q \leq p$ , мають непорожні множини рівня  $L^m(\gamma)$ . Будемо розглядати їх як звичайні вектори  $S(\tilde{A}^{m(j)}) = \{x^{1(j)}, x^{2(j)}, \dots, x^{m(j)}\}, j = \overline{1, q}$ .

**Алгоритм пікового групування.** У алгоритмі пікового групування, який був запропонований Р. Єгером та Д. Фільовим [6], розглянуто міру щільності розміщення векторів  $S(\tilde{A}^{m(j)}), j = \overline{1, q}$ , для чого генеруються так звані пікові функції. При застосуванні  $p$  вхідних векторів створюється сітка, що рівномірно накриває простір векторів  $S(\tilde{A}^{m(j)}), j = \overline{1, q}$ . Вузли цієї сітки розглядаються як потенційні нечіткі центри

$$\tilde{C}^{m(i)} = \{(x^{1(i)}, \mu_{\tilde{A}_1}(x^{1(i)})), (x^{2(i)}, \mu_{\tilde{A}_2}(x^{2(i)})), \dots, (x^{m(i)}, \mu_{\tilde{A}_m}(x^{m(i)}))\}, i = \overline{1, k},$$

і для кожного з них розраховується пікова функція  $M(S(\tilde{C}^{m(i)})), i = \overline{1, k}$ ,

$$M(S(\tilde{C}^{m(i)})) = \sum_{j=1}^p \exp\left(-\|S(\tilde{C}^{m(i)}), S(\tilde{A}^{m(j)})\|^{2b} / 2\sigma^2\right), \quad (4)$$

де  $\|S(\tilde{C}^{m(i)}), S(\tilde{A}^{m(j)})\|, i = \overline{1, k}, j = \overline{1, p}$  - це евклідові відстані між парами векторів, що визначають центри  $\tilde{C}^{m(i)}, i = \overline{1, k}$  та складені нечіткі числа  $\tilde{A}^{m(j)}, j = \overline{1, p}$ , відповідно, коефіцієнт  $\sigma$  - це константа, що індивідуально підбирається для кожної конкретної задачі, а  $b$  - показник ступеню узагальненої функції Гауса.

Величина функції  $M(S(\tilde{C}^{m(i)})), i = \overline{1, k}$ , розглядається як оцінка висоти пікової функції. Вона пропорційна кількості векторів  $S(\tilde{A}^{m(j)}), j = \overline{1, q}$ , що знаходяться в околі потенційного центра  $S(\tilde{C}^{m(i)}), i = \overline{1, k}$ . Мале значення  $M(S(\tilde{C}^{m(i)})), i = \overline{1, k}$ , свідчить про те, що центр  $S(\tilde{C}^{m(i)}), i = \overline{1, k}$ , розташований в області, в якій зосереджено невелику кількість векторів  $S(\tilde{A}^{m(j)}), j = \overline{1, q}$ . Слід звернути увагу на те, що коефіцієнт  $\sigma$  має незначний вплив на кінцеві пропорції між  $M(S(\tilde{C}^{m(i)})), i = \overline{1, k}$ , для різних значень  $S(\tilde{C}^{m(i)}), i = \overline{1, k}$ , тому підбір його величини не є критичним.

Після обрахунку значень  $M(S(\tilde{C}^{m(i)})), i = \overline{1, k}$ , для усіх потенціальних центрів відбираються перші  $k_1$  точки, що мають найбільше значення  $M(S(\tilde{C}_1^{m(i)})), i = \overline{1, k_1}$ . Для вибору наступних центрів необхідно, перш за все, виключити  $k_1$  центрів та вузли, що розташовані в безпосередній близькості від них. Це можна зробити шляхом перевизначення пікової функції за рахунок відсікання від неї значень функції Гауса з центрами в точках  $S(\tilde{C}_1^{m(l)}), l = \overline{1, k_1}$ . Якщо цю знов визначену функцію позначити як  $M_{new}(S(\tilde{C}^{m(l)})), i = \overline{1, k}$ , то

$$M_{new}(S(\tilde{C}^{m(i)})) = M(S(\tilde{C}^{m(i)})) - M(S(\tilde{C}_1^{m(l)})) \exp\left(-\|S(\tilde{C}_1^{m(l)}), S(\tilde{C}^{m(i)})\|^{2b} / 2\sigma^2\right). \quad (5)$$

Необхідно звернути увагу на те, що функція  $M_{new}(S(\tilde{C}^{m(i)})), i = \overline{1, k}$ , набуває нульового значення в точках  $S(\tilde{C}_1^{m(l)}), l = \overline{1, k_1}$ . Звідси зрозуміло, що послідовне відсікання центрів (з максимальним значенням пікової функції) дозволяє виявляти та усувати наступні центри.

Процес знаходження наступних центрів  $S(\tilde{C}_2^{m(l)}), l = \overline{k_1 + 1, k_2}, S(\tilde{C}_3^{m(l)}), l = \overline{k_2 + 1, k_3}, \dots$ , здійснюється послідовно на модифікованих значеннях функції  $M_{new}(S(\tilde{C}^{m(i)})), i = \overline{1, k}$ , що отримуються при видаленні близького оточення центру, виявленого на попередньому етапі. Він завершується в момент локалізації усіх центрів, що використовуються в моделі. Метод пікового групування ефективний, якщо розмірність вектора  $S(\tilde{A}^{m(j)}), j = \overline{1, q}$  невелика. В іншому випадку (за великої кількості компонент  $S(\tilde{A}^{m(j)}), j = \overline{1, q}$ ), кількість потенційних центрів зростає досить швидко, і процес розрахунку чергових пікових функцій стає довготривалим, а процедура малоефективною.

**Алгоритм різницевого групування** даних [6] – це модифікація алгоритму пікового групування, в якій усі навчальні вектори  $S(\tilde{A}^{m(j)})$ ,  $j = \overline{1, q}$ , розглядаються в якості  $k$  потенційних центрів,  $k = q$ . Пікова функція  $D(S(\tilde{A}^{m(i)}))$ ,  $i = \overline{1, k}$ , у цьому алгоритмі задається у вигляді

$$D(S(\tilde{A}^{m(i)})) = \sum_{j=1}^p \exp \left( -\|S(\tilde{A}^{m(i)}), S(\tilde{A}^{m(j)})\|^{2b} / \left( \frac{r_a}{2} \right)^2 \right), \quad i = \overline{1, k}. \quad (6)$$

Значення коефіцієнта  $r_a$  визначає сферу сусідства. На значення  $D(S(\tilde{A}^{m(i)}))$ ,  $i = \overline{1, k}$ , істотно впливають лише ті вектори  $S(\tilde{A}^{m(j)})$ ,  $j = \overline{1, q}$ , що розташовані в межах цієї сфери. При великій щільності точок в околі  $S(\tilde{A}^{m(i)})$ ,  $i = \overline{1, k}$ , (потенційного центра) значення функції  $D(S(\tilde{A}^{m(i)}))$ ,  $i = \overline{1, k}$ , велике. Навпаки, її мале значення свідчить про те, що в околі  $S(\tilde{A}^{m(i)})$ ,  $i = \overline{1, k}$ , незначна кількість даних. Така точка вважається "невдалими" кандидатом в центри. Після розрахунку значень пікової функції для кожної точки  $S(\tilde{A}^{m(j)})$ ,  $j = \overline{1, q}$ , відбирається вектор  $S(\tilde{A}^m)$ , для якого міра щільності  $D(S(\tilde{A}^m))$  виявилась найбільшою. Саме ця точка стає першим відібраним центром  $S(\tilde{C}_1^m)$ . Вибір наступного центра можливий після виключення попереднього центру і всіх точок, що лежать в його околі. Так само, як і в методі пікового групування, пікова функція перевизначається у вигляді

$$D_{new}(S(\tilde{A}^{m(i)})) = D(S(\tilde{A}^{m(i)})) - D(S(\tilde{C}_1^m)) \exp \left( -\|S(\tilde{A}^{m(i)}), S(\tilde{C}_1^m)\|^{2b} / \left( \frac{r_a}{2} \right)^2 \right), \quad i = \overline{1, k}. \quad (7)$$

При новому визначенні функції  $D(S(\tilde{A}^m))$  коефіцієнт  $r'_a$  задає нове значення константи, яка окреслює сферу сусідства чергового центра. Зазвичай дотримується умова  $r'_a \geq r_a$ . Пікова функція  $D_{new}(S(\tilde{A}^{m(i)}))$ ,  $i = \overline{1, k}$ , приймає нульове значення при  $S(\tilde{A}^{m(i)}) = S(\tilde{C}_1^m)$ ,  $i = \overline{1, k}$ , і близька до нуля в найближчому околі цієї точки.

Після модифікацій значень пікової функції знаходиться наступна точка  $S(\tilde{A}^m)$ , для якої величина  $D_{new}(S(\tilde{A}^m))$  максимальна. Ця точка стає наступним центром  $S(\tilde{C}_2^m)$ . Процес пошуку чергового центра поновлюється після видалення компонентів, відповідних вже отриманим точкам.

Кожен з методів має свої власні особливості та уточнення в залежності від умов застосування та його структурно-концептуальної реалізації. Головним чинником, звичайно, виступає показник конструктивності метода, що на практиці суттєво залежить від умов конкретної задачі. Результати практичного застосування методів кластеризації на заданій сітці двомірних складених нечітких чисел дозволив зробити ряд висновків.

Якщо у методі C-means необхідно обирати початкові коефіцієнти [4], що показують ступінь належності кластеру, виключно випадково, то в алгоритмі пікового групування головну роль грає покриття площини рівномірною сіткою з розмірами комірок, залежними від максимального та мінімального значень двомірних даних по-координатно. Вузли цієї сітки будуть виступати як претенденти в центри майбутніх кластерів і виявляється, що обрати розміри комірок цієї сітки також є задачею, залежною від конкретної ситуації. Дану задачу можна вирішувати за правилом Рунге, яке полягає в знаходженні такого розбиття області, при якому величина  $\Delta_{2n} \approx \Theta |I_{2n} - I_n|$ , (тут  $I_n$  – результат розрахунку, отриманого при використанні  $n$  вузлів по кожній координаті сітки), досягає своєї наперед заданої точності. В обчислювальних експериментах для знаходження результатів вважалось  $\Theta = 1/3$ . Точність правила Рунге задається користувачем в залежності від умов прикладної області.

Параметри  $b$  та  $\sigma$ , що використовуються у методі, також підбираються індивідуально для кожної задачі. При обчисленні експоненціального виразу для оцінки згущення точок навколо кандидата, виникає проблема нульового значення виразу для деяких претендентів (через обмеженість розміру чисел в машинній інтерпретації), тому перед тим, як підставляти отримані в перетвореннях значення в показник експоненти, їх бажано поділити на константу (в обчислювальних прикладах, що розглядалися було використано значення 1000). Крім цього, при реалізації виникає проблема врахування близькості вузла сітки – кандидата в центри кластера, що робить значення суми експоненціальних виразів дуже великим, навіть без урахування значення цієї функції для інших точок. Тому вважається за потрібне ввести обмеження на ступінь близькості можливого кандидата у центри кластера до інших точок. Як варіант, пропонується визначати мінімально допустиму відстань за правилом Рунге.

Алгоритм різницевого групування представляє собою спрощену, але більш ефективну модифікацію алгоритму пікового групування. Усі зауваження, зроблені для метода пікового групування, у цьому випадку несуттєві. Єдиним невіршеним питанням залишається значення коефіцієнта  $r_a$ , що визначає сферу сусідства. При практичному використанні це значення може також визначатися за правилом Рунге з заданою точністю. Головним же плюсом алгоритму є те, що кандидатами в центри кластерів виступають самі нечіткі вхідні дані.

Проведено тестування методів розв'язання задачі кластеризації на прикладі групування станів нечіткої системи, що функціонує у двовимірному просторі [7].

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Jamba M. Hierarchical cluster analysis and compliance/ M. Jamba. – Moscow: Finance and Statistics, 1988. – 345 p.
2. Durand B. Cluster analysis/ B. Durand, P. Odell. – Moscow: Statistics, 1977. – 128 p.
3. Viattchenin D. A. Fuzzy methods of automatic classification/ D. A. Viattchenin. – Minsk: Tehnoprint, 2004. – 219 c.
4. Івохін Є. В. Один метод кластеризації складених нечітких множин / Є. В. Івохін, К. О. Косинський // Журнал обчислювальної та прикл. математики. – 2007. – № 2. – С. 54–58.
5. Zadeh L. A. Fuzzy sets / L. A. Zadeh. // Inf. Contr. – 1965. – V.8. – P. 308–353.
6. Осовский С. Нейронные сети для обработки информации / С. Осовский. – М.: Финансы и статистика, 2002. – 343 с.
7. Shumetov V. G. Cluster analysis: an approach to the use of computers / V. G. Shumetov, L. Shumetova. – Orel: OrelGTU, 2000. – 118 с.

Апанасенко Д. В., асп.,  
КНУ імені Тараса Шевченка, Київ

### О МЕТОДАХ КЛАСТЕРИЗАЦИИ МНОЖЕСТВ СОСТАВНЫХ НЕЧЕТКИХ ЧИСЕЛ

*В статье рассмотрены и решены задачи группирования элементов нечеткого множества, заданного совокупностью составных нечетких чисел треугольного вида. Предложены и проанализированы несколько методов кластеризации. Проведено тестирование работы алгоритмов на конкретном примере.*

*Ключевые слова:* кластеризация, неопределенность, нечеткое треугольное число, методы группирования данных.

Apanasenko D. V., post-graduate,  
Kyiv National Taras Shevchenko University, Kyiv

### ABOUT CLUSTERIZATION METHODS FOR DATASETS OF COMPOSITE FUZZY NUMBERS

*In this article there are discussed and solved the problems of fuzzy set elements grouping. Fuzzy sets are formalized by complex fuzzy sets. A few methods of clusterization are proposed and analyzed. The use of the algorithm is provided on the real example.*

*Key words:* clusterization, uncertainty, fuzzy triangle number, methods of data grouping.

УДК 519.6

О. Ю. Грищенко, д-р фіз.-мат. наук, проф.,  
Г. О. Загородня, магістр,  
В. В. Оноцький, канд. фіз.-мат. наук, асист.,  
КНУ імені Тараса Шевченка, Київ

### ДВОКРОКОВИЙ СИМЕТРИЗОВАНИЙ АЛГОРИТМ ДЛЯ ЧИСЕЛЬНОГО МОДЕЛЮВАННЯ ПРОЦЕСУ РОЗПОВСЮДЖЕННЯ ВИПРОМІНЮВАННЯ В ДЕФОРМУЮЧОМУ СЕРЕДОВИЩІ

*Розглянуто математичну модель, що описує процес розповсюдження високоенергетичних пучків випромінювання, що опієється системою диференціальних рівнянь в частинних похідних з комплексними коефіцієнтами. Для чисельного моделювання запропоновано модифікацію двокрокового симетризованого різницевого алгоритму (ДС-алгоритму). Досліджено основні властивості даного алгоритму.*

*Ключові слова:* різницевої схеми, двокроковий симетризований різницевий алгоритм, ДС-алгоритм, чисельне моделювання, процес розповсюдження оптичного випромінювання.

**Вступ.** Загальні моделі адаптивної оптики запропоновано в 80-ті роки минулого сторіччя [1, 2]. Адаптивна (квазі-лінійна) оптика це наука, що в даний час знаходиться на грані між інженерними розробками та науковими дослідженнями. Вона вивчає процеси розповсюдження високоенергетичного оптичного випромінювання в середовищі, яке змінює свої властивості під дією енергетичного поля самого пучка випромінювання і тим самим деформує початковий напрям розповсюдження пучка.

Такі явища виникають при роботі лазерних приладів, як високоточних, що використовуються у медицині (лазерна хірургія, офтальмологія), так і потужних промислових облаштуваннях лазерної різки та зварювання об'єктів, тощо. Тепловий потік цих приладів приводить до нагрівання навколишнього середовища, оптичних елементів, а це часто призводить до відхилення променя та зменшення якості роботи. Нелінійність розповсюдження високоенергетичних пучків випромінювання також частково пояснюється нерівномірним збуренням середовища, в якому розповсюджується пучок. Ці збурення є наслідком впливу енергетичного поля випромінювання на щільність середовища, а отже і на коефіцієнт проникнення світла. Нерівномірний нагрів призводить до утворення місцевої деформуючої сили, що впливає на зміну напрямку розповсюдження пучка. Введення адаптивної корекції в цих умовах дозволяє компенсувати негативні ефекти.

Адаптивної корекції потребують як скануючі мікроскопи, так і потужні астрономічні телескопи, а також прилади, які досліджують атмосферу та проводять спостереження за наземними об'єктами з космосу. При роботі приладів у атмосферному середовищі значний вплив на якість роботи мають погодні умови, "прозорість" атмосфери та інші фактори.

Основним завданням чисельного моделювання таких процесів є дослідження енергетичних полів, можливість "викривлень" таких оптичних випромінювань та можливість їх компенсації.

В даній роботі розглядається функціонування системи, робота якої вивчається в умовах "теплого самозбурення". При цьому враховується, що методи, побудовані на основі чисто фазового керування пучком не дають повної інформації [1]. Тому в моделі враховуються як амплітудні так і фазові ефекти. Це приводить до дослідження комплексно-значних енергетичних полів.

**1. Постановка задачі.** Базова система рівнянь для моделювання пучка "типу гаусівського випромінювання", який розповсюджується в умовах теплового саморозігріву середовища має вигляд [2]

$$2ik \left[ \frac{\partial E}{\partial z} + a \frac{\partial E}{\partial t} \right] = \frac{\partial^2 E}{\partial x^2} + \frac{\partial^2 E}{\partial y^2} + bE + v_0 E + v_1 E,$$

$$\rho C_p \left[ \frac{\partial T}{\partial t} + (V \nabla) T \right] - k \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) = \alpha |E|.$$

тут  $E$  – комплексно-значна функція енергії пучка,  $T$  – теплота, яка виділяється при його випромінюванні,  $k, \rho, C_p, V, a, b, v_0, v_1, \alpha$  – фізичні коефіцієнти та параметри керування системою. В системі перше рівняння – це рів-



няння розповсюдження енергетичного пучка у середовищі, – друге визначає температурне поле, утворене енергією пучка. Потрібно за допомогою чисельного моделювання вибрати параметри  $v_0, v_1$  так, щоб одержати найбільш сприятливий стан функціонування системи, а саме створити ефективні енергетичне та теплове поля. Постановка задачі оптимального керування, коли  $E = E(T)$  є складною і мало дослідженою. Керуваність таких систем у лінійно-му випадку доводиться за допомогою теорії, розробленої професором С. І. Ляшком [3].

В для проведення чисельного моделювання вказаного процесу нами запропоновано модифікацію чисельного ДС-алгоритму до даної системи рівнянь. Досліджено основні властивості даного алгоритму.

В загальному випадку поставлену задачу потрібно досліджувати в два етапи. На першому, до входження в атмосферу, промінь рухається всередині приладу, випромінювання проходить в зоні застою (немає ніяких зовнішніх впливів і потік формувався лише за рахунок власної конвекції). Другий пов'язаний безпосередньо зовнішні середовищем, яке збурюється.

Одним з можливих варіантів такої моделі є початково-крайова задача для лінійного нестационарного рівняння, що описує процес розповсюдження гаусівського пучка з комплексною функцією розподілу енергії  $E(x, y, z, t)$  у випадково-неоднорідному слабкопоглинаючому середовищі [1]:

$$2i \left( k_1 \frac{\partial E}{\partial x} + k_2 \frac{\partial E}{\partial y} + k_3 \frac{\partial E}{\partial z} + \gamma \frac{\partial E}{\partial t} \right) = \frac{\partial^2 E}{\partial x^2} + \frac{\partial^2 E}{\partial y^2} + \alpha E + H, \quad (1)$$

в області

$$\Omega = \{ (x, y, z) | 0 < x < l_x, 0 < y < l_y, 0 < z < l_z \}, \quad 0 < t < t_1$$

з початковими

$$E|_{t=0} = f \quad (2)$$

та граничними умовами

$$E|_{\partial\Omega} = g. \quad (3)$$

тут  $x, y$  – координати в площині, перпендикулярній напрямку розповсюдження пучка,  $z$  – координата в напрямку його розповсюдження; всі коефіцієнти дійсні,  $k_i$  – дійсні хвильові функції, залежні від відповідної просторової координати,  $\gamma$  – величина, обернена до групової швидкості хвилі,  $\alpha$  – коефіцієнт, що враховує заломлення пучка, який в загальному випадку може залежати від температури середовища,  $g$  – експериментально визначена функція,  $E(x, y, z, t)$  – шукана функція для якої:  $E = u + iv$ ,  $H = \phi + i\psi$ .

Тоді

$$2i \left( k_1 \frac{\partial(u+iv)}{\partial x} + k_2 \frac{\partial(u+iv)}{\partial y} + k_3 \frac{\partial(u+iv)}{\partial z} + \gamma \frac{\partial(u+iv)}{\partial t} \right) = \frac{\partial^2(u+iv)}{\partial x^2} + \frac{\partial^2(u+iv)}{\partial y^2} + \alpha(u+iv) + \phi + i\psi$$

Розкривши дужки та врахувавши, що  $i^2 = -1$ , і виділивши з рівняння дійсну та уявну частини отримуємо систему:

$$2 \left( k_1 \frac{\partial u}{\partial x} + k_2 \frac{\partial u}{\partial y} + k_3 \frac{\partial u}{\partial z} + \gamma \frac{\partial u}{\partial t} \right) = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \alpha v + \psi, \quad (4)$$

$$2 \left( k_1 \frac{\partial v}{\partial x} + k_2 \frac{\partial v}{\partial y} + k_3 \frac{\partial v}{\partial z} + \gamma \frac{\partial v}{\partial t} \right) = -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} - \alpha u - \phi \quad (5)$$

при початкових та граничних умовах:

$$u|_{t=0} = \operatorname{Re} f, \quad v|_{t=0} = \operatorname{Im} f, \quad (6)$$

$$u|_{\partial\Omega} = \operatorname{Re} g, \quad v|_{\partial\Omega} = \operatorname{Im} g, \quad (7)$$

$f$  та  $g$  задані функції.

Далі рівняння (4)–(5) у подамо у вигляді:

$$\frac{\partial u}{\partial t} = L^{(1)}(u, v),$$

$$\frac{\partial v}{\partial t} = L^{(2)}(u, v),$$

де

$$L^{(1)}(u, v) = - \left( \frac{k_1}{\gamma} \frac{\partial u}{\partial x} + \frac{k_2}{\gamma} \frac{\partial u}{\partial y} + \frac{k_3}{\gamma} \frac{\partial u}{\partial z} \right) + \frac{1}{2\gamma} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + \frac{\alpha v}{2\gamma} + \frac{\psi}{2\gamma}$$

$$L^{(2)}(u, v) = - \left( \frac{k_1}{\gamma} \frac{\partial v}{\partial x} + \frac{k_2}{\gamma} \frac{\partial v}{\partial y} + \frac{k_3}{\gamma} \frac{\partial v}{\partial z} \right) - \frac{1}{2\gamma} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \frac{\alpha u}{2\gamma} - \frac{\phi}{2\gamma}$$

**2. Чисельна модель.** Широковживані явні, неявні, тришарові різницеві схеми, тощо, мають певні недоліки та переваги. Явні схеми прості у реалізації, але мають жорсткі умови стійкості, неявні схеми є безумовно стійкими, але приводять до великих систем алгебраїчних рівнянь, які потрібно розв'язувати на кожному часовому кроці. Тришарові схеми мають другий порядок апроксимації, але вимагають задання додаткових, непритаманних диференціальним задачам, умов. В роботі для розв'язання поставленої задачі використовується двокроковий алгоритм (ДС-алгоритм), в якому в кожній фіксованій просторовій точці на двох послідовних часових кроках за певним алгоритмом використовується явні та неявні схеми. Ідею послідовного в часі використання явних та неявних схем було анонсовано ще у 60 рр. Саульєвим [4] та Гурлі [5]. Для параболічних лінійних та нелінійних рівнянь та систем рівнянь Нав'є – Стокса цей метод досліджувався в наших роботах [6–9]. В даній роботі ДС-алгоритм, розповсюджений на системи рівнянь вказаних вище.

Ідея ДС-алгоритму полягає в наступному. На області  $\Omega \times [0, t_1]$  вводиться рівномірна сітка

$$\Omega_{nt} = \left\{ (x_i, y_j, z_k, t_n), i \in I_1, j \in I_2, k \in I_3, n \in I_4 \right\},$$

яка розбивається на дві підмножини:

$$\Omega_1^n = \left\{ (x_i, y_j, z_k, t_n) \mid (i + j + k + n) \bmod 2 = 0 \right\}$$

$$\Omega_2^n = \left\{ (x_i, y_j, z_k, t_n) \mid (i + j + k + n) \bmod 2 = 1 \right\}.$$

На Рис. 1 точки множини  $\Omega_1^n$  позначено хрестиками, а  $\Omega_2^n$  кружечками.

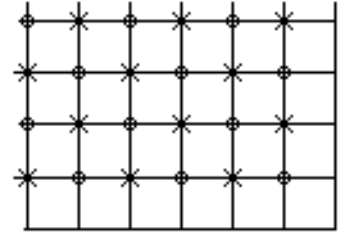


Рис. 1

Значення функцій  $u^{2n+1}, v^{2n+1}$  спочатку шукається у всіх точках сітки  $\Omega_1^{2n+1}$  за явними різницевиими схемами:

$$\begin{cases} u^{2n+1} = u^{2n} + \tau L_h^{(1)}(u^{2n}, v^{2n}), \\ v^{2n+1} = v^{2n} + \tau L_h^{(2)}(u^{2n}, v^{2n}), \end{cases} \quad (8)$$

де  $L_h^{(1)}$  та  $L_h^{(2)}$  - різницеві оператори, що апроксимують  $L^{(1)}$  та  $L^{(2)}$  відповідно

$$\begin{aligned} L_h^{(1)}(u_{i,j}^{2n}, v_{i,j}^{2n}) &= -\frac{k_1}{\gamma} \frac{u_{i+1,jk}^{2n} - u_{i-1,jk}^{2n}}{2h_x} - \frac{k_2}{\gamma} \frac{u_{ij+1k}^{2n} - u_{ij-1k}^{2n}}{2h_y} - \frac{k_3}{\gamma} \frac{u_{ijk+1}^{2n} - u_{ijk-1}^{2n}}{2h_z} + \\ &+ \frac{1}{2\gamma} \left( \frac{v_{i+1,jk}^{2n} - 2v_{ijk}^{2n} + v_{i-1,jk}^{2n}}{h_x^2} + \frac{v_{ij+1k}^{2n} - 2v_{ijk}^{2n} + v_{ij-1k}^{2n}}{h_y^2} \right) + \frac{a}{2\gamma} v_{ijk}^{2n} + \frac{\psi}{2\gamma} \\ L_h^{(2)}(u_{i,j}^{2n}, v_{i,j}^{2n}) &= -\frac{k_1}{\gamma} \frac{v_{i+1,jk}^{2n} - v_{i-1,jk}^{2n}}{2h_x} - \frac{k_2}{\gamma} \frac{v_{ij+1k}^{2n} - v_{ij-1k}^{2n}}{2h_y} - \frac{k_3}{\gamma} \frac{v_{ijk+1}^{2n} - v_{ijk-1}^{2n}}{2h_z} - \\ &- \frac{1}{2\gamma} \left( \frac{u_{i+1,jk}^{2n} - 2u_{ijk}^{2n} + u_{i-1,jk}^{2n}}{h_x^2} + \frac{u_{ij+1k}^{2n} - 2u_{ijk}^{2n} + u_{ij-1k}^{2n}}{h_y^2} \right) - \frac{a}{2\gamma} u_{ijk}^{2n} - \frac{\phi}{2\gamma} \end{aligned}$$

В точках множини  $\Omega_2^{2n+1}$  знаходиться решта значень  $u_{i,j,k}^{2n+1}, v_{i,j,k}^{2n+1}$  за допомогою неявних різницевиих схем:

$$\begin{cases} u^{2n+1} = u^{2n} + \tau L_h^{(1)}(u^{2n+1}, v^{2n+1}), \\ v^{2n+1} = v^{2n} + \tau L_h^{(2)}(u^{2n+1}, v^{2n+1}). \end{cases} \quad (9)$$

Значення  $u_{i,j,k}^{2n+1}$  та  $v_{i,j,k}^{2n+1}$  сприймаємо як допоміжні, а розв'язком вважаємо значення  $u_{i,j,k}^{2n+2}, v_{i,j,k}^{2n+2}$ , обчислені за тими ж розрахунковими формулами, але уже для множин  $\Omega_1^{2n+2}$  та  $\Omega_2^{2n+2}$  відповідно.

Відзначимо, що у даному випадку розв'язувати великі системи лінійних алгебраїчних рівнянь не потрібно. Дійсно. Перепишемо систему (9) у розгорнутому вигляді

$$\begin{aligned} u_{ijk}^{2n+1} + \left( \frac{1}{\gamma} \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) - \frac{a}{2\gamma} \right) v_{ijk}^{2n+1} &= u_{ijk}^{2n} + \tau \left( -\frac{k_1}{\gamma} \frac{u_{i+1,jk}^{2n+1} - u_{i-1,jk}^{2n+1}}{2h_x} - \frac{k_2}{\gamma} \frac{u_{ij+1k}^{2n+1} - u_{ij-1k}^{2n+1}}{2h_y} \right) + \\ + \tau \left( -\frac{k_3}{\gamma} \frac{u_{ijk+1}^{2n+1} - u_{ijk-1}^{2n+1}}{2h_z} + \frac{1}{2\gamma} \left( \frac{v_{i+1,jk}^{2n+1} + v_{i-1,jk}^{2n+1}}{h_x^2} + \frac{v_{ij+1k}^{2n+1} + v_{ij-1k}^{2n+1}}{h_y^2} \right) + \frac{\psi}{2\gamma} \right) \\ \left( \frac{1}{\gamma} \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) - \frac{a}{2\gamma} \right) u_{ijk}^{2n+1} + v_{ijk}^{2n+1} &= v_{ijk}^{2n} + \tau \left( -\frac{k_1}{\gamma} \frac{v_{i+1,jk}^{2n} - v_{i-1,jk}^{2n}}{2h_x} - \frac{k_2}{\gamma} \frac{v_{ij+1k}^{2n} - v_{ij-1k}^{2n}}{2h_y} \right) + \\ + \tau \left( -\frac{k_3}{\gamma} \frac{v_{ijk+1}^{2n} - v_{ijk-1}^{2n}}{2h_z} - \frac{1}{2\gamma} \left( \frac{u_{i+1,jk}^{2n} + u_{i-1,jk}^{2n}}{h_x^2} + \frac{u_{ij+1k}^{2n} + u_{ij-1k}^{2n}}{h_y^2} \right) - \frac{\phi}{2\gamma} \right). \end{aligned}$$

Позначимо

$$\begin{aligned} A &= \frac{1}{\tau}, \quad E = \frac{1}{\tau}, \quad B = \frac{1}{\gamma} \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) - \frac{a}{2\gamma}, \quad D = -\frac{1}{\gamma} \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) + \frac{a}{2\gamma} \\ C_{ijk} &= \frac{u_{ijk}^{2n}}{\tau} + \left( -\frac{k_1}{\gamma} \frac{u_{i+1,jk}^{2n+1} - u_{i-1,jk}^{2n+1}}{2h_x} - \frac{k_2}{\gamma} \frac{u_{ij+1k}^{2n+1} - u_{ij-1k}^{2n+1}}{2h_y} \right) + \\ + \left( -\frac{k_3}{\gamma} \frac{u_{ijk+1}^{2n+1} - u_{ijk-1}^{2n+1}}{2h_z} + \frac{1}{2\gamma} \left( \frac{v_{i+1,jk}^{2n+1} + v_{i-1,jk}^{2n+1}}{h_x^2} + \frac{v_{ij+1k}^{2n+1} + v_{ij-1k}^{2n+1}}{h_y^2} \right) + \frac{\psi}{2\gamma} \right) \\ F_{ijk} &= \frac{v_{ijk}^{2n}}{\tau} + \left( -\frac{k_1}{\gamma} \frac{v_{i+1,jk}^{2n} - v_{i-1,jk}^{2n}}{2h_x} - \frac{k_2}{\gamma} \frac{v_{ij+1k}^{2n} - v_{ij-1k}^{2n}}{2h_y} \right) + \end{aligned}$$

$$+ \left( -\frac{k_3}{\gamma} \frac{v_{ijk+1}^{2n} - v_{ijk-1}^{2n}}{2h_z} - \frac{1}{2\gamma} \left( \frac{u_{i+1jk}^{2n} + u_{i-1jk}^{2n}}{h_x^2} + \frac{u_{ij+1k}^{2n} + u_{ij-1k}^{2n}}{h_y^2} \right) - \frac{\varphi}{2\gamma} \right)$$

Тоді останню систему можна подати у вигляді:

$$\begin{cases} Au_{ijk}^{2n+1} + Bv_{ijk}^{2n+1} = C_{ijk} \\ Du_{ijk}^{2n+1} + Ev_{ijk}^{2n+1} = F_{ijk} \end{cases} \quad (10)$$

Отже, в кожній точці сітки маємо систему лінійних рівнянь, але лише з двома невідомими. Тобто, замість розв'язання системи з  $M = (N_x - 1)(N_y - 1)(N_z - 1)$  лінійних алгебраїчних рівнянь з  $M$  невідомими, що притаманно для неявних схем, ми знаходимо розв'язок за явними формулами

$$u_{ijk}^{2n+1} = \frac{\begin{vmatrix} C_{ijk} & B \\ F_{ijk} & E \end{vmatrix}}{\begin{vmatrix} A & B \\ D & E \end{vmatrix}} \quad \text{та} \quad v_{ijk}^{2n+1} = \frac{\begin{vmatrix} A & C_{ijk} \\ D & F_{ijk} \end{vmatrix}}{\begin{vmatrix} A & B \\ D & E \end{vmatrix}}.$$

Легко переконатися, що має місце твердження.

**Твердження.** На кожному часовому кроці система (10) має єдиний розв'язок.

Для доведення достатньо показати, що головний визначник системи не рівний нулю. Підрахуємо його значення

$$\begin{aligned} \Delta &= AE - BD = \frac{1}{\tau^2} - \left( \frac{1}{\gamma} \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) - \frac{a}{2\gamma} \right) \left( -\frac{1}{\gamma} \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) + \frac{a}{2\gamma} \right) = \\ &= \frac{1}{\tau^2} + \frac{1}{\gamma^2} \left( \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) - \frac{a}{2} \right)^2 > 0 \end{aligned}$$

Отже, система має єдиний розв'язок.

Отримані значення  $u^{2n+1}, v^{2n+1}$  ми вважаємо допоміжним розв'язком, а за розв'язок приймаємо значення, отримані на парних часових кроках. Рівняння для знаходження значень  $u^{2n+2}, v^{2n+2}$  на парних часових кроках отримуються за аналогічними формулами, що і для непарних часових кроків.

**Теорема (про порядок апроксимації)**

ДС-алгоритм (9)–(10) для задачі (4)–(7) має другий порядок апроксимації.

Доведення. Розглянемо перше рівняння (4) системи

$$2 \left( k_1 \frac{\partial u}{\partial x} + k_2 \frac{\partial u}{\partial y} + k_3 \frac{\partial u}{\partial z} + \gamma \frac{\partial u}{\partial t} \right) = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \alpha v + \psi.$$

Запишемо апроксимацію цього рівняння в довільній фіксованій точці  $(x_i, y_j, z_k)$  в момент часу  $2n+1$  (явна схема)

$$\begin{aligned} 2\gamma \frac{u_{ijk}^{2n+1} - u_{ijk}^{2n}}{\tau} + 2 \left( k_1 \frac{u_{i+1jk}^{2n} - u_{i-1jk}^{2n}}{2h_x} + k_2 \frac{u_{ij+1k}^{2n} - u_{ij-1k}^{2n}}{2h_y} + k_3 \frac{u_{ijk+1}^{2n} - u_{ijk-1}^{2n}}{2h_z} \right) = \\ = \frac{v_{i+1jk}^{2n} - 2v_{ijk}^{2n} + v_{i-1jk}^{2n}}{h_x^2} + \frac{v_{ij+1k}^{2n} - 2v_{ijk}^{2n} + v_{ij-1k}^{2n}}{h_y^2} + \alpha v_{ijk}^{2n} + \psi_{ijk}^{2n} \end{aligned}$$

та в момент часу  $2n+2$  (неявна схема)

$$\begin{aligned} 2\gamma \frac{u_{ijk}^{2n+2} - u_{ijk}^{2n+1}}{\tau} + 2 \left( k_1 \frac{u_{i+1jk}^{2n+2} - u_{i-1jk}^{2n+2}}{2h_x} + k_2 \frac{u_{ij+1k}^{2n+2} - u_{ij-1k}^{2n+2}}{2h_y} + k_3 \frac{u_{ijk+1}^{2n+2} - u_{ijk-1}^{2n+2}}{2h_z} \right) = \\ = \frac{v_{i+1jk}^{2n+2} - 2v_{ijk}^{2n+2} + v_{i-1jk}^{2n+2}}{h_x^2} + \frac{v_{ij+1k}^{2n+2} - 2v_{ijk}^{2n+2} + v_{ij-1k}^{2n+2}}{h_y^2} + \alpha v_{ijk}^{2n+2} + \psi_{ijk}^{2n+2} \end{aligned}$$

Додамо ці два рівняння та зведемо подібні доданки:

$$\begin{aligned} 2\gamma \left( \frac{u_{ijk}^{2n+1} - u_{ijk}^{2n}}{\tau} + \frac{u_{ijk}^{2n+2} - u_{ijk}^{2n+1}}{\tau} \right) + 2k_1 \left( \frac{u_{i+1jk}^{2n} - u_{i-1jk}^{2n}}{2h_x} + \frac{u_{i+1jk}^{2n+2} - u_{i-1jk}^{2n+2}}{2h_x} \right) + \\ + 2k_2 \left( \frac{u_{ij+1k}^{2n} - u_{ij-1k}^{2n}}{2h_y} + \frac{u_{ij+1k}^{2n+2} - u_{ij-1k}^{2n+2}}{2h_y} \right) + 2k_3 \left( \frac{u_{ijk+1}^{2n} - u_{ijk-1}^{2n}}{2h_z} + \frac{u_{ijk+1}^{2n+2} - u_{ijk-1}^{2n+2}}{2h_z} \right) = \\ = \alpha (v_{ijk}^{2n} + v_{ijk}^{2n+2}) + \psi_{ijk}^{2n} + \psi_{ijk}^{2n+2} + \left( \frac{v_{i+1jk}^{2n} - 2v_{ijk}^{2n} + v_{i-1jk}^{2n}}{h_x^2} + \frac{v_{i+1jk}^{2n+2} - 2v_{ijk}^{2n+2} + v_{i-1jk}^{2n+2}}{h_x^2} \right) + \\ + \left( \frac{v_{ij+1k}^{2n} - 2v_{ijk}^{2n} + v_{ij-1k}^{2n}}{h_y^2} + \frac{v_{ij+1k}^{2n+2} - 2v_{ijk}^{2n+2} + v_{ij-1k}^{2n+2}}{h_y^2} \right). \end{aligned}$$

Розкладемо тепер кожен доданок у ряд Тейлора відносно відповідної точки:  $u_{i\pm 1} = u_i \pm hu'_i + \frac{h^2}{2}u''_i \pm \frac{h^3}{3!}u'''_i + O(h^4)$ ,

де  $h_i$  – крок по відповідній змінній.

Тоді

$$\begin{aligned} \frac{u_{i+1jk}^{2n} - u_{i-1jk}^{2n}}{2h_x} &= \left( u'_x + \frac{h_x^2}{3} u''_x \right)^{2n} + O(h_x^3), & \frac{u_{i+1jk}^{2n+2} - u_{i-1jk}^{2n+2}}{2h_x} &= \left( u'_x + \frac{h_x^2}{3} u''_x \right)^{2n+2} + O(h_x^3) \\ \frac{u_{ij+1k}^{2n} - u_{ij-1k}^{2n}}{2h_y} &= \left( u'_y + \frac{h_y^2}{3} u''_y \right)^{2n} + O(h_y^3), & \frac{u_{ij+1k}^{2n+2} - u_{ij-1k}^{2n+2}}{2h_y} &= \left( u'_y + \frac{h_y^2}{3} u''_y \right)^{2n+2} + O(h_y^3) \\ \frac{u_{ijk+1}^{2n} - u_{ijk-1}^{2n}}{2h_z} &= \left( u'_z + \frac{h_z^2}{3} u''_z \right)^{2n} + O(h_z^3), & \frac{u_{ijk+1}^{2n+2} - u_{ijk-1}^{2n+2}}{2h_z} &= \left( u'_z + \frac{h_z^2}{3} u''_z \right)^{2n+2} + O(h_z^3) \\ \left( \frac{u_{ijk}^{2n+1} - u_{ijk}^{2n}}{\tau} + \frac{u_{ijk}^{2n+2} - u_{ijk}^{2n+1}}{\tau} \right) &= \left( \frac{u_{ijk}^{2n+2} - u_{ijk}^{2n}}{\tau} \right) = \left( 2u'_t + \frac{\tau^2}{3} u''_t \right)^{2n+1} + O(\tau^3) \\ \frac{v_{i+1jk}^{2n} - 2v_{ijk}^{2n} + v_{i-1jk}^{2n}}{h_x^2} &= (v''_x)^{2n} + O(h_x^2), \\ \frac{v_{i+1jk}^{2n+2} - 2v_{ijk}^{2n+2} + v_{i-1jk}^{2n+2}}{h_x^2} &= (v''_x)^{2n+2} + O(h_x^2) \\ \frac{v_{ij+1k}^{2n} - 2v_{ijk}^{2n} + v_{ij-1k}^{2n}}{h_y^2} &= (v''_y)^{2n} + O(h_y^2), \\ \frac{v_{ij+1k}^{2n+2} - 2v_{ijk}^{2n+2} + v_{ij-1k}^{2n+2}}{h_y^2} &= (v''_y)^{2n+2} + O(h_y^2) \end{aligned}$$

Підставивши це все у рівняння і скориставшись тим, що  $F(u^{2n} + u^{2n+2}) = 2F(u^{2n+1}) + O(\tau^2)$  отримаємо:

$$\begin{aligned} &2\gamma \left( 2u'_t + \frac{\tau^2}{3} u''_t \right)^{2n+1} + O(\tau^3) + 2k_1 \cdot 2 \left( u'_x + \frac{h_x^2}{3} u''_x \right)^{2n+1} + O(h_x^3) + \\ &+ 2k_2 \cdot 2 \left( u'_y + \frac{h_y^2}{3} u''_y \right)^{2n+1} + O(h_y^3) + 2k_3 \cdot 2 \left( u'_z + \frac{h_z^2}{3} u''_z \right)^{2n+1} + O(h_z^3) = \\ &= 2\alpha v_{ijk}^{2n+1} + 2\psi_{ijk}^{2n+1} + 2(v''_x)^{2n+1} + 2(v''_y)^{2n+1} \end{aligned} \tag{11}$$

Перепишемо наше початкове рівняння у вигляді  $2\gamma u'_t + 2k_1 u'_x + 2k_2 u'_y + 2k_3 u'_z = v''_x + v''_y + \alpha v + \psi$ , помножимо його на 2 та віднімемо від (11) на кроці  $2n + 1$ . Маємо:

$$\begin{aligned} &2\gamma \left( 2u' + \frac{\tau^2}{3} u'' \right)^{2n+1} + 2k_1 \cdot 2 \left( u' + \frac{h_x^2}{3} u'' \right)^{2n+1} + 2k_2 \cdot 2 \left( u' + \frac{h_y^2}{3} u'' \right)^{2n+1} + \\ &+ 2k_3 \cdot 2 \left( u' + \frac{h_z^2}{3} u'' \right)^{2n+1} - (4k_1 u'_x + 4k_2 u'_y + 4k_3 u'_z + 4\gamma u'_t)^{2n+1} + O(h_x^3) + O(h_y^3) + O(h_z^3) + \\ &+ O(\tau^3) = 2\alpha v_{ijk}^{2n+1} + 2\psi_{ijk}^{2n+1} + 2(v''_x)^{2n+1} + 2(v''_y)^{2n+1} - (2v''_x + 2v''_y + 2\alpha v + 2\psi)^{2n+1} \end{aligned}$$

Після зведення подібних доданків, отримаємо, що залишковий член похибки апроксимації рівний

$$\begin{aligned} \zeta &= 2k_1 \left( \frac{h_x^2}{3} u'' \right)^{2n+1} + 2k_2 \left( \frac{h_y^2}{3} u'' \right)^{2n+1} + 2k_3 \left( \frac{h_z^2}{3} u'' \right)^{2n+1} + \gamma \left( \frac{\tau^2}{3} u'' \right)^{2n+1} + \\ &+ O(h_x^3) + O(h_y^3) + O(h_z^3) + O(\tau^3). \end{aligned}$$

В силу довільності точки  $(x_i, y_j, z_k)$ , можна стверджувати, що похибка апроксимації розв'язку нашої задачі буде  $O(\tau^2 + h_x^2 + h_y^2 + h_z^2)$ . Для рівняння (5) за тією ж схемою провести дослідження та отримати аналогічний результат.

Якщо коефіцієнти системи різницевих рівнянь сталі, то безумовна стійкість ДС-алгоритму, як і у [6], встановлюється за допомогою методу фон Неймана. Отже за теоремою Філіпова – Лакса маємо, що при  $\tau \rightarrow 0$  і  $h \rightarrow 0$  чисельний розв'язок збігається до розв'язку диференціального рівняння зі швидкістю збіжності  $\tau^2 + h^2$ .

**Тестові обчислювальні експерименти.** Для верифікації алгоритму були проведено обчислювальні експерименти на двох моделях з відомим точним розв'язком.

**Задача 1.** Покладаємо:  $E = e^{i(wi+x+y+(0.5-w)z)}$ ,  $w = 1$ ,  $t_1 = 1$ ,  $l_x = l_y = l_z = \pi/2$ ,  $k_1 = k_2 = H \equiv 0$ ,  $k_3 = \gamma = \alpha = 1$ ,  $h_x = h_y = h_z = \pi/200$ ,  $\tau = 0.001$ .

Рис. 3 Максимальні значення абсолютних похибок для розв'язку  $u$  та  $v$ .

Рисунки 1 та 2 практично підтверджують стійкість запропонованого ДС-алгоритму. Абсолютна похибка алгоритму не перевищує 0,008 (див. рис. 3).

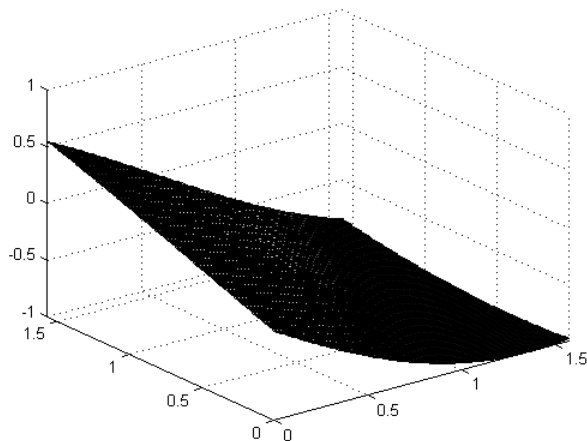


Рис. 1. Переріз поверхні графіка розв'язку  $u$  площиною  $x = \frac{\pi}{4}$  при  $t = 1$

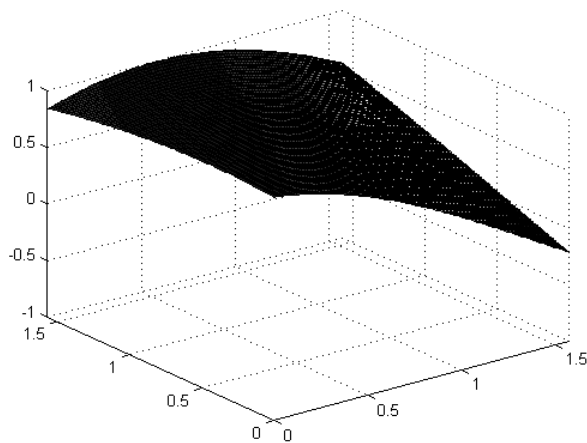


Рис. 2. Переріз поверхні графіка розв'язку  $v$  площиною  $x = \frac{\pi}{4}$  при  $t = 1$

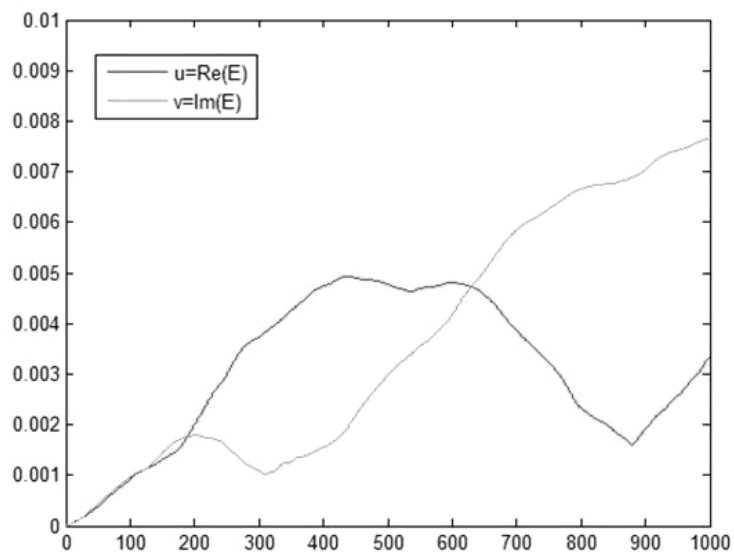


Рис. 3. Максимальні значення абсолютних похибок для розв'язку  $u$  та  $v$

Рисунки 1 та 2 практично підтверджують стійкість запропонованого ДС-алгоритму. Абсолютна похибка алгоритму не перевищує 0,008 (див. рис. 3).

**Задача 2.** Коефіцієнти  $k_i(x)$  вибиралися так, щоб при прямуванні  $x$  до  $\pi/2$  вони швидко затухали:

$$k_i(x) = \exp\left(-\left(\frac{\pi}{2} - x\right)^{-1}\right); \quad i = 1, 2, 3; \quad \gamma = \alpha = 1; w = 1; \quad l_x = l_y = l_z = \pi/2; t_1 = 1. \quad \text{Точним розв'язком цієї задачі є}$$

$$E = e^{i(wt+x+y+(0.5-w)z)}.$$

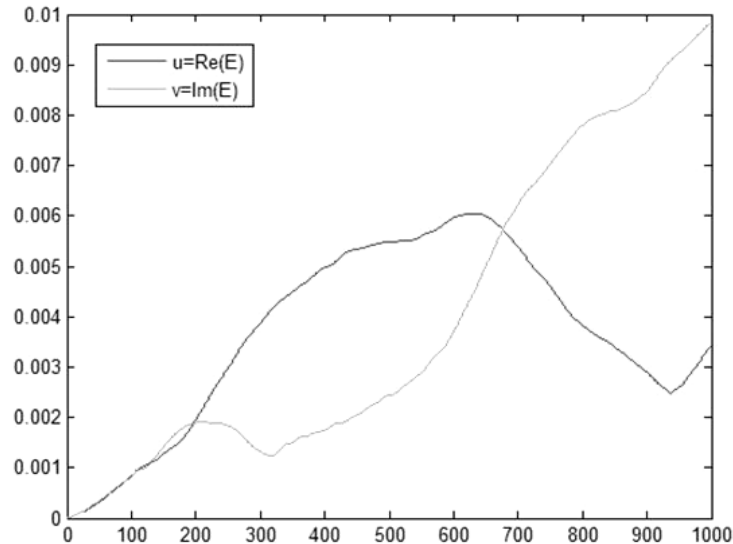


Рис. 4 Графіки максимальних абсолютних похибок для розв'язків  $u$  та  $v$

Можна помітити, що в даному випадку похибка досягає 0.01, що дає підставу вважати, що алгоритм дає більш точний розв'язок в задачі зі сталими коефіцієнтами, аніж із змінними.

**Висновки.** Запропоновано ефективний двокроковий симетризований алгоритм другого порядку апроксимації  $O(\tau^2 + h^2)$  для чисельного моделювання. Проведені чисельні експерименти на тестових задачах, наведені відповідні малюнки і графіки. Стійкість запропонованого ДС-алгоритму встановлена експериментально.

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Воронцов М. А. Принципы адаптивной оптики / Воронцов М. А., Шмальгаузен В. И. – М.: Наука, 1985. – 335 с.
2. Канев Ф. Ю. Адаптивная оптика. Численные и экспериментальные исследования / Канев Ф. Ю., Лукин В. П. – Томск.: Изд-во Института оптики атмосферы СО РАН, 2005. – 250 с.
3. Ляшко С. И. Обобщенное управление линейными системами / Ляшко С. И. – К.: Наукова думка, 1998. – 466 с.
4. Саульев В. К. Интегрирование уравнений параболического типа методом сеток / Саульев В. К. – М.: ГИМФЛ, 1960 г., – 324 с.
5. Gourlay A. R. Hopscotch a fast second-order partial differential equation solver / Gourlay A. R. – I. Inst. Math. Appl., 1970. – V.6. – P. 375–390.
6. Грищенко О. Ю. ДС-різницьвий алгоритм розв'язання крайових задач для параболических рівнянь другого порядку / Грищенко О. Ю. // Вісник Київського університету. Серія фіз.-мат. науки, 2000. – В.1. – С.227–231.
7. Gristhenko O. Yu. On an Algorithm for a Numerical Study of the Visualisation of Holographic Relief / Gristhenko O. Yu. // J. of Math. Sciences. - Kluwer Academic Press / Plenum Publishers, 2000. – Vol. 102. – No. 1. – P. 3749–3755.
8. Грищенко О. Ю. Двокроковий різницьвий алгоритм знаходження періодичних розв'язків рівнянь типу Шредінгера / Грищенко О. Ю., Слюсаренко В. Г., Оноцький В. В. // Вісник Київського національного університету імені Тараса Шевченка. Серія: фіз.-мат. науки, 2004. – Вып. №2. – С. 204–209.
9. Грищенко О. Ю. Модифікація ДС-алгоритму розв'язування початково-крайових задач для систем рівнянь Нав'є-Стокса / Грищенко О. Ю., Довбня П. І.; Журнал обчислювальної та прикладної математики, 2011. – №3(106). – С. 17–25.
10. Gryshchenko A. Yu. A two-stepsplitting algorithm in heat and mass transfer problems / Gryshchenko A. Yu. and. Martsafei A. S. // Cybernetics and Systems Analysis, 2011. – Vol.47. – Issue 6. – pp. 941–947.
11. Грищенко О. Ю. Numerical simulation of the process of optical radiation in the deforming medium / Грищенко О. Ю., Загородня Г. О., Оноцький В. В. // XXIV Міжнародна конференція "Problems of Decision Making under Uncertainties" (PDMU-2014), Чеський Рудолець (Чеська республіка), 1–5 вересня 2014 р. – с. 111–112.

Стаття надійшла до редколегії 25.09.15

Грищенко А. Е., д-р фіз.-мат. наук, проф.

Загородня А. А., магістр,

Оноцький В. В., канд. фіз.-мат. наук, ассист.

КНУ імені Тараса Шевченка, Київ

## ДВУШАГОВЫЙ СИММЕТРИЗОВАННЫЙ АЛГОРИТМ ДЛЯ ЧИСЛЕННОГО МОДЕЛИРОВАНИЯ ПРОЦЕССА РАСПРОСТРАНЕНИЯ ИЗЛУЧЕНИЯ В ИСКАЖАЮЩЕЙ СРЕДЕ

Рассмотрено математическую модель, которая описывает процесс распространения высокоэнергетических пучков излучения, который описывается системой дифференциальных уравнений в частных производных с комплексными коэффициентами. Для численного моделирования предложено модификацию двушагового симметризованного разностного алгоритма (ДС-алгоритма) Исследовано основные свойства данного алгоритма.

**Ключевые слова:** разностные схемы, двушаговый симметризованный разностный алгоритм, ДС-алгоритм, численное моделирование, процесс распространения оптического излучения.

Gryshchenko Oleksandr, Dr. Sci., prof.,  
Zagorodnia Ganna, stud.  
Onotskyi Viacheslav, Ph.D, assist.,  
Kyiv National Taras Shevchenko University, Kyiv

### SYMMETRIZED TWO-STEP DIFFERENCE ALGORITHM FOR THE NUMERICAL MODELING OF THE PROCESS OF RADIATION PROPAGATION IN DEFORMING ENVIRONMENT

The mathematical model, describing the process of high energy beams radiation propagation is considered. The model is described by a system of partial differential equations with complex coefficients. A two-step finite difference algorithm (DS-algorithm) is proposed for numerical modeling. The main properties of the algorithm are investigated.

**Key words:** finite difference schemes, two-step symmetrized finite difference algorithm, DS-algorithm, numerical modeling, propagation of optic radiation process.

УДК 519.87

Є. В. Івохін, д-р фіз.-мат. наук, проф.,  
КНУ імені Тараса Шевченка, Київ

### ВИКОРИСТАННЯ ПРОСТИХ ЧИСЕЛ ДЛЯ ІНТЕРВАЛЬНОГО ВИЗНАЧЕННЯ НЕЧІТКИХ ТРИКУТНИХ ЦІЛИХ ЧИСЕЛ

В статті запропоновано конструктивний підхід для представлення величини міри належності нечітких трикутних цілих чисел, в основу якого покладено ідею використання послідовності простих чисел спеціального вигляду. Запропоновано формалізацію операцій над нечіткими числами, наведено ряд практичних прикладів, в яких застосовується розроблена технологія. Запропонований підхід може бути використаний при розв'язанні різних задач підтримки прийняття рішень в умовах невизначеності.

**Ключові слова:** нечіткі множини та числа, прості числа, трикутні нечіткі числа, функція належності.

При проведенні різних досліджень в умовах невизначеності важливо коректно формалізувати процеси, що відбуваються в об'єкті дослідження. Дуже часто формалізація пов'язана з встановленням параметрів, які мають цілочисельний вигляд (кількість повторень, розмір об'єкта, бальна оцінка і т. і.). Врахування "наближеного" характеру згаданих параметрів можливо, наприклад, за допомогою нечіткого представлення цілих чисел, що базується на використанні нечітких трикутних чисел [1] та залучення спеціальних послідовностей простих чисел.

Класичне поняття нечіткої множини заданої універсальної множини  $X$  у відповідності до Заде [2] формулюється наступним чином.

**Означення 1.** [2] Нечіткою множиною  $\tilde{A}$  універсальної множини  $X$ , називають сукупність пар  $\tilde{A} = \{(\mu_{\tilde{A}}(x), x)\}$ , де  $\mu_{\tilde{A}}(x): X \rightarrow [0,1]$  – відображення множини  $X$  в одиничний відрізок  $[0,1]$ , яке називається функцією належності нечіткій множині.

Величина функції належності  $\mu_{\tilde{A}}(x)$  для елемента  $x \in X$  визначає ступінь належності до нечіткої множини. Інтерпретацією ступеня належності  $\mu_{\tilde{A}}(x)$  є суб'єктивна міра того, наскільки елемент  $x \in X$  відповідає поняттю, сенс якого формалізується нечіткою множиною  $\tilde{A}$ .

Визначимо в якості універсальної множини  $X$  простір над полем дійсних чисел  $R^1$ , тобто  $X = R^1$ . У цьому випадку розглядаються нечіткі величини, а при розв'язанні прикладних задач для формалізації нечіткості використовують інші більш конструктивні та прагматичні означення нечітких величин, що еквівалентні класичному означенню 1.

**Означення 2.** [3] Нечітким числом  $\tilde{b}$  називають впорядковану пару функцій  $(u(r), v(r))$ ,  $r \in [0,1]$ , які задовольняють наступним умовам

1.  $u(r)$  обмежена, неперервна зліва, неспадна функція на  $[0,1]$ ;
2.  $v(r)$  обмежена, неперервна зліва, незростаюча функція на  $[0,1]$ ;
3.  $u(r) \leq v(r)$ ,  $r \in [0,1]$ .

**Означення 3.** [1] Нечітким трикутним числом  $\tilde{b}$  називають впорядковану трійку чисел  $(a, b, c)$ ,  $a \leq b \leq c$ , що визначає функцію належності  $\mu_{\tilde{b}}(x)$  у вигляді:

1.  $\mu_{\tilde{b}}(x) = \frac{x-a}{b-a}$ ,  $x \in [a, b]$ ;
2.  $\mu_{\tilde{b}}(x) = \frac{c-x}{c-b}$ ,  $x \in [b, c]$ ;
3.  $\mu_{\tilde{b}}(x) = 0$ ,  $x \notin [a, c]$ .

Нечітке трикутне число  $\tilde{b}$ , задане у вигляді трійки  $(a, b, c)$ , іноді називають триплетом, при чому, для довільного числа  $x \in [a, b]$  справедливе представлення  $x = a + \lambda(b-a)$ , а для довільного  $x \in [b, c]$  –  $x = c - \lambda(c-b)$ , де  $0 \leq \lambda \leq 1$  – заданий рівень міри належності числа  $x$  нечіткій множині  $\tilde{b}$ . Нечітке трикутне число виду  $(a, b, b)$ , яке називається лівим нечітким трикутним числом, визначається функцією належності

$$\mu_{\tilde{b}}(x) = 0, x < a; \quad \mu_{\tilde{b}}(x) = \frac{x-a}{b-a}, x \in [a, b]; \quad \mu_{\tilde{b}}(x) = 1, x > b, \quad (3)$$

а нечітке трикутне число виду  $(b, b, c)$ , яке називається правим нечітким трикутним числом, – функцією належності

$$\mu_{\tilde{b}}(x) = 1, x < b; \quad \mu_{\tilde{b}}(x) = \frac{c-x}{c-b}, x \in [b, c]; \quad \mu_{\tilde{b}}(x) = 0, x > c. \quad (4)$$

Використання простих чисел для моделювання процесів у нечітких умовах запропоновано у [4]. В рамках підходу розглянуто ефективні обчислювальні схеми, які дозволяють отримувати значення функцій належності для елементів нечітких множин, що застосовуються для опису невизначеності параметрів процесу.

**Означення 4.** [4] Просте число  $P_j(a) \geq 0, a \geq 0, j \in \mathbb{Z}$ , яке належить інтервалу  $[a, \infty)$  при  $j \geq 0$  або інтервалу  $[0, a)$  при  $j < 0$  для заданого, необов'язково простого, цілого числа  $a \geq 0$ , будемо називати  $j$  – тим простим числом відносно числа  $a$ .

Розглянемо довільні прості числа  $P_j(a) \geq 0$  з порядковими номерами  $j \in \mathbb{Z}$  з множини простих чисел відносно числа  $a \geq 0$ . Нескладно перевірити, що справедливі властивості:

- 1)  $P_0(0) = 0, P_0(1) = 1, P_1(0) = 1;$
- 2)  $P_0(a) = a$ , якщо число  $a \geq 0$  – просте,  $P_0(a)$  – не існує, якщо  $a \geq 0$  – непросте;
- 3)  $P_j(a) \leq P_k(a)$ , якщо  $j \leq k, P_j(a) < P_k(a)$  при  $j < k, j \in \mathbb{Z}, k \in \mathbb{Z};$
- 4)  $P_j(a) = P_j(a+1) = \dots = P_j(a+l)$  для всіх  $1 \leq l < P_{j+1}(a) - P_j(a), j = 0, 1, 2, \dots, a \geq 0;$
- 5)  $P_j(a) = P_1(P_{j-1}(a)) = \dots = P_1(P_1(P_{j-2}(a))) = P_2(P_{j-2}(a)) = \dots = P_{j-1}(P_1(a))$ , якщо число  $a \geq 0$  – просте,  $j \in \mathbb{Z};$
- 6)  $P_j(a) \geq a$  для усіх  $j = 0, 1, 2, \dots$ , якщо число  $a \geq 0$  – просте,  $P_j(a) > a$  для усіх  $j = 1, 2, \dots$ , якщо  $a \geq 0$  – непросте;
- 7)  $P_j(a) < a$  для  $j = -1, -2, \dots, j_0$ , де номер  $j_0 < 0$  визначається як найменший індекс простого числа з загальної множини, для якого  $0 \leq P_{j_0}(a) < a$ .

**Означення 5.** Нечітким цілим числом  $\tilde{n}$  будемо називати впорядковану трійку чисел  $(k, n, l), k \leq n \leq l, k, n, l \in \mathbb{Z}$ , де

$$k = \begin{cases} P_{-1}(n), n \geq 0, \\ -P_1(-n), n < 0, \end{cases} \quad l = \begin{cases} P_1(n), n \geq 0, \\ -P_{-1}(-n), n < 0, \end{cases} \quad (5)$$

а  $P_1(\cdot), P_{-1}(\cdot)$  – наступне та попереднє прості числа відносно  $n, n \geq 0$ , та  $-n, n < 0$ .

Іншими словами, довільне нечітке ціле число  $\tilde{n}$  може бути представлене у вигляді триплету, ліве ( $k$ ) та праве ( $l$ ) значення якого подаються найближчими простими числами.

Такий підхід дозволяє для довільного  $n \in \mathbb{Z}$  задати нечітке ціле число  $\tilde{n} = (k, n, l)$ , не визначаючи діапазон представлення нечіткого числа  $[k, l]$ : можна покласти  $k$  та  $l$  у вигляді (5) і використати лінійну функцію належності

$$\mu_{\tilde{n}}(x) = \frac{x-k}{n-k}, x \in [k, n]; \quad \mu_{\tilde{n}}(x) = \frac{l-x}{l-n}, x \in [n, l]; \quad \mu_{\tilde{n}}(x) = 0, x \notin [k, l]. \quad (6)$$

Легко перевірити, що у випадках, коли  $k=0$  або  $l=0$ , зберігається справедливість представлення нечіткого цілого числа за означенням 5 з урахуванням властивості 1 для простих чисел, а при  $n=0$  маємо представлення нечіткого числа  $\tilde{0} = (0, 0, 1)$ .

Використання означення нечіткого цілого числа спрощує виконання традиційних арифметичних операцій додавання, віднімання, множення та ділення над нечіткими цілими числами. Для двох нечітких цілих  $\tilde{n}$  та  $\tilde{m}$ , заданих у вигляді нечітких трикутних чисел  $(k_n, n, l_n)$  та  $(k_m, m, l_m)$  відповідно, маємо

$$1. \tilde{n} + \tilde{m} = (k_+, n+m, l_+), \quad (7)$$

$$\text{де } k_+ = \begin{cases} P_{-1}(n+m), n+m \geq 0, \\ -P_1(-n-m), n+m < 0, \end{cases} \quad l_+ = \begin{cases} P_1(n+m), n+m \geq 0, \\ -P_{-1}(-n-m), n+m < 0, \end{cases}$$

$$2. \tilde{n} - \tilde{m} = (k_-, n-m, l_-), \quad (8)$$

$$\text{де } k_- = \begin{cases} P_{-1}(n-m), n-m \geq 0, \\ -P_1(-n+m), n-m < 0, \end{cases} \quad l_- = \begin{cases} P_1(n-m), n-m \geq 0, \\ -P_{-1}(-n+m), n-m < 0, \end{cases}$$

$$3. \tilde{n} * \tilde{m} = (k_*, n*m, l_*), \quad (9)$$

$$\text{де } k_* = \begin{cases} P_{-1}(n*m), n*m \geq 0, \\ -P_1(-n*m), n*m < 0, \end{cases} \quad l_* = \begin{cases} P_1(n*m), n*m \geq 0, \\ -P_{-1}(-n*m), n*m < 0, \end{cases}$$

$$4. \tilde{n} / \tilde{m} = (k_{div}, n/m, l_{div}), m \neq 0, \quad (10)$$

$$\text{де } k_{div} = \begin{cases} P_{-1}(n/m), n/m \geq 0, \\ -P_1(-n/m), n/m < 0, \end{cases} \quad l_{div} = \begin{cases} P_1(n/m), n/m \geq 0, \\ -P_{-1}(-n/m), n/m < 0, \end{cases}$$

$$5. \tilde{n} \% \tilde{m} = (k_{mod}, n \% m, l_{mod}), n \geq 0, m > 0, \quad (11)$$

$$\text{де } k_{mod} = \begin{cases} P_{-1}(n \% m), n \% m \geq 0, \\ -P_1(-n \% m), n \% m < 0, \end{cases} \quad l_{mod} = \begin{cases} P_1(n \% m), n \% m \geq 0, \\ -P_{-1}(-n \% m), n \% m < 0, \end{cases}$$

і при цьому результати не залежать від початкових інтервалів  $[k_n, l_n]$  та  $[k_m, l_m]$  представлення нечітких цілих чисел  $\tilde{n}$  та  $\tilde{m}$ .



Необхідно звернути увагу ще на одну важливу деталь. Обчислення простих чисел з великими номерами відносно довільного  $a \geq 0$  є досить ресурсоємним процесом. В той же час, знаходження простих чисел  $P_1(a)$  та  $P_{-1}(a)$  для довільного  $a \geq 0$  відбувається дуже швидко і не вимагає суттєвих обчислювальних витрат.

Інший приклад є більш теоретичним і стосується принципів формування нечітких множин. Розглянемо поняття довільного нечіткого числа. За традиційним означенням 1 нечітке число  $\tilde{b}$  подається у вигляді множини пар  $\tilde{b} = \{(b_i, \mu(b_i)) \mid b_i \in [a, c], i \in I, a \leq b \leq c, \mu(b_i) \in [0, 1], i \in I, \mu(b) = 1\}$ .

Іншими словами, представлення нечіткого числа  $\tilde{b}$  обов'язково залежить від самого числа  $b$  і характеризується параметрами у вигляді діапазону представлення  $[a, c]$  та функції належності. Для кожного з нечітких чисел необхідно визначити свій інтервал представлення і обумовити вигляд функції належності. На основі запропонованого представлення можна побудувати нечіткі числа, що не залежать від параметрів нечіткості.

Розглянемо трикутні нечіткі числа з лінійними функціями належності. За означенням 5 нечіткого цілого числа будь-яке число  $\tilde{b}$  подається у вигляді множини трійок  $\tilde{b} = \{(a, b, c)\}$ ,  $b \in Z$ ,  $a, c$  – прості числа, які обчислюються за формулою (6), а функція належності має вигляд

$$\mu_{\tilde{b}}(x) = \frac{x-a}{b-a}, x \in [a, b]; \quad \mu_{\tilde{b}}(x) = \frac{c-x}{c-b}, x \in [b, c]; \quad \mu_{\tilde{b}}(x) = 0, x \notin [a, c]. \quad (12)$$

Таким чином, для нечіткого числа  $\tilde{b}$  стає непотрібним визначення діапазону представлення  $[a, c]$ . Більше того, у цьому випадку можна визначити нові нечіткі поняття, наприклад, на основі нечіткого правила.

Розглянемо нечітко визначене просте число (більш коротко, нечітко просте число). Для звичайних цілих прості числа є підмножиною, яка чітко визначається правилом щодо кількості елементарних дільників. У випадку нечіткого цілого числа  $\tilde{b}$  кількість дільників числа  $b \in Z$  та їх задання потребують вводу додаткових представлень. Замість цього, пропонується розглядати нечітко просте число, яке для довільного  $b \in Z$  буде задаватися у вигляді нечіткого числа  $\tilde{b} = \{(a, b, c)\}$  з інтервалу  $[a, c]$ , ліва та права границі якого подаються найближчими простими числами.

На основі такого представлення довільне ціле число  $b \in Z$  є нечітко простим числом  $\tilde{b}$  з лінійною функцією належності (12) та представленням у вигляді трійок  $\tilde{b} = \{(a, b, c)\}$ . Наприклад, число 18 не є простим числом, однак воно є нечітко простим числом, яке подається у вигляді трійки  $\tilde{18} = \{(17, 18, 19)\}$ .

В якості практичного застосування введеного представлення нечіткого цілого числа можна навести два невеликі приклади. Перший з них стосується процесу фільтрації графічної інформації на основі аналізу віконних зображень, отриманих за допомогою прямокутних апертур [5]. Як правило, розмір апертури суттєво впливає на якість результатів фільтрації та на час роботи алгоритмів обробки. Невизначеність розмірів апертури може бути усунена встановленням діапазонів представлення ширини та довжини вікна апертури на базі нечітких цілих чисел. Більше того, виходячи з правила, за яким розміри задаються лише непарними цілими числами, можна розглядати тільки половину значень з кожного діапазону (для ширини та довжини).

Зрозуміло, що такий підхід легко розповсюдити на різні поняття, в основі яких лежать правила відбору або формалізації нечітко визначених даних. Наприклад, використовуючи запропонований спосіб для подання нечітких цілих значень, можна розширити опис типів в мовах програмування новими типами для позначення нечітких чисел. Вони можуть бути утворені на основі використання лексем для стандартних типів, перед якими записується ключове слово *fuzzy* (для цілих, відповідно, *fuzzy int* або *fuzzy long*). При цьому, якщо деяка змінна ініціалізується значенням відповідного стандартного (скалярного) типу, то описана з префіксом *fuzzy* змінна передбачає подання даного значення у вигляді нечіткого трикутного числа з носієм, що задається у формі інтервалу, побудованому на основі найближчих простих чисел.

Даний підхід дозволяє також розширити семантику мов програмування операторами, що працюють з нечіткими числами. Для прикладу, оператор *for* (*fuzzy int i=1; i<=n; i++*) передбачає, що змінна  $i$  пробігає значення від 0 до  $n$ , при цьому  $n$  вважається нечітким трикутним числом з інтервалом представлення носія у формі  $[P_{-1}(n), P_{+1}(n)]$ , де  $P_{-1}(\cdot)$ ,  $P_{+1}(\cdot)$  – попереднє та наступне відносно заданого цілого числа прості числа. Результатом виконання такого оператора вважається нечітке трикутне значення з лінійною трикутною функцією належності, яке визначено на інтервалі, що містить послідовні значення, отримані відповідно для різних інтервалів змінної циклу  $i \in [1, P_{-1}(n)]$ , ...,  $i \in [1, n]$ , ...,  $i \in [1, P_{+1}(n)]$ . Формально цей інтервал можна записати у вигляді

$$[\min(Rfor(P_{-1}(n)), \dots, Rfor(P_{+1}(n))), \max(Rfor(P_{-1}(n)), \dots, Rfor(P_{+1}(n)))] \quad (13)$$

де через  $Rfor(n)$  позначено результат роботи оператора *for* (*int i=1; i<=n; i++*).

У цьому випадку значення функції належності визначаються відповідними величинами функції належності числових елементів нечіткої величини  $n$ .

Нарешті, даний підхід надає можливість створення засобів обробки слабоформалізованої інформації, визначає спосіб пошуку кортежів нечіткого відношення нечіткої бази даних за кількісними атрибутами, який базується на реалізації SQL-запитів спеціального вигляду до нечітких даних [6, 7].

Отже, запропоновані результати дослідження дозволили обґрунтувати методику застосування трикутних нечітких чисел у вигляді інтервалів, межі яких вибираються з елементів спеціальних послідовностей простих чисел.

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Bablu Jana, Tapan Kumar Roy. Multi-Objective Fuzzy Linear Programming and Its Application in Transportation Model// Tamsui Oxford Journal of Mathematical Sciences, 2005. – 21(2). – P.243–268.
2. Zadeh L. A. Fuzzy sets // Inf. Contr. – 1965. – 8. – P. 338–353.

3. Dehghan M., Hashemi B. Iterative solution of fuzzy linear systems // Appl. Math. Comput., 2006. – 175. – P.645–674.  
 4. Івохін Є. В. Про застосування спеціальних множин простих чисел для визначення міри належності нечітких множин / Є. В. Івохін // Журнал обчислювальної та прикл. математики. – №4. – 2013. – С.87–94.  
 5. Яншин В. В. Обработка изображений на языке С для IBM PC / В. В. Яншин, Г. А. Калинин – М.: Мир, 1994. – 241 с.  
 6. Івохін Є. В. Про підхід до реалізації нечітких баз даних / Є. В. Івохін, К. О.Косинський // Вісник Київського університету. Серія: фіз.-мат. науки. – 2008. – Вип. 2. – С. 83–87.  
 7. Івохін Є. В. Про реалізацію кількісних запитів в нечітких базах даних / Є. В. Івохін, Д. О.Вадньов // Вісник Київського університету. Серія: фіз.-мат. науки. – 2015. – Вип. 2. – С. 93–96.

Стаття надійшла до редколегії 25.09.15

Івохін Е. В., д-р фіз.-мат. наук, проф.,  
 КНУ імені Тараса Шевченка, Київ

## ИСПОЛЬЗОВАНИЕ ПРОСТЫХ ЧИСЕЛ ДЛЯ ИНТЕРВАЛЬНОГО ОПРЕДЕЛЕНИЯ НЕЧЕТКИХ ТРЕУГОЛЬНЫХ ЦЕЛЫХ ЧИСЕЛ

В статье предложен конструктивный подход для представления величины меры принадлежности нечетких треугольных целых чисел, в основу которого положена идея использования последовательности простых чисел специального вида. Предложена формализация операций над нечеткими числами, приведен ряд практических примеров, в которых применяется разработанная технология. Предложенный подход может быть использован при решении разных задач поддержки принятия решений в условиях неопределенности.

**Ключевые слова:** нечеткие множества и числа, простые числа, треугольные нечеткие числа, функция принадлежности.

Ivokhin E. V., Dr. Sci., prof.,  
 Kyiv National Taras Shevchenko University, Kyiv

## A SIMPLE NUMBERS USING FOR INTERVAL DETERMINATION OF FUZZY TRIANGULAR INTEGERS

The article suggests a constructive approach to represent the values of the measures of fuzzy triangular integers, which is based on the idea of using a sequence of prime numbers of a special form. A formalization of operations on fuzzy numbers is proposed, some practical examples that use the the developed technology are given. The proposed approach can be used in solving various problems of decision-making support under uncertainty.

**Key words:** fuzzy sets and numbers, simple numbers, simple numbers, triangle fuzzy numbers, membership function.

УДК 519.925.51

В. Р. Кулян, канд. техн. наук, доц.  
 КНУ імені Тараса Шевченка, Київ  
 О. О. Юнькова, канд. фіз.-мат. наук, доц.,  
 Київський національний економічний університет імені Вадима Гетьмана, Київ

## МЕТОДИ ОПТИМАЛЬНОГО КЕРУВАННЯ В ЗАДАЧАХ ДИВЕРСИФИКАЦІЇ ПОРТФЕЛЯ ІНВЕСТИЦІЙ

Розглядається проблема дослідження оптимальної структури портфеля інвестицій. Для розроблених математичних моделей формування ринкової вартості однієї акції та портфеля акцій сформульовано задачі оптимального керування структурою таких інвестицій. Досліджено властивості та особливості застосування динамічних моделей, побудованих у класі систем звичайних диференціальних рівнянь, на прикладі нових задач оптимального інвестування у цінні папери. Розглянуто широкий спектр прикладних задач, пов'язаних з оптимізацією інвестиційних операцій з цінними паперами.

**Ключові слова:** математичне моделювання, оптимальне керування, портфель інвестицій

Як показано в роботах [1] та [2], математичні моделі формування ринкової вартості однієї акції та портфеля акцій у загальному вигляді можуть бути записані так

$$\dot{r}_i = \frac{dr_i}{dt} = f(r_i, t, \alpha), \quad r_i(t_0) = r_{i_0}, \quad t \in [t_0, T], \quad i = \overline{1, n} \quad (1)$$

i

$$\dot{r}_p = f^p(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i, t) \quad (2)$$

відповідно, та є заданими параметрично. Тут  $r_i$  – очікувана ринкова вартість акції;  $r_p$  – очікувана ринкова вартість інвестиційного портфеля;  $x_i$  – частка акцій  $i$  – того виду у портфелі;  $t$  – час.

Розглянемо для таких моделей нові прикладні задачі портфельного інвестування.

### Задача 1.

**Дано:** математична модель динаміки формування ринкової вартості інвестиційного портфеля (2); бажаний рівень прибутковості портфеля у момент часу  $T$   $r_p(T) = r_{pr}$ ; часовий інтервал  $t \in [t_0, T]$ ; обмеження на керування у кожен момент часу  $x(t) \in U(t)$ ; критерій якості

$$J(x(t)) = \int_{t_0}^T (r_p(t) - r_p(T))^2 dt + \Phi(r_p(t_0)) \rightarrow \min_{x(t) \in U(t)}$$

тут  $\Phi(r_p(t_0))$  – задана функція.

**Необхідно:** визначити  $x(t_0)$  та, відповідно,  $r_p(t_0)$ .

При цьому нагадаємо, що вектор  $x$  описує частки акцій різних видів у портфелі і є таким  $x = (x_1, x_2, \dots, x_n)$ . Тут  $n$  – кількість видів акцій у інвестиційному портфелі.

Для розв'язання поставленої вище задачі як задачі оптимального керування із одним закріпленим кінцем траєкторії та фіксованим часом застосуємо принцип максимуму. В результаті отримаємо можливість визначити  $x(t_0)$  і на його основі  $r_p(t_0)$ . Функція Гамільтона матиме вигляд

$$H(r_p(t), \psi(t), t, x(t)) = -(r_p(t) - r_p(T))^2 + \psi(t) * f(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i).$$

Необхідна умова її оптимальності за керуванням є такою

$$\frac{\partial(- (r_p(t) - r_p(T))^2 + \psi(t) * f(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i))}{\partial x_i} = 0, \quad i = \overline{1, n}. \quad (3)$$

Розв'язком рівняння (3) є функція керувань інвестиційним портфелем  $x^*(\psi(t), r_p(t), t)$ . Побудуємо спряжену систему у вигляді

$$\begin{aligned} \dot{\psi}(t) &= -\frac{\partial H(r_p(t), \psi(t), t, x(t))}{\partial r_p}, \\ \dot{\psi}(t) &= -2(r_p(t) - r_p(T)) + \psi(t) * f'_p(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i). \end{aligned}$$

Сформуємо умову трансверсальності на лівому кінці траєкторії

$$\psi(t_0) = -\frac{\partial \Phi(r_p(t_0))}{\partial r_p}.$$

Крайова задача принципу максимуму матиме вигляд

$$\begin{cases} \dot{\psi}(t) = -2(r_p(t) - r_p(T)) + \psi(t) * f'_p(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i) \\ \dot{r}_p(t) = f(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i) \end{cases}$$

при умовах  $\psi(t_0) = \psi_0$ ,  $r_p(T) = r_{pT}$ .

Аналізуючи математичну модель динаміки формування ринкової вартості однієї акції, видно, що права частина рівнянь залежить від параметрів  $\alpha_1, \alpha_2 \in A$ , де  $A$  – замкнута обмежена множина параметрів моделі.

Сформулюємо додаткові рівняння для коректної постановки задачі

$$\sum_{i=1}^n \int_{t_0}^T \psi(t) \frac{\partial f_i(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i)}{\partial \alpha_j} dt = 0, \quad j = 1, 2.$$

Скористаємось твердженнями теореми про оптимальність керування для задачі з одним закріпленим кінцем траєкторії та фіксованим часом. Розв'язавши побудовану систему звичайних диференціальних рівнянь, визначимо функції  $r_p(t)$ ,  $\psi(t)$ , які, будучи підставленими у розв'язок  $x^*(\psi(t), r_p(t), t)$ , дадуть можливість визначити структуру оптимального інвестиційного портфеля на вибраному інтервалі часу.

Аналіз описаного вище алгоритму дає можливість відмітити деякі недоліки, найбільш важливими із яких є параметрично задана цільова функція та суттєві обмеження на функцію керувань.

Математична процедура диверсифікації портфеля інвестицій дає можливість для динамічних математичних моделей однієї акції та портфеля акцій розв'язати задачу вибору початкового портфеля інвестицій при відомому значенні його очікуваної прибутковості на обраний у майбутньому момент часу.

### Задача 2.

**Дано:** математична модель динаміки формування ринкової вартості інвестиційного портфеля (2); прибутковість портфеля у початковий момент часу  $r_p(t_0) = r_{p0}$ ; бажаний рівень прибутковості портфеля у момент часу  $T$   $r_p(T) = r_{pT}$ ; часовий інтервал  $t \in [t_0, T]$ ; обмеження на керування у кожен момент часу  $x(t) \in X(t)$ ; критерій якості

$$J(x(T)) = r_p(T) = \sum_i x_i(T) r_i(T). \quad (4)$$

**Необхідно:** визначити оптимальні  $x(t_i)$ ,  $i = \overline{1, n}$ .

Сформульовану постановку можна розглянути як задачу оптимального керування із двома закріпленими кінцями траєкторії та фіксованим часом при обмеженнях на функцію керувань.

Для її розв'язання, як і для задачі з одним закріпленим кінцем траєкторії та фіксованим часом, можна застосувати процедуру принципу максимуму. При цьому слід звернути увагу на параметричне представлення математичної моделі динамічного формування ринкової вартості однієї акції. Тому, як і раніше, додаємо умову

$$\sum_{i=1}^n \int_{t_0}^T \psi_i(t) \frac{\partial f_i(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i)}{\partial \alpha_j} dt = 0, \quad j = 1, 2.$$

### Задача 3.

Як задача оптимального керування з фіксованим часом, закріпленим лівим кінцем та вільним правим кінцем траєкторії розглядається у розумінні максимуму функціонала

$$J = \sum_i x_i(T) r_i(T), \quad (5)$$

де  $x_i(t)$  – задані величини, які характеризують структуру портфеля. Варіюючи такими частками протягом вибраного інтервалу часу, будемо розв'язувати задачу оптимальної диверсифікації інвестиційного портфеля або з точки зору формального математичного моделювання, або як задачу оптимального керування складовими портфеля акцій.

Математичну модель портфеля акцій, як і раніше, розглянемо у вигляді

$$\frac{dr_p}{dt} = 2r_p(t) - \sum_i \sum_j x_i(t) r_i(t) \left( \frac{f_j}{r_j(t)} + \frac{dx_j(t)}{dt} \frac{1}{x_j(t)} \right), \quad i \neq j. \quad (6)$$

Граничну умову, що характеризує ринкову вартість портфеля акцій у початковий  $t_0$  момент часу сформулюємо у вигляді

$$r_p(t_0) = r_{p0}. \tag{7}$$

У задачі оптимального керування (5), (6), (7) за змінну фазового стану розглядається ринкова вартість портфеля акцій  $r_p$ .

Можемо припустити, що застосовуючи "наївну політику" інвестування для заданого інтервалу часу ці величини є постійними. Таким чином, необхідно побудувати оптимальний процес, що починається у заданій точці  $x_0$  для моменту часу  $t_0$  і надає функціоналу (14) оптимального значення.

$$J = \sum_i x_i r_i(T) = \sum_i x_i (r_i(T) - r_i(t_0)) + \sum_i x_i r_i(t_0) = \sum_i x_i \int_{t_0}^T f_i(r(t), x(t)) dt + const = \int_{t_0}^T \sum_i x_i f_i(r(t), x(t)) dt + const.$$

Із наведеного вище слідує, що  $J = I + const$ , де  $I$  – функціонал в задачі принципу максимуму.

Тому розв'язок даної задачі безпосередньо слідує із відповідної теореми про оптимальність розв'язку задачі оптимального керування із закріпленням часом [2]. Можна показати також, що, як і в задачі 1, побудова відповідних множин досяжності для моменту  $t_1 \in [t_0, T]$  може бути зведена до розв'язання серії задач мінімізації функціонала  $J = (x, r_p(T))$ . Варіюючи складовими вектора  $x$  і, розв'язуючи відповідні задачі оптимального керування для різних значень  $x$ , отримаємо точки границі множини досяжності і опорні гіперплощини у цих точках.

У розвиток розглянутої математичної моделі та формулювання для неї задачі оптимального керування можна побудувати послідовність оптимальних на інтервалах часу керувань, яка б враховувала можливості оптимальної диверсифікації портфеля акцій у фіксовані моменти часу.

Зважаючи на властивості математичної моделі (2), а саме на те, що задана вона у параметричному вигляді, розглянемо ще одну задачу оптимального керування портфелем інвестицій із закріпленням одним кінцем траєкторії та фіксованим часом у такому вигляді

$$\frac{dr_p(t, \alpha)}{dt} = f(r_p(t), x(t), \alpha, t), \tag{8}$$

$$r_p(t_0) = r_{p0},$$

$t_0 \leq t \leq T$ ,  $x(t) \in U$ , де  $U$  – множина допустимих керувань;  $\alpha \in A$  – вектор параметрів моделі. Для параметрично заданої математичної моделі (2) сформулюємо оптимізаційну задачу з таким критерієм якості

$$I(x(t), t) = \min_{x(t) \in U(t)} \left( \int_{t_0}^T f^0(r_p(t), x(t), \alpha, t) dt + \Phi(r_p(T)) \right).$$

Вважаємо, що функція  $\Phi(r_p(T))$  – відома. Тут моменти часу  $t_0, T$  – задані;  $x = (x_1, \dots, x_n)$ ,  $f^0 = (f_1^0, \dots, f_n^0)$ ; керування  $u_i(t) \in$  кусково-неперервними функціями на заданому відрізку;  $f_i(x, r_p, \alpha, t)$  та  $\Phi(r_p(T))$  – задані функції; параметр  $\alpha \in A$ , де  $A$  – обмежена замкнута множина. Таким чином, переходимо до розв'язання задачі

$$x^0 = \arg \min_{x \in U} \max_{\alpha \in A} I(x, r_p, \alpha),$$

де символом  $x^0$  позначено оптимальне керування.

На цьому етапі формулювання ітераційної оптимізаційної процедури задамо деяке початкове значення  $x_0 \in U$ , тоді відповідне значення параметрів моделі  $\alpha_0$  знайдемо як розв'язок задачі

$$\alpha_0 = \arg \max_{\alpha \in A} I(x_0, \alpha).$$

Відповідно на  $k$  – тому кроці процедури отримаємо

$$\alpha_k = \arg \max_{\alpha \in A} I(x_k, \alpha).$$

Згідно принципу максимуму, умова доповнюючої нежорсткості має вигляд

$$\phi(T) = -\nabla_{r_p} \Phi(r_p(T), r_p(T)), \tag{9}$$

де символом  $\nabla_{r_p} \Phi$  позначено градієнт функції  $\Phi$  за змінною  $r_p$ . Парі  $(r_p(t), x(t))$  поставимо у відповідність систему лінійних диференціальних рівнянь

$$\frac{d\phi_i(t)}{dt} = -\nabla_{r_p} f(x(t), r_p(t), t)^* \phi(t) + \nabla_{r_p} f^0(x(t), r_p(t), \alpha, t), \quad i = \overline{1, n}.$$

Ця система лінійна і однорідна, тому при довільних початкових умовах для  $\phi$  вона допускає єдиний розв'язок, визначений на всьому інтервалі часу, для якого побудуємо керування  $x(t)$  та відповідну фазову траєкторію  $r_p(t)$ . Значення  $\phi(T)$  для деякого крокового множника  $h$  та довільного моменту часу  $t$  такого, що  $t_0 < t \leq T$ , можемо визначити

$$\phi(t - h) = \phi(t) + h \nabla_{r_p} f(x(t), r_p(t), \alpha, t)^* \phi(t) - h \nabla_{r_p} f^0(x(t), r_p(t), \alpha, t).$$

Повторюючи наведену вище процедуру визначимо  $\phi(t_0)$ . Таким чином, маємо можливість сформулювати задачу Коші для  $\phi(t)$

$$\frac{d\phi(t)}{dt} = -\nabla_{r_p} f(x(t), r_p(t), \alpha, t)^* \phi(t) + \nabla_{r_p} f^0(x(t), r_p(t), \alpha, t), \quad \phi(t_0) = \phi_0,$$

і, підставивши знайдене значення функції  $\phi(t)$  в рівняння моделі, знайдемо оптимальний розв'язок  $x^0(t)$ , такий, що

$$x^0 = \arg \max_{x \in U} ((\phi^k(t), f_{r_p}(x, r_{p_k}, \alpha_k, t)) - f_{r_p}^0(x, r_{p_k}, \alpha_k, t)).$$

Скориставшись цим розв'язком, перейдемо до розв'язання траєкторної задачі

$$\begin{aligned} \dot{r}_p^0(t) &= f(x(t), r_p(t), t), \\ r_p(t_0) &= r_{p0}. \end{aligned}$$

Умову зупинки процедури побудови оптимального керування сформулюємо, використавши  $\Phi(x(T))$

$$\|\phi_k(T) - \nabla_x(\Phi(x_k(T)))\| < \varepsilon, \quad (10)$$

де  $\varepsilon > 0$  – деяке наперед задане число, яке задається в залежності від умов функціонування системи і характеризує відхилення розрахункової траєкторії  $r_{p_k}(t)$ , отриманої при керуванні  $x_k(t)$ , від умови на правому кінці.

За умови невиконання нерівності (19) завершимо пошук оптимальних значень параметрів математичної моделі (2), інакше – визначимо

$$\phi_{k+1}(T) = [\nabla_{r_p} \Phi(r_p^0(T), u^0) - \phi_k(T)]h + \phi_k(T),$$

і повторимо цикл операцій з формування нового керування  $x_k(t)$  та траєкторії фазових змінних  $r_{p_k}(t)$ .

#### Задача 4.

Актуальною постановкою задачі про оптимальне інвестування може бути формулювання, що використовує "програмну траєкторію". Така траєкторія, з огляду на властивості прикладної задачі, може бути побудована дослідником і змістом її буде бажаний рівень очікуваної прибутковості інвестиційного портфеля у кожен момент часу на обраному інтервалі. Позначимо її  $r^*(t)$ ,  $t \in [t_0, T]$ .

Формально постановка задачі може бути такою: для математичної моделі (2), за умов

$$r_p(t_0) = r_{p0}, \quad r_p(T) = r_{pT}, \quad t \in [t_0, T] \quad (11)$$

та критерію якості

$$J(r_p(t), t) = \int_{t_0}^T (r_p(t) - r^*(t))^2 dt \quad (12)$$

визначити функцію  $r_p(t)$ , яка на заданому інтервалі часу надає оптимального значення критерію якості (21) та задовольняє наведені умови.

Для розв'язання задачі (2), (11), (12) як задачі оптимального керування із двома закріпленими кінцями траєкторії та фіксованим часом, застосуємо процедуру принципу максимуму. Побудуємо функцію Гамільтона

$$\begin{aligned} H(\psi(t), t, u(t)) &= -(r_p(t) - r^*(t))^2 + \psi(t) * f(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i). \\ \frac{\partial (-(r_p(t) - r^*(t))^2 + \psi(t) * f(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i))}{\partial r_i} &= 0, \quad i = \overline{1, n}. \end{aligned} \quad (13)$$

Нехай розв'язком рівняння (13) є функція керувань інвестиційним портфелем  $x^*(t)$ . Побудуємо спряжену систему у вигляді

$$\dot{\psi}(t) = - \frac{\partial H(r_p(t), \psi(t), t)}{\partial r_p}.$$

Крайова задача принципу максимуму матиме вигляд

$$\begin{cases} \dot{\psi}(t) = -2(r_p(t) - r^*(t)) + \psi(t) * f'_{r_p}((r_p, x_i, \dot{x}_i, r_i, \dot{r}_i)) \\ \dot{r}_p(t) = f(r_p, x_i, \dot{x}_i, r_i, \dot{r}_i) \end{cases}$$

при умовах  $r_p(t_0) = r_{p0}$ ,  $r_p(T) = r_{pT}$ .

Скористаємось твердженнями теореми про оптимальність керування для задачі з двома закріпленими кінцями траєкторії та фіксованим часом.

Розв'язавши побудовану систему звичайних диференціальних рівнянь, визначимо функції  $r_p(t)$ ,  $\psi(t)$ , які, будучи підставленими у розв'язок рівняння (2), дадуть можливість визначити структуру оптимального інвестиційного портфеля на вибраному інтервалі часу.

Важливо відмітити, що у даній постановці задачі необхідно у кожен момент часу враховувати обмеження  $r_p(t) \in R_p(t)$ , де  $R_p(t)$  – обмежена замкнута множина допустимих портфелів.

**Висновки.** Наведені математичні процедури дають можливість для динамічних математичних моделей однієї акції та портфеля акцій розв'язати задачі оптимального управління інвестиціями і на основі розроблених алгоритмів визначити допустимі множини траєкторій динаміки ринкової вартості однієї акції для математичної моделі (1) та розв'язати задачу оптимальної диверсифікації інвестиційного портфеля ризикованих цінних паперів на основі моделі (2).

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гаращенко Ф. Г., Кулян В. Р., Рутицкая В. В. Моделирование и анализ динамики инвестиций. // Проблемы управления и информатики. – 2001. – № 6. – С.109–119.
2. Гаращенко Ф. Г., Кулян В. Р., Рутицкая В. В. Застосування методів практичної стійкості для розв'язування задач інвестиційного менеджменту. // Вісник Київського університету. Серія: фізико-математичні науки. – Випуск 3. – 2005р. – С.232–239.

3. Fedir G. Garashchenko, Viktor R. Kulian, Vladislava V. Rutitskaya Modelling and Analysis of Investment Trends. // Journal of Automation and Information. – New York, Connecticut – 2011. – v. 43, issue 12, – P.48–58.

4. Черноусько Ф. Л. Оценивание фазового состояния динамических систем. – М.: Наука, 1988. – С. 320.

Стаття надійшла до редколегії 26.10.15

Кулян В. Р., канд. техн. наук, доц.,  
КНУ імені Тараса Шевченка, Київ,  
Юнькова Е. А., канд. физ.-мат. наук, доц.,  
Київський національний економічний університет імені Вадима Гетьмана, Київ

## МЕТОДЫ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ В ЗАДАЧАХ ДИВЕРСИФИКАЦИИ ПОРТФЕЛЯ ИНВЕСТИЦИЙ

*Рассматривается проблема исследования оптимальной структуры портфеля инвестиций. Для построенных математических моделей формирования рыночной стоимости одной акции и портфеля акций сформулировано задачи оптимального управления структурой таких инвестиций. Исследованы свойства и особенности использования динамических моделей, построенных в классе систем обыкновенных дифференциальных уравнений, на примере новых задач оптимального инвестирования в ценные бумаги. Рассмотрен широкий спектр прикладных задач, связанных с оптимизацией инвестиционных операций с ценными бумагами.*

*Ключевые слова:* математическое моделирование, оптимальное управление, портфель инвестиций

Kulian V. R., Ph.D.  
Taras Shevchenko National University of Kyiv,  
Iunkova O. O., Ph.D. in Math  
Vadim Getman National Economy University of Kyiv

## METHODS OF OPTIMAL CONTROL PROBLEM IN THE DIVERSIFICATION OF THE PORTFOLIO

*The problem of the optimal structure of the portfolio is researched. To construct a mathematical model of the market value of one share and the share portfolio is formulated as optimal control problem of such investments. The properties and characteristics of the dynamic models, constructed in the class of ordinary differential equations, on the example of the new problems of optimal investment in securities are reviewed.*

*Key words:* Mathematical modeling, optimal control, investment portfolio

УДК 519.87

М. Ф. Махно, асп.,  
КНУ імені Тараса Шевченка, Київ

## ПРО ГИБРИДНУ МОДЕЛЬ ДИНАМІКИ ПРОЦЕСУ ОБРОБКИ СУКУПНОСТІ ЗАВДАНЬ

*В статті розглянуто та розв'язано задачу формалізації процесу обліку ресурсів обчислювального пристрою при обробці сукупності завдань у вигляді гібридної моделі з функціональних та диференціальних рівнянь. Отримано твердження про вигляд фундаментальної матриці розв'язків моделі. Проведено тестування її роботи для заданого способу обліку ресурсів.*

*Ключові слова:* гібридна модель, динаміка процесів, послідовність задач, фундаментальна матриця, задача Коші.

Одним із стандартних підходів, що використовуються при моделюванні динамічних систем, є опис еволюції станів системи шляхом задання початкових значень координат системи та рівнянь, які визначають зміну координат з часом. Складна динамічна система, як правило, представляє собою сукупність підсистем, для кожної з яких можлива власна математична формалізація. Традиційно система називається гібридною, якщо її підсистеми описуються різними типами моделей.

З іншої сторони, вивчення процесів керування складними системами зумовлює появу поняття ієрархічної структури взаємодії окремих складових (підсистем) системи та привело до формулювання специфічних для такого підходу математичних та інженерних проблем [1].

Однією з характерних особливостей складних ієрархічних систем є наявність у кожній підсистемі власної задачі керування. У даному випадку для керування процесами на рівні підсистем послідовно розв'язуються три основні задачі: отримання даних про об'єкт, аналіз структури й визначення процесів керування, а також використання результатів дослідження для оптимізації функціонування підсистем. Однак наявність у підсистем стратегій самоорганізації часто приводить до виникнення конфліктів в їх діях. Це, як правило, пов'язано з антагоністичністю критеріїв функціонування підсистем, що, у свою чергу, приводить до необхідності вирішення проблем узгодження їх діяльності.

Досить часто [2] розв'язування задач управління в складних ієрархічних системах здійснюють на прикладі дворівневих систем, що складаються з однієї підсистеми верхнього рівня та скінченної кількості ( $M$ ) підсистем нижнього. Такий підхід можна пояснити відносною простотою аналізу функціонування дворівневих ієрархічних систем і відповідністю більшості реальних систем дворівневій моделі взаємодії окремих їх складових.

Аналізу ієрархічних систем керування та дослідженню процесів взаємодії окремих підсистем присвячено багато наукових робіт [1, 3, 4, 5]. Докладно вивчено проблеми керування та координації процесів функціонування в таких системах. При цьому важливим залишається аналіз задач самоорганізації підсистем і впливу антагоністичних факторів їх діяльності на функціонування системи в цілому.

У даній роботі розглядається гібридна динамічна система, яка описує процес обробки довільної кількості задач (не більше заданого  $N$ ), які надходять на виконання на обчислювальний пристрій (процесор).

Припустимо, що задачі, які мають виконуватися, характеризуються поняттям складності, яку можна оцінити за обсягом необхідних для виконання ресурсів. Відповідно до цього обчислювальний пристрій виділяє необхідний об-

сяг ресурсів пропорційно складності кожної задачі, що виконується в даний момент. Обробка задачі у часі приводить до зменшення складності завдання зі швидкістю, обернено пропорційною загальній складності усіх задач.

Опис стану обчислювального пристрою може бути здійснений на основі підходу, що розглядається для моделювання процесів у нейронах. Роботу процесора опишемо за допомогою моделі функціонування нейроподібного елемента, що має  $n$  входів і один вихід. Стан елемента подається скалярною величиною завантаженості  $u(t)$ , яка називається потенціалом (величину  $u(t)$  можна інтерпретувати як усереднене за часом значення завантаженості обчислювального пристрою [6]).

Позначимо через  $x_i(t)$  – величину складності  $i$ -ї задачі у момент часу  $t$ ,  $i = 1, 2, \dots, N$ ;  $w(t) = (w_1(t), \dots, w_N(t))$  – вектор булівських компонент, одиничне значення яких ( $w_i(t) = 1$ ) визначає обслуговування, а нульове ( $w_i(t) = 0$ ) – не обслуговування  $i$ -ї задачі у момент часу  $t$ ,  $i = 1, 2, \dots, N$ ;  $b(t)$  – сумарну величину складності задач у момент часу  $t$ . Розглянемо процес функціонування обчислювального пристрою на інтервалі  $[t_0, T]$ , де  $T$  – довільне додатне число.

Потенціал процесора зростає за часом пропорційно зваженій сумі  $\frac{1}{N} \sum_{i=1}^N w_i(t) x_i(t)$  поточних рівнів складності задач  $x_i(t)$ ,  $i = 1, 2, \dots, N$ . Поява кожної нової задачі та завершення виконання задач визначають дискретні моменти часу  $t_0^i$  і  $t_E^i$ ,  $i = 1, 2, \dots, N$ , відповідно. Для всіх проміжків  $T_i = t_E^i - t_0^i$ ,  $i = 1, 2, \dots, N$ , можна знайти найбільше спільне кратне  $\Delta t$ , таке, що  $t_E^i = t_0^i + k_i \Delta t$ ,  $k_i \in \mathbb{N}$ ,  $i = 1, 2, \dots, N$ . Без обмеження загальності, можна вважати, що  $t_0 = 0$  і  $\Delta t = 1$ .

Інтервал роботи кожної задачі  $t \in [t_0^i, t_0^i + k_i \Delta t]$ ,  $i = 1, 2, \dots, N$ , в межах якого процесор здійснює виконання задачі, розглядається у вигляді суми проміжних часових інтервалів  $[t_0^i + k_j^i \Delta t, t_0^i + k_{j+1}^i \Delta t]$ ,  $j = 0, 1, 2, \dots, r_i$ ,  $i = 1, 2, \dots, N$ ,  $k_j^i \in \mathbb{N} \cup \{0\}$ ,  $k_0^i = 0$ ,  $\sum_{j=0}^{r_i} k_j^i = k_i$ . Це розбиття необхідне, тому що в межах кожного інтервалу виконання  $i$ -ї задачі відбуваються події, зв'язані з появою та завершенням інших задач. Зрозуміло, що якщо на інтервалі роботи  $i$ -ї задачі  $[t_0^i, t_0^i + k_i \Delta t]$ ,  $i = 1, 2, \dots, N$ , жодних інших подій не спостерігається, то  $r_i = 0$  і відповідно маємо  $k_0^i = 0$  та  $k_1^i = k_i$ .

Зміна сумарної величини складності задач  $b(t)$ , що виконуються у момент часу  $t$ , також пов'язана з моментами появи і завершення задач. Відповідно до запропонованих припущень щодо часової дискретизації визначаються моменти часу  $t_k = k \Delta t = k$ ,  $k \in \mathbb{N} \cup \{0\}$ , в які відбувається зміна величини  $b(t)$ . Нескладно перевірити, що

$$b(t_{k+1}) = b(t_k) + \sum_{\substack{i=1 \\ w_i(t_{k+1}) > w_i(t_k)}}^N 2^{i-1} - \sum_{\substack{i=1 \\ w_i(t_{k+1}) < w_i(t_k)}}^N 2^{i-1}, \quad b(t_0) = b(0) = 0, \quad (1)$$

і на кожному проміжку  $[t_k, t_{k+1})$  величина  $b(t) \equiv \text{const}$ . Крім цього, послідовність  $t_k = k$ ,  $k \in \mathbb{N} \cup \{0\}$  повністю визначає моменти подій, що відбуваються в системі і описуються за допомогою розглянутих вище значень  $t_0^i + k_j^i \Delta t$ ,

$$j = 0, 1, 2, \dots, r_i, \quad i = 1, 2, \dots, N, \quad k_j^i \in \mathbb{N} \cup \{0\}, \quad k_0^i = 0, \quad \sum_{j=0}^{r_i} k_j^i = k_i.$$

Таким чином, гібридна динамічна система, яка використовується для моделювання процесу обробки заданої кількості задач за допомогою обчислювального пристрою, складається з однієї лінійної функціональної підсистеми (1) та  $N+1$  лінійної диференціальної підсистеми, які записуються у вигляді

$$\dot{x}_i(t) = -\frac{1}{b(t)} x_i(t), \quad x_i(t_0^i) = x_0^i, \quad i = 1, 2, \dots, N, \quad (2)$$

$$t \in [t_0^i + k_j^i \Delta t, t_0^i + k_{j+1}^i \Delta t], \quad j = 0, 1, 2, \dots, r_i, \quad i = 1, 2, \dots, N, \quad k_j^i \in \mathbb{N} \cup \{0\}, \quad \sum_{j=0}^{r_i} k_j^i = k_i,$$

$$\dot{u}(t) = -u(t) + \frac{1}{N} \sum_{i=1}^N w_i(t) x_i(t), \quad u(t_0) = u(0) = 0, \quad t \in [t_0, T], \quad (3)$$

$$b(t) = b(t-1) + \sum_{\substack{i=1 \\ w_i(t) > w_i(t-1)}}^N 2^{i-1} - \sum_{\substack{i=1 \\ w_i(t) < w_i(t-1)}}^N 2^{i-1}, \quad b(0) = 0, \quad t = k, \quad k \in \mathbb{N} \cup \{0\}, \quad (4)$$

$$w_i(t) = \begin{cases} 0 \\ 1 \end{cases}, \quad i = 1, 2, \dots, N, \quad t = k, \quad k \in \mathbb{N} \cup \{0\}, \quad (5)$$

де  $x_0^i$  – початкова складність  $i$ -ї задачі, що поступає на виконання процесору,  $i = 1, 2, \dots, N$ .

Зважаючи на різну складність завдань, будемо впорядковувати наявні на обробці завдання за спаданням складності. Це не впливає на вигляд моделі, а лише призводить до перенумерації задач і, відповідно, до перестановки елементів вектора  $w(t) = (w_1(t), \dots, w_N(t))$  у моменти появи нових задач. При чому, якщо складність нової задачі дорівнює складності деяких поточних (наявних) задач, то формально вважаємо нову задачу більш складною (потребує деяких додаткових дій, не врахованих у моделі).

Відомо, що для лінійних стаціонарних диференціальних рівнянь вигляду

$$\dot{x}(t) = Ax(t), \quad x(t) \in \mathbb{R}^n, \quad t \geq 0, \quad (6)$$

розв'язок задачі Коші може бути представлений у вигляді добутку фундаментальної матриці рішень, нормованої при  $t = t_0 = 0$ , на вектор, що визначає початковий стан

$$x(t) = e^{At} x_0. \tag{7}$$

тут  $e^{At}$  матрична функція, що є матричним рядом, яка називається матричним експоненціалом [7].

Розглянемо узагальнену гібридну динамічну систему, складену зоднієї лінійної диференціальної стаціонарної підсистеми розмірності  $n$  та однієї лінійної функціональної підсистеми наступного вигляду

$$\dot{x}(t) = A(y(t))x(t), \quad x(t_0) = x(0) = x_0, \quad t \geq 0, \tag{8}$$

$$y(k) = C(k)x(k) + y(k-1), \quad k = 0, 1, 2, \dots \tag{9}$$

У даній системі  $A$ -квадратна  $(n \times n)$ - матриця з постійними коефіцієнтами на кожному з інтервалів  $t \in [k, k+1)$ ,  $k = 0, 1, 2, \dots$ ,  $x(t) \in R^n$ ,  $t \geq 0$ ,  $y(k) \in R^1$ ,  $k = 0, 1, 2, \dots$ ,  $C$ - вектор з  $n$  елементів. Під розв'язком системи (8), (9) будемо розуміти вектор-функцію  $(x_1(t), \dots, x_n(t), y(t))$ ,  $t \geq 0$ , що складається з векторної функції  $x(t)$ , яка є кусково неперервно-диференційовною та має розриви похідної у вузлових точках  $t = k$ ,  $k = 1, 2, \dots$ , і кусково-постійної функції  $y(t)$ , яка має зліва розриви першого роду у вузлових точках  $t = k$ ,  $k = 0, 1, 2, \dots$ . Початкові умови для розв'язку системи (8), (9) мають вигляд

$$x(0) = x_0, \quad y(t) \equiv 0, \quad -1 \leq t \leq 0. \tag{10}$$

Отримаємо представлення фундаментальної матриці рішень системи (8), (9) і, відповідно, розв'язок задачі Коші (8), (9), (10).

Вектор  $x_0$  початкових станів системи будемо записувати у вигляді  $x^0(0)$ , а розв'язок  $(x(t), y(t)) = (x_1(t), \dots, x_n(t), y(t))$ , визначений на проміжку  $k \leq t < k+1$ ,  $k = 0, 1, 2, \dots$ , – у вигляді  $(x^k(t), y(k))$ .

**Лема 1.** *Справедливе наступне співвідношення*

$$\int_{i-1}^i e^{A(1-s)} ds = \sum_{k=0}^{\infty} \frac{A^k}{(k+1)!} [(2-i)^{k+1} - (1-i)^{k+1}]. \tag{11}$$

Причому, якщо матриця  $A$  не вироджена, тобто існує  $A^{-1}$ , то

$$\int_{i-1}^i e^{A(1-s)} ds = A^{-1} [e^{A(i-2)} - e^{A(i-1)}]. \tag{12}$$

**Доведення.** Використовуючи вираз для матричного експоненціала  $e^{At}$  [7], отримуємо наступне

$$\begin{aligned} \int_{i-1}^i e^{A(1-s)} ds &= \int_{i-1}^i \left[ I + A \frac{1-s}{1!} + A^2 \frac{(1-s)^2}{2!} + \dots + A^k \frac{(1-s)^k}{k!} + \dots \right] ds = \\ &= \left[ -I \frac{1-s}{1!} - A \frac{(1-s)^2}{2!} - A^2 \frac{(1-s)^3}{3!} - \dots - A^k \frac{(1-s)^{k+1}}{(k+1)!} + \dots \right]_{s=i-1}^{s=i} = \\ &= \left[ I \frac{2-i}{1!} + A \frac{(2-i)^2}{2!} + A^2 \frac{(2-i)^3}{3!} + \dots + A^k \frac{(2-i)^{k+1}}{(k+1)!} + \dots \right] - \left[ I \frac{1-i}{1!} + A \frac{(1-i)^2}{2!} + A^2 \frac{(1-i)^3}{3!} + \dots + A^k \frac{(1-i)^{k+1}}{(k+1)!} + \dots \right]. \end{aligned}$$

Отже, маємо  $\int_{i-1}^i e^{A(1-s)} ds = \sum_{k=0}^{\infty} \frac{A^k}{(k+1)!} [(2-i)^{k+1} - (1-i)^{k+1}]$ , звідки випливає справедливість співвідношення (11).

Припустимо, що матриця  $A$  не вироджена, тобто існує  $A^{-1}$ . Тоді отриману вище залежність можна спростити наступним чином

$$\begin{aligned} \int_{i-1}^i e^{A(1-s)} ds &= A^{-1} \left[ I + A \frac{2-i}{1!} + A^2 \frac{(2-i)^2}{2!} + \dots + A^{k+1} \frac{(2-i)^{k+1}}{(k+1)!} + \dots - I \right] - \\ &- A^{-1} \left[ I + A \frac{1-i}{1!} + A^2 \frac{(1-i)^2}{2!} + \dots + A^{k+1} \frac{(1-i)^{k+1}}{(k+1)!} + \dots - I \right] = A^{-1} (e^{A(2-i)} - e^{A(1-i)}), \end{aligned}$$

звідки слідує справедливість співвідношення (12).

**Наслідок 1.** *Справедливе співвідношення*

$$\int_{k-1}^t e^{A(t-k+1-s)} ds = \sum_{j=0}^{\infty} \frac{A^j}{(j+1)!} [t^{j+1} - (1-k)^{j+1}]. \tag{13}$$

Причому, якщо матриця  $A$  не вироджена, тобто існує  $A^{-1}$ , то

$$\int_{k-1}^t e^{A(t-k+1-s)} ds = A^{-1} [e^{At} - e^{A(1-k)}]. \tag{14}$$

**Доведення.** Повторюючи викладення, наведені в лемі, отримаємо

$$\int_{k-1}^t e^{A(t-k+1-s)} ds = - \left( I \frac{t-k+1-s}{1!} + A \frac{(t-k+1-s)^2}{2!} + \dots + A^j \frac{(t-k+1-s)^{j+1}}{(j+1)!} + \dots \right)_{s=k-1}^{s=t} =$$



$$= \left[ I \frac{t}{1!} + A \frac{t^2}{2!} + \dots + A^j \frac{t^{j+1}}{(j+1)!} + \dots \right] - \left[ I \frac{1-k}{1!} + A \frac{(1-k)^2}{2!} + \dots + A^j \frac{(1-k)^{j+1}}{(j+1)!} + \dots \right] =$$

$$= \sum_{j=0}^{\infty} \frac{A^j}{(j+1)!} \left[ t^{j+1} - (1-k)^{j+1} \right].$$

Якщо матриця  $A$  не вироджена, то

$$\int_{k-1}^t e^{A(t-k+1-s)} ds = A^{-1} \left[ I + A \frac{t}{1!} + A^2 \frac{t^2}{2!} + \dots + A^{j+1} \frac{t^{j+1}}{(j+1)!} + \dots - I \right] -$$

$$- A^{-1} \left[ I + A \frac{1-k}{1!} + A^2 \frac{(1-k)^2}{2!} + \dots + A^{j+1} \frac{(1-k)^{j+1}}{(j+1)!} + \dots - I \right] = A^{-1} \left[ e^{At} - e^{A(1-k)} \right].$$

З використанням отриманих допоміжних результатів, знайдемо вигляд фундаментальної матриці рішень гібридної системи (8), (9).

**Теорема 1.** На проміжку  $k \leq t < k+1$  фундаментальна матриця розв'язків гібридної системи (8), (9) має вигляд

$$\Phi(t) = Z_k(t)Z(k), \quad (15)$$

$$Z_k(t) = \begin{bmatrix} e^{A(y(k))(t-k)} & 0 \\ 0 & 1 \end{bmatrix}, \quad k \leq t < k+1, \quad (16)$$

$$Z(k) = \begin{pmatrix} \prod_{j=1}^k e^{A(y(k-j))} & 0 \\ \sum_{j=1}^k C(k-j+1) \prod_{i=1}^{k+1-j} e^{A(y(k+1-j-i))} & 1 \end{pmatrix}, \quad j = \overline{1, k}. \quad (17)$$

**Доведення.** Розглянемо перший проміжок  $0 \leq t < 1$  інтегрування системи (8), (9). Враховуючи, що  $y(0) = 0$ , підсистема диференціальних рівнянь системи (8), (9) буде мати вигляд

$$\dot{x}^0(t) = A(y(0))x^0(t) = A(0)x^0(t), \quad 0 \leq t < 1.$$

Розв'язок  $x^0(t)$  підсистеми (8) на проміжку  $0 \leq t < 1$  має вигляд

$$x^0(t) = e^{A(y(0))t} x^0(0) = e^{A(0)t} x^0(0),$$

де  $e^{At}$  – матричний експоненціал  $e^{At} = I + A \frac{t}{1!} + A^2 \frac{t^2}{2!} + \dots + A^k \frac{t^k}{k!} + \dots$ , а загальний розв'язок системи (8), (9) записується у вигляді

$$\begin{pmatrix} x^0(t) \\ y(0) \end{pmatrix} = \begin{pmatrix} e^{A(0)t} x^0(0) \\ 0 \end{pmatrix}. \quad (18)$$

Розглянемо другий проміжок  $1 \leq t < 2$  інтегрування системи (8), (9). У цьому випадку розв'язок функціональної підсистеми має вигляд  $y(1) = C(1)x^1(1) + y(0)$ .

Виходячи умови неперервності векторної функції  $x(t)$  у вузловій точці  $t=1$ , маємо  $x^1(1) = x^0(1) = e^{A(0)}x^0(0)$ . Тоді розв'язок  $y(1)$  запишеться у вигляді

$$y(1) = C(1)x^0(1) + y(0) = C(1)e^{A(0)}x^0(0) + y(0). \quad (19)$$

Повернемося до першої підсистеми. На проміжку  $1 \leq t < 2$  вона приймає вигляд

$$\dot{x}^1(t) = A(y(1))x^1(t),$$

розв'язок  $x^1(t)$  якої записується у формі

$$x^1(t) = e^{A(y(1))(t-1)} x^1(1). \quad (20)$$

Враховуючи умови неперервності, отримуємо

$$x^1(t) = e^{A(y(1))(t-1)} x^1(1) = e^{A(y(1))(t-1)} x^0(1) = e^{A(y(1))(t-1)} e^{A(y(0))} x^0(0).$$

Продовжуючи цей процес далі, отримаємо, що на проміжку  $k \leq t < k+1$  розв'язок має вигляд

$$\begin{pmatrix} x^k(t) \\ y(k) \end{pmatrix} = \begin{bmatrix} e^{A(y(k))(t-k)} & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x^k(k) \\ y(k) \end{pmatrix}. \quad (21)$$

Розглянемо послідовно вузлові точки  $t = 1, 2, \dots, k-1, k$ . Як впливає з умов неперервності

$$\begin{pmatrix} x^1(1) \\ y(1) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ C(1) & 1 \end{bmatrix} \begin{pmatrix} x^0(1) \\ y(0) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ C(1) & 1 \end{bmatrix} \begin{pmatrix} e^{A(y(0))} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix} = \begin{pmatrix} e^{A(y(0))} & 0 \\ C(1)e^{A(y(0))} & 1 \end{pmatrix} \begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix},$$

$$\begin{pmatrix} x^2(2) \\ y(2) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ C(2) & 1 \end{bmatrix} \begin{pmatrix} x^1(2) \\ y(1) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ C(2) & 1 \end{bmatrix} \begin{pmatrix} e^{A(y(1))} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x^1(1) \\ y(1) \end{pmatrix} = \begin{pmatrix} e^{A(y(1))} & 0 \\ C(2)e^{A(y(1))} & 1 \end{pmatrix} \begin{pmatrix} x^1(1) \\ y(1) \end{pmatrix} =$$

$$= \begin{pmatrix} e^{A(y(1))} & 0 \\ C(2)e^{A(y(1))} & 1 \end{pmatrix} \begin{bmatrix} 1 & 0 \\ C(1) & 1 \end{bmatrix} \begin{pmatrix} e^{A(y(0))} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix} = \begin{pmatrix} e^{A(y(1))} & 0 \\ C(2)e^{A(y(1))} & 1 \end{pmatrix} \begin{pmatrix} e^{A(y(0))} & 0 \\ C(1)e^{A(y(0))} & 1 \end{pmatrix} \begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}, \dots, \quad (22)$$

$$\begin{aligned} \begin{pmatrix} x^k(k) \\ y(k) \end{pmatrix} &= \begin{bmatrix} 1 & 0 \\ C(k) & 1 \end{bmatrix} \begin{pmatrix} x^{k-1}(k) \\ y(k-1) \end{pmatrix} = \begin{bmatrix} 1 & 0 \\ C(k) & 1 \end{bmatrix} \begin{bmatrix} e^{A(y(k-1))} & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x^{k-1}(k-1) \\ y(k-1) \end{pmatrix} = \\ &= \begin{bmatrix} e^{A(y(k-1))} & 0 \\ C(k)e^{A(y(k-1))} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ C(k-1) & 1 \end{bmatrix} \begin{pmatrix} x^{k-2}(k-2) \\ y(k-2) \end{pmatrix} = \begin{bmatrix} e^{A(y(k-1))} & 0 \\ C(k)e^{A(y(k-1))} & 1 \end{bmatrix} \times \\ &\times \begin{pmatrix} e^{A(y(k-2))} & 0 \\ C(k-1)e^{A(y(k-2))} & 1 \end{pmatrix} \begin{pmatrix} x^{k-2}(k-2) \\ y(k-2) \end{pmatrix} = \dots = \begin{pmatrix} e^{A(y(k-1))} & 0 \\ C(k)e^{A(y(k-1))} & 1 \end{pmatrix} \begin{pmatrix} e^{A(y(k-2))} & 0 \\ C(k-1)e^{A(y(k-2))} & 1 \end{pmatrix} \dots \begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix} = \\ &= \begin{pmatrix} \prod_{j=1}^k e^{A(y(k-j))} & 0 \\ \sum_{j=1}^k C(k-j+1) \prod_{i=1}^{k+1-j} e^{A(y(k+1-j-i))} & 1 \end{pmatrix} \begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}. \end{aligned}$$

Враховуючи (21), маємо

$$\begin{pmatrix} x^k(t) \\ y(k) \end{pmatrix} = \begin{bmatrix} e^{A(y(k))(t-k)} & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} \prod_{j=1}^k e^{A(y(k-j))} & 0 \\ \sum_{j=1}^k C(k-j+1) \prod_{i=1}^{k+1-j} e^{A(y(k+1-j-i))} & 1 \end{pmatrix} \begin{pmatrix} x^0(0) \\ y(0) \end{pmatrix}, \quad k \leq t < k+1. \tag{23}$$

Таким чином, фундаментальна матриця розв'язків системи (8), (9) складається з двох частин, неперервної і дискретної. Як випливає (23), фундаментальна матриця рішень гібридної системи (8), (9) на проміжку  $k \leq t < k+1$  має вигляд  $\Phi(t) = Z_k(t)Z(k)$ , де  $Z_k(t)$  і  $Z(k)$  мають вигляд (16), (17) відповідно.

**Наслідок 2.** Для гібридної системи (8), (9) розв'язок задачі Коші з початковими умовами  $x(0) = x_0, y(t) \equiv y_0, -1 \leq t \leq 0$  на проміжку  $k \leq t < k+1$  має вигляд

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \Phi(t) \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \tag{24}$$

де фундаментальна матриця  $\Phi(t)$  задається у вигляді (15).

Повернемося до системи (2)–(5). Її можна записати у вигляді гібридної моделі (8),(9), динаміка станів якої задається вектором  $(x_1(t), \dots, x_N(t), u(t), y(t))$ , що складається з векторної функції  $x(t) = (x_1(t), \dots, x_N(t), x_{N+1}(t))$  ( $N+1$  змінна визначає стан потенціалу  $u(t)$  обчислювального пристрою),  $t \geq 0$ . Квадратна матриця  $A$  з постійними коефіцієнтами на кожному з інтервалів  $t \in [k, k+1), k = 0, 1, 2, \dots$ , розмірності  $(N+1 \times N+1)$  має вигляд

$$A = \begin{pmatrix} -1/b(t) & 0 & \dots & 0 & 0 \\ 0 & -1/b(t) & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & -1/b(t) & 0 \\ w_1(t)/N & w_2(t)/N & \dots & w_N(t)/N & -1 \end{pmatrix}, \tag{25}$$

$y(k) = b(t) \in R^1, k = 0, 1, 2, C$  вектор з  $N$  елементів, формальний результат застосування якого у виразі

$$C(k)x(k), k = 0, 1, 2, \dots, \text{ визначається величиною } \sum_{\substack{i=1 \\ w_i(t) > w_i(t-1)}}^N 2^{i-1} - \sum_{\substack{i=1 \\ w_i(t) < w_i(t-1)}}^N 2^{i-1}.$$

Без обмеження загальності моделі будемо вважати початковим моментом функціонування системи  $t = 0$  момент початку роботи першої задачі, виконання якої здійснюється обчислювальним пристроєм. Тоді розв'язок задачі Коші для системи (2)–(5) з початковими умовами  $x(0) = (x_1(0), \dots, x_N(0), 0) = x_0, y(t) = y_0 \equiv 0, -1 \leq t \leq 0$  на проміжку  $k \leq t < k+1$  буде мати вигляд

$$\begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \Phi(t) \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \tag{24}$$

де фундаментальна матриця  $\Phi(t)$  задається у вигляді (15).

Проведено обчислювані експерименти, в яких на основі отриманих теоретичних результатів здійснено моделювання процесів обробки заданої кількості задач за допомогою одного пристрою. Визначено шляхи оптимізації обсягів обчислювальних ресурсів в процесі виконання довільної сукупності задач.

**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Михалевич В. С. Вычислительные методы исследования и проектирования сложных систем / В. С. Михалевич, В. Л. Волкович. – М.: Наука, 1982. – 286 с.
2. Моисеев Н. Н. Элементы теории оптимальных систем / Н. Н. Моисеев. – М.: Наука, 1975. – 273 с.
3. Кухтенко А. И. Основные задачи теории управления сложными системами / А. И. Кухтенко // Сложные системы управления. – 1968. – Вып. 1. – С. 3–10.
4. Поспелов Г. С. Программно-целевое планирование и управление / Г. С. Поспелов, В. А. Ириков. – М.: Наука, 1976. – 261 с.
5. Хусаїнов Д. Я. Представление решений гибридных систем, составленных из дифференциальных и функциональных подсистем / Д. Я. Хусаїнов, С. В. Івохін, О. І. Кузьмич // Журнал обчислювальної та прикл. математики. – 2007. – № 1. – С. 110–116.
6. Jang J. S. Neuro-Fuzzy and soft computing / J.S. Jang, C.T. Sun, E. Mizutani – N. Y.: Prentice Hall, 1997. – 176 p.
7. Гантмахер Ф. Р. Теория матриц / Ф. Р. Гантмахер. – М.: Наука, 1966. – 576с.

Махно М. Ф., асп.  
КНУ імені Тараса Шевченка, Київ

### О ГИБРИДНОЙ МОДЕЛИ ДИНАМИКИ ПРОЦЕССА ОБРАБОТКИ СОВОКУПНОСТИ ЗАДАНИЙ

В статье рассмотрена и решена задача формализации процесса учета ресурсов вычислительного устройства при обработке совокупности заданий в виде гибридной модели из функциональных и дифференциальных уравнений. Получено утверждение о виде фундаментальной матрицы решенной системы. Проведено тестирование ее работы для заданного способа учета ресурсов.

**Ключевые слова:** гибридная модель, динамика процессов, последовательность задач, фундаментальная матрица, задача Коши.

Makhno M. F., post-graduate,  
Kyiv National Taras Shevchenko University, Kyiv

### ABOUT HYBRID MODEL OF DYNAMICS OF THE JOBS COLLECTION PROCESSING

In this article there are discussed and solved the problem of formalizing the process of resource accounting of computing device when processing a set of tasks. The hybrid model of functional and differential equations is proposed. An assertion about a fundamental matrix of the solutions was obtained. The use of the model is demonstrated for the given method of resource accounting.

**Key words:** hybrid model, processes dynamics, job collection, fundamental matrix, Cauchy problem.

УДК 517.929

А. В. Нікітін, канд. фіз.-мат. наук, доц.  
КНУ імені Тараса Шевченка, Київ

### УМОВИ СТІЙКОСТІ СТОХАСТИЧНИХ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ ІТО-СКОРОХОДА ЗІ СТАЛИМИ КОЕФІЦІЄНТАМИ У ГІЛЬБЕРТОВИХ ПРОСТОРАХ

Стаття присвячена дослідженню стійкості розв'язків стохастичних диференціальних рівнянь Іто-Скоророда зі сталими коефіцієнтами у гільбертових просторах

**Ключові слова:** Гільбертів простір, стійкість, стохастичні диференціальні рівняння.

**Вступ.** Одним з перспективних підходів дослідження випадкових процесів є їх представлення як кривих у гільбертовому просторі. Перенесення результатів, які стосуються стохастичних диференціальних рівнянь у скінченновимірних просторах на нескінченновимірний випадок є далеко не тривіальним [11]. Дана робота присвячена розвиненню цієї теорії і, зокрема, цікавим фактом, на думку автора, є зведення дослідження стійкості розв'язку стохастичного диференціального рівняння у гільбертовому просторі до розв'язності операторного рівняння Сільвестра.

#### 1. Стохастичні диференціальні рівняння Іто-Скоророда зі сталими коефіцієнтами у Гільбертових просторах.

Нехай  $X$  – сепарабельний гільбертовий простір зі скалярним добутком  $(x, y)$  і нормою  $|x|$ . на імовірнісному просторі  $(\Omega, F, P)$  з потоком  $\sigma$ -алгебр  $\{F_t, t \geq 0\} \subset F$  розглянемо стохастичне диференціальне рівняння

$$dX_t = X_t A dt + \sum_{k=1}^{\infty} (X_t B_k dW_k(t) + \int_U C_k(u) X_t \tilde{v}_k(dt, du)), \quad (1)$$

де  $A, B_k, C_k(u)$  – необмежені лінійні оператори, які визначені на деякій щільній в  $X$  множині  $D$ , причому для  $x \in D$

$$\sum_{k=1}^{\infty} |B_k x|^2 < \infty, \quad \sum_{k=1}^{\infty} |C_k(u) x|^2 < \infty,$$

$W_k(t)$  – незалежні між собою вінерівські процеси,  $\tilde{v}_k(dt, du)$  – незалежні між собою центровані пуассонівські міри.

Рівняння (1) розв'язується при заданій початковій умові  $X_0$ , незалежній від  $\{W_k, k = 1, 2, \dots\}$  та  $\{\tilde{v}_i, i = 1, 2, \dots\}$ . Нас цікавитиме випадок, коли рівняння (1) має розв'язок, що володіє другим моментом. Зауважимо, що розв'язком цього рівняння вважається такий сильний операторний процес  $X_t$ , для якого при  $x \in D$  функція  $X_t x$  має стохастичний диференціал, що співпадає з результатом дії на  $x$  правої частини (1).

Розглянемо сильні однорідні стохастичні півгрупи  $U_t^s$  з другими моментами. Введемо позначення для півгруп перших та других моментів:

$$M_t = M U_t^0, V_t(C) = M U_t^{0*} C U_t^0.$$

Відносно стохастичної півгрупи покладемо виконаною одну з властивостей неперервності:

а) півгрупа  $U_t^0$  сильно неперервна в середньому квадратичному, тобто

$$\lim_{t \rightarrow \infty} M |U_t^0 x - x|^2 = 0;$$

б) півгрупа  $U_t^0$  рівномірно неперервна в середньому квадратичному, тобто

$$\lim_{t \rightarrow \infty} \sup_{|x| \leq 1} M |U_t^0 x - x|^2 = 0.$$

Умові а) задовольняють розв'язки рівняння (1). Умова б) еквівалентна умові  $\lim_{t \rightarrow 0} M \|V_t(I) - I\| = 0$ .

Через  $Q(H)$  позначимо твірний оператор півгрупи  $V_t(H)$ :

$$Q(H) = \lim_{t \rightarrow 0} \frac{1}{t} [V_t(H) - H],$$

де границя операторів вважається слабкою. Зокрема, якщо стохастична півгрупа породжується рівнянням (1), то  $Q(H)$  задається формулою

$$Q(H) = A^*H + HA + \sum_{k=1}^{\infty} (B_k^*HB_k + \int_U C_k^*(u)HC(u)\Pi_k(du)). \tag{2}$$

Якщо  $H$  входить в область визначення твірного оператора  $Q$ , то справедливі рівняння:

$$\frac{d}{dt}V_t(H) = Q(V_t(H)) = V_t(Q(H)). \tag{3}$$

**Означення 1.** Стохастична півгрупа  $U_t^s$  називається рівномірно асимптотично стійкою в середньому квадратичному, якщо

$$\limsup_{t \rightarrow \infty} M |U_t^0 x|^2 = 0, \tag{4}$$

$$M |U_t^0 x|^2 = (V_t(I)x, x).$$

Співвідношення (4) еквівалентне умові  $\|V_t(I)\| \rightarrow 0$  при  $t \rightarrow \infty$ . Оскільки  $V_{t+s}(I) = V_t(V_s(I))$ ,

$$\|V_{t+s}(I)\| \leq V_t(\|V_s(I)\|I) \leq \|V_s(I)\| \|V_t(I)\|,$$

то з рівномірної асимптотичної стійкості в середньому квадратичному випливає експоненціальна стійкість в середньому квадратичному [1], тобто існує таке  $\alpha > 0$ , що

$$\|V_t(I)\| \leq \frac{1}{\alpha} \exp\{-\alpha t\}. \tag{5}$$

**Теорема 1.** Нехай стохастична півгрупа задовольняє умові б). Наступні твердження еквівалентні:

1)  $\lim_{t \rightarrow \infty} \|V_t(I)\| = 0$ ;

2) існує додатний оператор  $H$ , який має обмежений обернений, для якого  $Q(H) = -G$ , де  $G$  – додатний оператор, який має обмежений обернений;

3) для всіх  $x \in X \int_0^{\infty} (V_t(I)x, x) dt < \infty$ .

Якщо виконаюю є одна з цих умов, то  $P_x \{ \lim_{t \rightarrow \infty} |U_t^0 x| = 0 \} = 1$ .

**Доведення.** З твердження 1) випливає твердження 3) на підставі оцінки (5). Якщо виконана умова 2), яку можна подати у вигляді

$$Q(H) \leq \alpha H, \alpha > 0,$$

то

$$\frac{d}{dt}V_t(H) = V_t(Q(H)) \leq -\alpha V_t(H)$$

і, значить,

$$V_t(H) \leq \exp\{-\alpha t\} H, \|V_t(H)\| \leq \exp\{-\alpha t\} \|H\|.$$

Оскільки при деякому  $\delta > 0 \delta I \leq H$ , то

$$\|V_t(I)\| \leq \frac{1}{\delta} \|V_t(H)\| \leq \frac{\|H\|}{\delta} \exp\{-\alpha t\}.$$

Отже, з твердження 2) випливає твердження 1). Залишається показати, що з твердження 3) випливає твердження 2). Покладемо

$$C = \int_0^{\infty} V_t(I) dt.$$

Очевидно, що  $C$  – симетричний невід'ємний оператор. З умови б) випливає, що існує таке  $\delta > 0$ , що при  $t < \delta$   $\|V_t(I) - I\| < \frac{1}{2}$  і, значить,  $V_t(I) \geq \frac{1}{2}I$ . Тому

$$C \geq \int_0^{\delta} V_t(I) dt \geq \frac{\delta}{2} I.$$

Далі з урахуванням нерівності (5),

$$Q(H) = \int_0^{\infty} Q(V_t(I)) dt = \int_0^{\infty} \frac{d}{dt} V_t(I) dt = -I.$$

Отже,

$$Q(H) = -I \leq -\frac{1}{\|H\|} H.$$

Нехай виконана умова 2).

Тоді  $\exp\{\alpha t\}(CU_t^0 x, U_t^0 x)$  є супермартингалом: якщо  $F_t$  – потік  $\sigma$ -алгебр, породжених  $U_t^0$ , то

$$\begin{aligned} & M[\exp\{\alpha(t+h)\}(CU_{t+h}^0 U_t^0 x, U_{t+h}^0 U_t^0 x) / F_t] = \exp\{\alpha(t+h)\}(V_h(H)U_t^0 x, U_t^0 x) = \\ & = \exp\{\alpha t\}(CU_t^0 x, U_t^0 x) + \exp\{\alpha t\} \int_0^h \frac{d}{du} \exp\{\alpha u\}(V_u(C)U_t^0 x, U_t^0 x) du = \exp\{\alpha t\}(U_t^0 x, U_t^0 x) + \\ & + \exp\{\alpha t\} \int_0^h ([aV_u(C) + V_u(Q(C))]U_t^0 x, U_t^0 x) \exp\{\alpha u\} du = \exp\{\alpha t\}(CU_t^0 x, U_t^0 x) + \end{aligned}$$

$$+ \exp\{\alpha t\} \int_0^h (V_u(aC + Q(C))U_t^0 x, U_t^0 x) \exp\{\alpha t\} du \leq \exp\{\alpha t\} (CU_t^0 x, U_t^0 x)$$

(ми використали нерівність  $\alpha C + Q(C) \leq 0$ ). Тому,  $\exp\{\alpha t\} (CU_t^0 x, U_t^0 x)$  – величина обмежена та  $(CU_t^0 x, U_t^0 x) = 0 (\exp\{-\alpha t\})$ . Оскільки  $C > \delta I$  при деякому  $\delta > 0$ , то  $|U_t^0 x|^2 = 0 (\exp\{-\alpha t\})$ . Теорему доведено.

## 2. Ітераційна схема розв'язання операторного рівняння Сільвестра у гільбертовому просторі.

Розглянемо більш детально умову 2) теореми 1 попереднього пункту:

$$A^*H + HA + \sum_{k=1}^{\infty} (B_k^*HB_k + \int_U C_k^*(u)HC_k(u)\Pi_k(du)) = -G. \quad (6)$$

Конструкція (6) є узагальненим операторним рівнянням Ляпунова-Сільвестра. Дослідимо умови розв'язності такого рівняння.

Перепишемо (6) у вигляді

$$-A^*H - HA = G + \sum_{k=1}^{\infty} (B_k^*HB_k + \int_U C_k^*(u)HC_k(u)\Pi_k(du)). \quad (7)$$

Позначимо

$$F[H] = -A^*H - HA,$$

$$D[H] = G + \sum_{k=1}^{\infty} (B_k^*HB_k + \int_U C_k^*(u)HC_k(u)\Pi_k(du))$$

і побудуємо неявний розв'язок рівняння  $F[H] = D[H]$  за наступною ітераційною схемою

$$F[H_{k+1}] = D[H_k], \quad k = 0, 1, 2, \dots \quad (8)$$

Встановимо достатні умови збіжності методу (8) до розв'язку рівняння (7).

**Теорема 2.** Нехай виконуються наступні умови:

- 1) Спектр оператора  $A$  є від'ємним;
- 2)  $|T^*| \leq \beta$ ; де  $T \equiv A^* \otimes I + I \otimes A$ ,  $0 < \beta < 1$ ;
- 3)  $|\sum_{k=1}^{\infty} (B_k^*HB_k + \int_U C_k^*(u)HC_k(u)\Pi_k(du))| < \alpha H$ ,  $0 < \alpha < 1$ .

Тоді послідовність  $\{H_k, k \geq 1\}$  збігається до розв'язку рівняння (7).

**Доведення.** Оскільки оператор  $G$  є додатно визначеним, і виконується умова 1), то  $D$  – оператор, який переводить простір додатно визначених операторів у себе. Операторне рівняння Ляпунова

$$-A^*H - HA = K, \quad (9)$$

де  $K$  – довільний додатно визначений оператор, має розв'язком також додатно визначений оператор  $H$ , тобто  $F^{-1}$  також переводить простір додатно визначених операторів у себе (оскільки виконується умова 1)).

Введемо метрику наступним чином

$$\rho(H_1, H_2) = \|H_1 - H_2\|.$$

Простір додатно визначених операторів із введеною таким чином метрикою буде повним метричним простором. Тоді для збіжності ітераційної схеми залишилось перевірити виконання умови того, що оператори  $D$  і  $F^{-1}$  є операторами послідовного стиснення. Згідно умови 3), отримаємо

$$\rho(F(H_1), F(H_2)) < \alpha \rho(H_1, H_2).$$

Запишемо рівняння (9) у вигляді

$$TH = K,$$

де  $T \equiv A^* \otimes I + I \otimes A$  – тензорний добуток операторів. Тоді

$$H = T^{-1}K$$

і

$$\rho(F^{-1}(H_1), F^{-1}(H_2)) = \|T^{-1}H_1 - T^{-1}H_2\| \leq \beta,$$

що і означає виконання умови послідовного стиснення.

Теорему 2 доведено.

### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гихман И. И., Скороход А. В. Стохастические дифференциальные уравнения и их приложения. – Киев: Наукова думка, 1982. – 612 с.
2. Гихман И. И., Скороход А. В. Стохастические дифференциальные уравнения. – Киев: Наукова думка, 1968. – 354 с.
3. Далецкий Ю. Л., Крейн М. Г. Устойчивость решений дифференциальных уравнений в банаховом пространстве. – М.: Наука, 1970. – 536 с.
4. Данфорд Н., Шварц Дж. Т. Линейные операторы. 1. Общая теория. – М.: НЛ, 1962. – 895 с.
5. Дуб Дж. Л. Вероятностные процессы. – М.: НЛ, 1965. – 605 с.
6. Дынкин Е. Б. Марковские процессы. – М.: Физматгиз, 1963. – 859 с.
7. Жакод Ж., Ширяев А. Н. Предельные теоремы для случайных процессов. – Т.1. – М.: Наука, 1994. – 544 с.
8. Жакод Ж., Ширяев А. Н. Предельные теоремы для случайных процессов. – Т.2. – М.: Наука, 1996. – 628 с.
9. Кац И. Я. Метод функций Ляпунова в задачах устойчивости и стабилизации систем случайной структуры. – Екатеринбург: Изд-во Уральской госакадемии путей сообщения, 1998. – 222 с.
10. Колмановский В. Б., Носов В. Р. Устойчивость и периодические режимы регулируемых систем с последействием. – М. Наука, 1981. – 448 с.
11. Скороход А. В. Асимптотические методы теории стохастических дифференциальных уравнений. – Киев: Наукова думка, 1987. – 328 с.
12. Хасьяминский Р. З. Устойчивость систем дифференциальных уравнений при случайных возмущениях их параметров. – М.: Наука, 1969. – 367 с.
13. Хейл Дж. Теория функционально-дифференциальных уравнений. – М.: Мир, 1984. – 421 с.

Никитин А. В. канд. физ.-мат. наук, доц.  
КНУ имени Тараса Шевченка, Киев

## УСЛОВИЯ УСТОЙЧИВОСТИ СТОХАСТИЧЕСКИХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ИТО-СКОРОХОДА С ПОСТОЯННЫМИ КОЭФФИЦИЕНТАМИ В ГИЛЬБЕРТОВЫХ ПРОСТРАНСТВАХ

Посвящена исследованию устойчивости решений линейных стохастических дифференциальных уравнений с постоянными коэффициентами в гильбертовом пространстве.

**Ключевые слова:** Гильбертово пространство, устойчивость, стохастические дифференциальные уравнения

Nikitin A. V., Ph. D. Physics and Mathematics, assoc.prof.  
Taras Shevchenko National University of Kyiv

## STABILITY CONDITIONS ITO-SKHOROHOD STOCHASTIC DIFFERENTIAL EQUATIONS WITH CONSTANT COEFFICIENT IN HILBERT SPACE

This article is devoted to research of stability of decisions of linear stochastic differential equations with constant coefficients in Hilbert Space

**Key words:** Hilbert space, stability, stochastic differential equations

УДК 004.42:510.69

М. С. Нікітченко, д-р фіз.-мат. наук, проф.  
С. С. Шкільняк, д-р фіз.-мат. наук, проф.  
КНУ імені Тараса Шевченка, Київ

## ЧИСЛЕННЯ СЕКВЕНЦІЙНОГО ТИПУ ДЛЯ ПЕРЕВІРКИ ВИКОНУВАНOSTІ В ЛОГІКАХ КВАЗІАРНИХ ПРЕДИКАТИВ

Побудовано спеціальні числення для перевірки виконуваності множин формул в чистих першопорядкових логіках квазіарних предикатів – числення виконуваних множин. Проаналізовано застосовність традиційних секвенційних числень та числень виконуваних множин. Для пропонуванних числень доведено теореми коректності й повноти.

**Ключові слова:** логіка, предикат, числення, виконуваність, коректність та повнота

**Вступ.** Розвиток інформаційних технологій зумовлює розширення сфери застосування апарату математичної логіки (див., напр., [1]). Для успішного розв'язання низки задач, що виникають в сучасних програмних та інформаційних системах, необхідний ефективний пошук виведень. Потужним апаратом побудови виведень є числення секвенційного (ґенценівського) типу. Вони формалізують відношення логічного наслідку для пар множин формул. Широкий спектр таких числень розроблено для різних класів програмно-орієнтованих логічних формалізмів – композиційно-номінативних логік квазіарних предикатів (див., напр., [2–4]). Водночас для практичного пошуку виведень та перевірки виконуваності формул зазвичай використовують споріднений із секвенційними численнями метод семантичних таблиць (семантичних таблиць), який запропонував Е. Бет [5]. Семантичні таблиці подібні до виведень в секвенційних численнях – секвенційних дерев, проте вони не вимагають переписування незмінних формул. Водночас із практичного погляду семантичні таблиці ефективніші за секвенційні дерева. Проте семантичні таблиці не дають змогу відображати конкретні ситуації з максимальною простотою, секвенційні дерева більш наглядні в плані пошуку моделей чи контрмоделей. Тому в теоретичних дослідженнях доцільніше користуватись секвенційними численнями.

Для перевірки виконуваності формули чи множини формул використовують модифікований метод семантичних таблиць (див., напр., [6]), сутність якого полягає в пошуку моделі для формули чи множини формул. На відміну від традиційних, в таких модифікованих таблицях немає специфікації формул як істинні чи хибні, на кожному кроці виведення (побудови таблиці) використовується перевірка множини формул на виконуваність.

**Метою** даної роботи є побудова спеціальних числень секвенційного типу в стилі модифікованих семантичних таблиць для перевірки виконуваності в чистих першопорядкових логіках квазіарних предикатів. Такі числення назвемо численнями виконуваних множин. Виведення в цих численнях мають вигляд дерев, вершинами яких є множини формул. Проаналізовано застосовність числень виконуваних множин і традиційних секвенційних числень. Для пропонуванних числень виконуваних множин доведено теореми коректності й повноти.

Поняття, які тут не визначаються, тлумачимо в сенсі [2, 7]. Для зручності читання наведемо основні визначення.

**Основні поняття та визначення.** На першопорядкових рівнях предикати задаються на іменних множинах – множинах пар, перша компонента яких – ім'я, а друга – значення цього імені. Такі предикати названо квазіарними.

$V$ - $A$ -іменна множина ( $V$ - $A$ -IM) – це однозначна функція вигляду  $d: V \rightarrow A$ .

$V$ -IM подаємо як  $[V_1 \mapsto a_1, \dots, V_n \mapsto a_n, \dots]$ , де  $v_i \in V$ ,  $a_i \in A$ ,  $v_i \neq v_j$  при  $i \neq j$ . Множину всіх  $V$ - $A$ -IM будемо позначати  ${}^V A$ .

Функцію  $asn: {}^V A \rightarrow 2^V$  вводимо так:  $asn(d) = \{v \in V \mid v \mapsto a \in d \text{ для деякого } a \in A\}$ .

Операцію  $\|_{-x}$  видалення компонент з іменами із  $X \subseteq V$  задаємо так:  $d \|_{-x} = \{v \mapsto a \in d \mid v \notin X\}$ .

Замість  $d \|_{\{-x\}}$ , де  $x \in V$ , будемо скорочено писати  $d \|_{-x}$ .

Задамо операцію  $\nabla$  накладки IM  $d$  на IM  $h$ :  $h \nabla d = h \|_{-asn(d)} \cup d$ .

Параметричну операцію реномінації  $r_{x_1, \dots, x_n}^{v_1, \dots, v_n}: {}^V A \rightarrow {}^V A$  задаємо так:  $r_{x_1, \dots, x_n}^{v_1, \dots, v_n}(d) = d \nabla [v_1 \mapsto d(x_1), \dots, v_n \mapsto d(x_n)]$ .

Якщо множина пар імен реномінації відсутня, маємо реномінацію  $r$ , її трактуємо як тотожне відображення на  ${}^V A$ .

Введемо для  $y_1, \dots, y_n$  скорочене позначення  $\bar{y}$ . Тоді замість  $r_{x_1, \dots, x_n}^{v_1, \dots, v_n}$  також пишемо  $r_{\bar{x}}^{\bar{v}}$ .

Функцію вигляду  ${}^V A \rightarrow \{T, F\}$  назвемо  $V$ - $A$ -квазіарним предикатом. Тут  $\{T, F\}$  – множина істиннісних значень.

В цій роботі обмежимося розглядом часткових однозначних предикатів.

Клас  $V$ - $A$ -квазіарних предикатів позначимо  $Pr^A$ .

Кожний  $V$ - $A$ -квазіарний предикат  $P$  однозначно задається двома множинами – областями істинності та хибності.

Це множини  $T(P) = \{d \in {}^V A \mid P(d) = T\}$  та  $F(P) = \{d \in {}^V A \mid P(d) = F\}$ .

Для однозначних предикатів маємо  $T(P) \cap F(P) = \emptyset$ .

Предикат  $P: {}^V A \rightarrow \{T, F\}$  назвемо:

- неспростовним (частково істинним), якщо  $F(P) = \emptyset$ ;
- виконуваним, якщо  $T(P) \neq \emptyset$ .

Предметне ім'я  $x \in V$  (строго) неістотне для квазіарного предиката  $P$ , якщо  $d_1 \Vdash_x = d_2 \Vdash_x \Rightarrow P(d_1) = P(d_2)$ .

Базовими композиціями квазіарних предикатів на рівні чистих першопорядкових композитивно-номінативних логік (ЧКНЛ) є логічні зв'язки  $\neg$  та  $\vee$ , реномінації  $R_{\bar{x}}^{\bar{v}}$ , квантори  $\exists x$ .

Дамо визначення цих композицій через області істинності й хибності відповідних предикатів:

$$T(\neg P) = F(P); \quad F(\neg P) = T(P);$$

$$T(P \vee Q) = T(P) \cup T(Q); \quad F(P \vee Q) = F(P) \cap F(Q);$$

$$T(R_{\bar{x}}^{\bar{v}}(P)) = \{d \in {}^V A \mid r_{\bar{x}}^{\bar{v}}(d) \in T(P)\}; \quad F(R_{\bar{x}}^{\bar{v}}(P)) = \{d \in {}^V A \mid r_{\bar{x}}^{\bar{v}}(d) \in F(P)\};$$

$$T(\exists x P) = \{d \in {}^V A \mid P(\forall x \mapsto a) = T \text{ для деякого } a \in A\}; \quad F(\exists x P) = \{d \in {}^V A \mid P(d \forall x \mapsto a) = F \text{ для всіх } a \in A\}.$$

Композиції  $\rightarrow$ ,  $\&$ ,  $\forall x$  є похідними, вони задаються так:  $P \rightarrow Q = \neg P \vee Q$ ;  $P \& Q = \neg(\neg P \vee \neg Q)$ ;  $\forall x P = \neg \exists x(\neg P)$ .

Властивості композицій  $\neg$ ,  $\vee$ ,  $\rightarrow$ ,  $\&$  аналогічні властивостям відповідних класичних [8] логічних зв'язок (див [2, 7]).

Традиційні властивості композицій  $\exists x$  та  $\forall x$  в основному аналогічні властивостям відповідних класичних кванторів.

Для немонотонних квазіарних предикатів невірні [2, 7] деякі закони класичної логіки. Це пов'язано з тим, що значення  $P(d)$  може бути різним залежно від того, входить чи не входить до  $d$  компонента з певним іменем. Наприклад:

- існують спростовні предикати вигляду  $P \rightarrow \exists x P$ ;
- існують виконувани предикати вигляду  $\neg \exists x P \& P$ .

Справді, задамо предикат  $P$  так:  $P(d) = T$  при  $x \notin \text{asn}(d)$  та  $P(d) = F$  при  $x \in \text{asn}(d)$ . Для таких  $d$ , що  $x \notin \text{asn}(d)$ , маємо  $P(d) = T$  та  $\exists x P(d) = F$ .

Таким чином, при інтерпретаціях формул доцільно явно вказувати означені та неозначені предметні імена. Для цього використаємо спеціальні 0-арні композиції – параметризовані за предметними іменами предикати  $\varepsilon z$  (див. [9]), які визначають наявність в даних компоненти з відповідним іменем  $z$ . Предикати-індикатори  $\varepsilon z$  задаємо так:

$$T(\varepsilon z) = \{d \mid d(z) \uparrow\} = \{d \in {}^V A \mid z \notin \text{asn}(d)\}; \quad F(\varepsilon z) = \{d \mid d(z) \downarrow\} = \{d \in {}^V A \mid z \in \text{asn}(d)\}.$$

Наведемо основні властивості композицій  $R_{\bar{x}}^{\bar{v}}$  (див. також [2, 7]):

R)  $R(P) = P$  – тотожна реномінація.

RI)  $R_{z, \bar{x}}^{\bar{v}}(P) = R_{\bar{x}}^{\bar{v}}(P)$  – згортка тотожної пари імен у реномінації.

R $\neg$ )  $R_{\bar{x}}^{\bar{v}}(\neg P) = \neg R_{\bar{x}}^{\bar{v}}(P)$  – R $\neg$ -дистрибутивність.

R $\vee$ )  $R_{\bar{x}}^{\bar{v}}(P \vee Q) = R_{\bar{x}}^{\bar{v}}(P) \vee R_{\bar{x}}^{\bar{v}}(Q)$  – R $\vee$ -дистрибутивність.

RR)  $R_{\bar{x}}^{\bar{v}}(R_{\bar{y}}^{\bar{w}}(P)) = R_{\bar{x}}^{\bar{v}} \circ_{\bar{y}}^{\bar{w}}(P)$  – згортка реномінацій.

RU) Нехай  $z \in V$  неістотне для предиката  $P$ . Тоді  $R_{y, \bar{x}}^{\bar{v}}(P) = R_{\bar{x}}^{\bar{v}}(P)$ .

Ren) Якщо  $z$  неістотне для  $P$ , то  $\exists y P = \exists z R_z^y(P)$  – перейменування кванторного імені.

R $\exists$ s)  $R_{\bar{x}}^{\bar{v}}(\exists y P) = \exists y R_{\bar{x}}^{\bar{v}}(P)$ , якщо  $y \notin \{\bar{v}, \bar{x}\}$  – проста (обмежена) R $\exists$ -дистрибутивність.

R $\exists$ )  $R_{\bar{x}}^{\bar{v}}(\exists y P) = \exists z R_{\bar{x}}^{\bar{v}} \circ_z^y(P)$ , якщо  $z$  неістотне для  $P$  та  $z \notin \{\bar{v}, \bar{x}\}$  – R $\exists$ -дистрибутивність.

Семантичними моделями ЧКНЛ є композиційні системи квазіарних предикатів вигляду  $CS = (A, Pr^A, C)$ , де множина композицій  $C = \{\neg, \vee, R_{\bar{x}}^{\bar{v}}, \exists x\}$ . Така композиційна система визначає композиційну алгебру  $V$ - $A$ -квазіарних предикатів  $(Pr^A, C)$  та алгебру (алгебраїчну систему) даних  $(A, Pr^A)$ . Надалі розглядаємо ЧКНЛ, розширені шляхом виділення підмножини  $U \subseteq V$  тотально неістотних предметних імен, тобто імен, неістотних для всіх базових предикатів.

Побудова композиційної алгебри дає змогу визначити мову ЧКНЛ. Алфавіт мови: множина предметних імен (змінних)  $V$ , множина тотально неістотних предметних імен  $U \subseteq V$ ; множина  $Ps$  предикатних символів (ПС); множина  $\{\neg, \vee, R_{\bar{x}}^{\bar{v}}, \exists x\}$  символів базових композицій. Множина  $Fr$  формул мови ЧКНЛ визначається індуктивно:

- $Ps \subseteq Fr$ , формули вигляду  $p \in Ps$  – атомарні;
- $\Phi, \Psi \in Fr \Rightarrow \neg \Phi, \vee \Phi \Psi, R_{\bar{x}}^{\bar{v}} \Phi, \exists x \Phi \in Fr$ .

Множину  $Ps$  назвемо сигнатурою мови ЧКНЛ. Формули вигляду  $R_{\bar{x}}^{\bar{v}}(\Phi)$  назвемо  $R$ -формулами.

Атомарні формули та формули вигляду  $R_{\bar{x}}^{\bar{v}}(p)$ , де  $p \in Ps$ , назвемо примітивними.

Нехай  $\Gamma \subseteq Fr$ . Позначимо  $nm(\Gamma)$  множину всіх  $x \in V$ , які фігурують у символах реномінації та квантифікації, що входять до складу формул множини  $\Gamma$ .

Для множини  $\Gamma \subseteq Fr$  будемо скорочено позначати:

$$\bigcap_{\Phi \in \Gamma} T(\Phi_J) \text{ як } T^\wedge(\Gamma), \quad \bigcap_{\Phi \in \Gamma} F(\Phi_J) \text{ як } F^\wedge(\Gamma), \quad \bigcup_{\Phi \in \Gamma} T(\Phi_J) \text{ як } T^\vee(\Gamma), \quad \bigcup_{\Phi \in \Gamma} F(\Phi_J) \text{ як } F^\vee(\Gamma).$$

Символи  $Fs$  позначають (виділяють) базові предикати в множині  $Pr^A$ . Для опису такого позначення задаємо тотальне однозначне відображення  $I: Ps \rightarrow Pr^A$ . При цьому кожне  $z \in U$  неістотне для кожного  $P \in I(Ps)$ . Продовжимо  $I$  до відображення інтерпретації  $I: Fr \rightarrow Pr^A$  згідно побудови формул із простіших за допомогою символів композицій:

$$I(\neg \Phi) = \neg(I(\Phi)); \quad I(\vee \Phi \Psi) = \vee(I(\Phi), I(\Psi)); \quad I(R_{\bar{x}}^{\bar{v}} \Phi) = R_{\bar{x}}^{\bar{v}}(I(\Phi)); \quad I(\exists x \Phi) = \exists x(I(\Phi)).$$

Трійку  $J = (CS, Ps, I)$  назвемо *інтерпретацією* (моделлю) мови ЧКНЛ сигнатури  $Ps$ . Скорочено інтерпретації мови також позначаємо як  $(A, Ps, I)$  чи  $(A, I)$ .

Предикат  $I(\Phi)$ , який є значенням формули  $\Phi$  при інтерпретації  $J = (A, I)$ , позначаємо  $\Phi_J$ .

Ім'я  $x \in V$  неістотне для формули  $\Phi$ , якщо для кожної  $J = (A, I)$  ім'я  $x$  неістотне для  $\Phi_J$ .

Для кожної  $\Phi \in Fr$  задамо множину  $v(\Phi)$  гарантовано неістотних предметних імен.

Для кожного  $p \in Ps$  візьмемо  $v(p) = U$ , далі задаємо індукцією за побудовою формули:

$$v(\neg\Phi) = v(\Phi); v(\vee\Phi\Psi) = v(\Phi) \cup v(\Psi); v(\exists x\Phi) = v(\Phi) \cup \{x\}; v(R_{x_1, \dots, x_n}^{\vee_1, \dots, \vee_n} \Phi) = (v(\Phi) \cup \{v_1, \dots, v_n\}) \setminus \{x_i \mid v_i \notin v(\Phi), i \in \{1, \dots, n\}\}.$$

Нехай  $\Gamma \subseteq Fr$ . Тоді задаємо  $v(\Gamma) = \bigcap_{\Phi \in \Gamma} v(\Phi)$ .

Множину "нових" для формул множини  $\Gamma$  неістотних імен задамо так:  $fu(\Gamma) = U \setminus nm(\Gamma)$ .

Можна довести (див. [2, 7]):  $x \in fu(\Gamma) \Rightarrow x \in v(\Gamma) \Rightarrow x$  неістотне для кожної  $\Phi \in \Gamma$ .

Формула  $\Phi$  неспростовна при інтерпретації  $J$ , якщо предикат  $\Phi_J$  неспростовний, тобто  $F(\Phi_J) = \emptyset$ .

Формула  $\Phi$  неспростовна (частково істинна), якщо  $\Phi$  є неспростовною при кожній інтерпретації  $J$ .

Формула  $\Phi$  виконувана при інтерпретації  $J$ , якщо предикат  $\Phi_J$  виконуваний, тобто  $T(\Phi_J) \neq \emptyset$ .

Формула  $\Phi$  виконувана, якщо  $\Phi$  виконувана при деякій інтерпретації  $J$ .

Множина  $\Gamma \subseteq Fr$  виконувана, якщо  $T^\wedge(\Gamma) \neq \emptyset$  для деякої інтерпретації  $J$ .

Це означає, що існують інтерпретація  $J = (A, I)$  та  $\delta \in {}^V A$  такі:  $\Phi_J(\delta) = T$  для всіх  $\Phi \in \Gamma$ .

Таку пару  $(J, \delta)$  назвемо моделлю множини формул  $\Gamma$ .

Отже,  $\Gamma$  виконувана  $\Leftrightarrow \Gamma$  має модель.

**Відношення логічного наслідку.** Для логік квазіарних предикатів можна ввести [7] низку відношень логічного наслідку для множин формул. Як зазначено вище, ми розглядаємо логіки однозначних предикатів.

Задамо відношення наслідку для множин формул при фіксованій інтерпретації. Нехай  $\Gamma, \Delta \subseteq Fr, J$  – інтерпретація.

$\Delta \in ID$ -наслідком  $\Gamma$  при  $J$  (позн.  $\Gamma \models_{ID} \Delta$ ), якщо  $T^\wedge(\Gamma) \cap F^\wedge(\Delta) = \emptyset$ .

$\Delta \in T$ -наслідком  $\Gamma$  при  $J$  (позн.  $\Gamma \models_T \Delta$ ), якщо  $T^\wedge(\Gamma) \subseteq T^\vee(\Delta)$ .

$\Delta \in F$ -наслідком  $\Gamma$  при  $J$  (позн.  $\Gamma \models_F \Delta$ ), якщо  $F^\wedge(\Delta) \subseteq F^\vee(\Gamma)$ .

$\Delta \in TF$ -наслідком  $\Gamma$  при  $J$  (позн.  $\Gamma \models_{TF} \Delta$ ), якщо  $\Gamma \models_T \Delta$  та  $\Gamma \models_F \Delta$ .

Відповідні відношення логічного наслідку визначаємо за схемою:

$\Gamma \models_* \Delta$ , якщо  $\Gamma \models_* \Delta$  для кожної інтерпретації  $J$ .

Властивості відношень  $\models_{ID}, \models_T, \models_F, \models_{TF}$  досліджено в [7]. Відношення  $\models_{ID}$  в цій роботі позначається як  $\models_{CI}$ .

Для відношення  $\models_{ID}$  можна переносити формулу з лівої частини у праву і навпаки, накидаючи заперечення:

**Твердження 1.**  $\Phi, \Gamma \models_{ID} \Delta \Leftrightarrow \Gamma \models_{ID} \Delta, \neg\Phi$ ;  $\Gamma \models_{ID} \Delta, \Phi \Leftrightarrow \neg\Phi, \Gamma \models_{ID} \Delta$ .

Водночас (див. [7]) для відношень  $\models_T, \models_F$  та  $\models_{TF}$  таке перенесення неможливе.

Розглянемо особливості відношень  $\models_{ID}, \models_T, \models_F$ , коли одна з множин  $\Gamma$  чи  $\Delta$  – порожня.

$\emptyset \models_{ID} \Delta$  та  $\emptyset \models_F \Delta$  означають  $F^\wedge(\Delta) = \emptyset$ .

$\emptyset \models_T \Delta$  означає  $T^\vee(\Delta) = {}^V A$ ;  $\Gamma \models_F \emptyset$  означає  $F^\vee(\Gamma) = {}^V A$ .

$\Gamma \models_{ID} \emptyset$  та  $\Gamma \models_T \emptyset$  означають  $T^\wedge(\Gamma) = \emptyset$ .

Таким чином, встановлення виконуваності можна звести до встановлення відсутності логічного наслідку:

множина формул  $\Gamma$  виконувана  $\Leftrightarrow T^\wedge(\Gamma) \neq \emptyset$  для деякої  $J \Leftrightarrow \Gamma \not\models_{ID} \emptyset \Leftrightarrow \Gamma \not\models_T \emptyset$ .

Позначимо  $\neg\Delta = \{\neg\Phi \mid \Phi \in \Delta\}$ . Тоді на основі твердження 1 маємо:

**Твердження 2.**  $\Gamma \models_{ID} \Delta \Leftrightarrow \Gamma, \neg\Delta \models_{ID} \emptyset \Leftrightarrow \Gamma, \neg\Delta$  невиконувана.

Це означає, що встановлення  $\Gamma \models_{ID} \Delta$  можна звести до встановлення невиконуваності  $\Gamma, \neg\Delta$ .

Водночас для відношень  $\models_T, \models_F$  та  $\models_{TF}$  таке зведення неможливе, адже для цих відношень не можна переносити формулу з лівої частини у праву і навпаки, накидаючи заперечення.

Таким чином, метод перевірки наявності логічного наслідку для множин формул істотно універсальніший за метод перевірки виконуваності множини формул навіть для логік однозначних предикатів. Для логік тотальних чи часткових неоднозначних предикатів метод перевірки виконуваності не працює через тривіальність в таких логіках самого поняття виконуваності: для таких логік клас виконуваних формул збігається з класом усіх формул мови. Водночас секвенційні числення побудовано для різних відношень логічного наслідку в логіках як однозначних, так і тотальних чи часткових неоднозначних предикатів.

Згідно твердження 2, кожне відношення  $\Gamma \models_{ID} \Delta$  можна подати як  $\Gamma, \neg\Delta \models_{ID} \emptyset$ . Отже, властивості відношення  $\models_{ID}$  можна переписати як властивості відношення  $\models_{ID} \emptyset$ . З іншого боку, множина формул  $\Sigma$  виконувана  $\Leftrightarrow \Sigma \not\models_{ID} \emptyset$ , тому властивості виконуваності множини формул *дуальні* до властивостей відношення  $\models_{ID} \emptyset$ .

Таким чином, отримуємо наступні властивості виконуваності множини формул.

Властивості пропозиційного рівня:

MS) Нехай  $\Gamma \subseteq \Delta$ , тоді  $\Delta$  виконувана  $\Rightarrow \Gamma$  виконувана;

$\neg$ )  $\neg\neg\Phi, \Gamma$  виконувана  $\Leftrightarrow \Phi, \Gamma$  виконувана;

$\vee$ )  $\Phi \vee \Psi, \Gamma$  виконувана  $\Leftrightarrow \Phi, \Gamma$  виконувана або  $\Psi, \Gamma$  виконувана;

$\neg\vee$ )  $\neg(\Phi \vee \Psi), \Gamma$  виконувана  $\Leftrightarrow \neg\Phi, \neg\Psi, \Gamma$  виконувана.

Властивості спрощення:

R)  $R(\Phi), \Gamma$  виконувана  $\Leftrightarrow \Phi, \Gamma$  виконувана;

$\neg R$ )  $\neg R(\Phi), \Gamma$  виконувана  $\Leftrightarrow \neg\Phi, \Gamma$  виконувана;

RI)  $R_{z, \bar{x}}^{z, \bar{y}}(\Phi), \Gamma$  виконувана  $\Leftrightarrow R_{\bar{x}}^{\bar{y}}(\Phi), \Gamma$  виконувана;



$\neg RI) \neg R_{z,\bar{x}}^z(\Phi), \Gamma$  виконувана  $\Leftrightarrow \neg R_{\bar{x}}^{\bar{x}}(\Phi), \Gamma$  виконувана;

$RI) \text{ за умови } y \in v(\Phi) \text{ маємо: } R_{z,\bar{x}}^y(\Phi), \Gamma$  виконувана  $\Leftrightarrow R_{\bar{x}}^{\bar{x}}(\Phi), \Gamma$  виконувана;

$\neg RI) \text{ за умови } y \in v(\Phi) \text{ маємо: } \neg R_{z,\bar{x}}^y(\Phi), \Gamma$  виконувана  $\Leftrightarrow \neg R_{\bar{x}}^{\bar{x}}(\Phi), \Gamma$  виконувана.

Властивості еквівалентних перетворень:

$RR) R_{\bar{x}}^{\bar{x}}(R_{\bar{y}}^{\bar{y}}(\Phi)), \Gamma$  виконувана  $\Leftrightarrow R_{\bar{x}}^{\bar{x}} \circ_{\bar{y}}^{\bar{y}}(\Phi), \Gamma$  виконувана;

$\neg RR) \neg R_{\bar{x}}^{\bar{x}}(R_{\bar{y}}^{\bar{y}}(\Phi)), \Gamma$  виконувана  $\Leftrightarrow \neg R_{\bar{x}}^{\bar{x}} \circ_{\bar{y}}^{\bar{y}}(\Phi), \Gamma$  виконувана;

$R\neg) R_{\bar{x}}^{\bar{x}}(\neg\Phi), \Gamma$  виконувана  $\Leftrightarrow \neg R_{\bar{x}}^{\bar{x}}(\Phi), \Gamma$  виконувана;

$\neg R\neg) \neg R_{\bar{x}}^{\bar{x}}(\neg\Phi), \Gamma$  виконувана  $\Leftrightarrow R_{\bar{x}}^{\bar{x}}(\Phi), \Gamma$  виконувана;

$R\vee) R_{\bar{x}}^{\bar{x}}(\Phi \vee \Psi), \Gamma$  виконувана  $\Leftrightarrow R_{\bar{x}}^{\bar{x}}(\Phi) \vee R_{\bar{x}}^{\bar{x}}(\Psi), \Gamma$  виконувана;

$\neg R\vee) \neg R_{\bar{x}}^{\bar{x}}(\Phi \vee \Psi), \Gamma$  виконувана  $\Leftrightarrow \neg R_{\bar{x}}^{\bar{x}}(\Phi), \neg R_{\bar{x}}^{\bar{x}}(\Psi), \Gamma$  виконувана;

$R\exists s) \text{ за умови } y \notin \{\bar{v}, \bar{x}\} \text{ маємо: } R_{\bar{x}}^{\bar{x}}(\exists y\Phi), \Gamma$  виконувана  $\Leftrightarrow \exists y R_{\bar{x}}^{\bar{x}}(\Phi), \Gamma$  виконувана;

$\neg R\exists s) \text{ за умови } y \notin \{\bar{v}, \bar{x}\} \text{ маємо: } \neg R_{\bar{x}}^{\bar{x}}(\exists y\Phi), \Gamma$  виконувана  $\Leftrightarrow \neg \exists y R_{\bar{x}}^{\bar{x}}(\Phi), \Gamma$  виконувана;

$R\exists) \text{ за умови } z \in fu(R_{\bar{x}}^{\bar{x}}(\exists x\Phi)) \text{ маємо: } R_{\bar{x}}^{\bar{x}}(\exists x\Phi), \Gamma$  виконувана  $\Leftrightarrow \exists z R_{\bar{x}}^{\bar{x}} \circ_z^y(\Phi), \Gamma$  виконувана;

$\neg R\exists) \text{ за умови } z \in fu(R_{\bar{x}}^{\bar{x}}(\exists x\Phi)) \text{ маємо: } \neg R_{\bar{x}}^{\bar{x}}(\exists x\Phi), \Gamma$  виконувана  $\Leftrightarrow \neg \exists z R_{\bar{x}}^{\bar{x}} \circ_z^y(\Phi), \Gamma$  виконувана.

Властивості елімінації кванторів,  $\varepsilon$ -розподілу та первісного означення:

$\exists) \text{ за умови } z \in fu(\Gamma, \Delta, \exists x\Phi) \text{ маємо: } \exists x\Phi, \Gamma$  виконувана  $\Leftrightarrow R_z^x(\Phi), \neg \varepsilon z, \Gamma$  виконувана;

$\neg \exists) \neg \exists x\Phi, E_y, \Gamma$  виконувана  $\Leftrightarrow \neg \exists x\Phi, \neg R_y^x(\Phi), \neg \varepsilon y, \Gamma$  виконувана;

$\varepsilon d) \Gamma$  виконувана  $\Leftrightarrow \varepsilon y, \Gamma$  виконувана або  $\neg \varepsilon y, \Gamma$  виконувана;

$\varepsilon v) \text{ за умови } z \in fu(\Gamma) \text{ маємо: } \Gamma$  виконувана  $\Leftrightarrow \neg \varepsilon z, \Gamma$  виконувана.

Доведемо для прикладу  $\varepsilon v$ . Твердження  $\Leftarrow$  очевидне, доведемо твердження  $\Rightarrow$ .

Нехай  $\Gamma$  виконувана, тобто існують інтерпретація  $J$  та  $d \in V_A$  такі, що  $\Gamma_J(d) = T$ . Але  $z \in fu(\Gamma)$ , тому  $\Gamma_J(d \nabla z \mapsto a) = T$  для кожного  $a \in A$ . За визначенням  $\varepsilon z$  маємо  $\neg \varepsilon z(d \nabla z \mapsto a) = T$ . Отже, множина  $\neg \varepsilon z, \Gamma$  теж виконувана.

**Числення виконуваних множин.** Наведена аргументація переваг традиційних семантичних таблиць та секвенційних числень стосовно модифікованих семантичних таблиць, орієнтованих на перевірку виконуваності, ніяк не применшує важливість побудови засобів перевірки виконуваності для множин формул логік квазіарних предикатів.

Ми пропонуємо спеціальні числення секвенційного типу для перевірки виконуваності множини формул в чистих першопорядкових логіках квазіарних предикатів – числення виконуваних множин. Назвемо їх *QSF-численнями*.

Зазначені числення будують так: множина формул  $\Sigma$  невиконувана  $\Leftrightarrow \Sigma$  має виведення.

Виведення в *QSF-численнях* мають вигляд дерев, вершинами яких є множини формул. Такі дерева назвемо семантичними. Аксиомами *QSF-числень* є замкнені (невиконувані) множини формул, правилами виведення – форми декомпозиції формули.

*QSF-числення* множин в певному розумінні еквівалентні секвенційним численням для відношення  $\models_{ID}$ , названих [3] *QSC-численнями*: секвенція  $\vdash \Gamma \neg \Delta$  має виведення в *QSC-численні*  $\Leftrightarrow \Gamma \models_{ID} \Delta \Leftrightarrow \Gamma, \neg \Delta \models_{ID} \emptyset \Leftrightarrow \Gamma, \neg \Delta$  невиконувана.

Звідси отримуємо: множина формул  $\Sigma$  виконувана  $\Leftrightarrow \Sigma \not\models_{ID} \emptyset \Leftrightarrow$  секвенція  $\vdash \Sigma$  не має виведення в *QSC-численні*  $\Leftrightarrow$  існують інтерпретація  $J = (A, I)$  та  $\delta \in V_A$  такі, що  $\Phi_J(\delta) = T$  для всіх  $\Phi \in \Sigma$ . При цьому  $A$  може бути нескінченною навіть для 1-елементної множини  $\Sigma$ .

Отже, треба реформувати виведення секвенції  $\vdash \Sigma$  так, щоб усі вершини будованого дерева мали вигляд  $\vdash \Lambda$ . Це означатиме побудову в *QSF-численні* семантичного дерева для множини формул  $\Sigma$ . Тоді множина формул  $\Sigma$  виконувана  $\Leftrightarrow$  можна збудувати модель для  $\Sigma$  (скінченну чи нескінченну) на основі виведення  $\vdash \Sigma$  в *QSF-численні*.

Можливість отримання нескінченної моделі робить встановлення виконуваності множини формул неконструктивним. Водночас встановлення невиконуваності множини формул конструктивне в такому розумінні:

$\Sigma$  невиконувана  $\Leftrightarrow \vdash \Sigma$  вивідна в *QSC-численні*  $\Leftrightarrow$  маємо скінченне замкнене секвенційне дерево для  $\vdash \Sigma \Leftrightarrow$  множина  $\Sigma$  вивідна в *QSF-численні*  $\Leftrightarrow$  маємо скінченне замкнене семантичне дерево для  $\Sigma$ .

При побудові секвенційного дерева в *QSC-численні* ми перевіряємо *відсутність* логічного наслідку для множини формул кожної секвенції-вершини, тому при побудові семантичного дерева в *QSF-численні* ми для кожної вершини  $\Sigma$  будемо перевіряти  $\Sigma \not\models_{ID} \emptyset$ , тобто перевіряти *виконуваність*  $\Sigma$ . Це означає, що форми декомпозиції формули для *QSF-числень* повинні індукуватися властивостями виконуваності множини формул.

Форми декомпозиції (далі просто форми) будемо записувати у вигляді  $\frac{\Sigma}{\Omega}$  або  $\frac{\Sigma \ \Lambda}{\Omega}$ .

Множини формул над ризкою назвемо засновками, під ризкою – висновками.

Індуктивне визначення семантичного дерева:

1) множина формул  $\Sigma$  утворює тривіальне семантичне дерево з єдиною вершиною  $\Sigma$ , яка є коренем дерева;

2)  $\alpha$  – семантичне дерево з коренем  $\Sigma$ ,  $\frac{\Sigma}{\Omega}$  – форма  $\Rightarrow$   $\left| \begin{array}{c} \alpha \\ \Omega \end{array} \right.$  – семантичне дерево з коренем  $\Omega$ ;

3)  $\alpha$  та  $\beta$  – семантичні дерева з коренями  $\Sigma$  та  $Y$ ,  $\frac{\Sigma \ Y}{\Omega}$  – форма  $\Rightarrow$   $\left| \begin{array}{c} \alpha \ \beta \\ \Omega \end{array} \right.$  – семантичне дерево з коренем  $\Omega$ .

Множина формул  $X$  – наступник множини формул  $Y$  у семантичному дереві  $\delta$  з коренем  $\Sigma$ , якщо в  $\delta$  існує шлях  $\Sigma = \Sigma_1, \dots, \Sigma_n, \dots, \Sigma_m, \dots$  такий, що  $X = \Sigma_n$  та  $Y = \Sigma_m$ .

Множина формул  $\Sigma$  *вивідна* в  $QSF$ -численні, або *має виведення*, якщо існує замкнене семантичне дерево з коренем  $\Sigma$ . Таке дерево називають виведенням множини формул  $\Sigma$ .

Тривіальне семантичне дерево замкнене, якщо це замкнена множина формул.

Нетривіальне семантичне дерево замкнена, якщо кожний його лист (кінцева вершина, відмінна від кореня) – замкнена множина формул.

Опишемо умови замкненості множини формул. Спочатку наведемо необхідні визначення.

Прикладами формул  $\exists x\Phi$ ,  $\neg\exists x\Phi$  назвемо відповідно формули вигляду  $R_y^x(\Phi)$ ,  $\neg R_y^x(\Phi)$ .

Для множини формул  $\Sigma$  запис  $\Sigma_J(d) = T$  далі буде означати, що  $\Phi_J(d) = T$  для кожної  $\Phi \in \Sigma$ .

Прикладами формул  $\exists x\Phi$  чи  $\neg\exists x\Phi$  можуть бути лише формули вигляду  $R_y^x(\Phi)$  чи  $\neg R_y^x(\Phi)$ , де  $y$  – означене. Тому

при побудові виведень треба враховувати множини означених та неозначених предметних імен. Виділення означених та неозначених імен робимо за допомогою ПС вигляду  $\varepsilon x$ , які є іменами відповідних предикатів-індикаторів  $\varepsilon x$ .

Таким чином, при побудові виведення використовуємо збагачені множини, до складу яких можуть входити символи  $\varepsilon x$ . В наших численнях предикати-індикатори  $\varepsilon x$  є допоміжним інструментом побудови виведень, тому символи  $\varepsilon x$  не входять до складу інших формул, вони можуть фігурувати лише як окремі атомарні формули. Початкова множина формул, для якої будують виведення, не містить символів  $\varepsilon x$ .

Введемо для  $\Gamma \subseteq Fr$  множини *означених* та *неозначених* предметних імен, або множини *val*-змінних та *unv*-змінних:

$$val(\Gamma) = \{x \in V \mid \neg \varepsilon x \in \Gamma\}; \quad unv(\Gamma) = \{x \in V \mid \varepsilon x \in \Gamma\}.$$

Множину *нерозподілених* для  $\Gamma$  імен введемо так:  $ud(\Gamma) = nm(\Gamma) \setminus (val(\Gamma) \cup unv(\Gamma))$ .

$Rs$ -формою  $R$ -формули  $R_{x,y,z}^{\bar{x},\bar{y},\bar{z}}(\Phi)$ , де  $\{\bar{u}\} \subseteq v(\Phi)$ , назвемо  $R$ -формулу  $R_z^{\bar{y}}(\Phi)$ , утворену із  $R_{x,y,z}^{\bar{x},\bar{y},\bar{z}}(\Phi)$  спрощенням зовнішньої реномінації на основі властивостей  $RI$ ,  $RU$  та  $R$  ( $R$  може дати  $Rs$ -форму  $\Phi$ , яка не є  $R$ -формулою).

Властивості  $RI$ ,  $RU$ ,  $R$  гарантують:

якщо  $\Psi$  та  $\Theta$  мають однакові  $Rs$ -форми, то для всіх  $J$  маємо  $T(\Psi_J) = T(\Theta_J)$  та  $F(\Psi_J) = F(\Theta_J)$ .

Нехай  $Un = unv(\Gamma)$ , формула  $R_{s_1, \dots, s_k, y_1, \dots, y_n, v_1, \dots, v_m}^{r_1, \dots, r_k, x_1, \dots, x_n, u_1, \dots, u_m} \Phi$  така:  $\{r_1, \dots, r_k, s_1, \dots, s_k, y_1, \dots, y_n\} \subseteq Un$ ,  $\{x_1, \dots, x_n, v_1, \dots, v_m\} \cap Un = \emptyset$ .

Вираз  $R_{\varepsilon_1, \dots, \varepsilon_n, v_1, \dots, v_m}^{x_1, \dots, x_n, u_1, \dots, u_m} \Phi$ , де  $\varepsilon$  позначає невизначене значення, назвемо  $Un$ - $unv$ -формою формули  $R_{s_1, \dots, s_k, y_1, \dots, y_n, v_1, \dots, v_m}^{r_1, \dots, r_k, x_1, \dots, x_n, u_1, \dots, u_m} \Phi$ .

Зауважимо, що  $Un$ - $unv$ -форма  $R$ -формули  $\Psi$  може "втратити" зовнішню реномінацію (якщо отримуємо порожню реномінацію). Для формули  $\Psi$ , яка не є  $R$ -формулою, її  $Un$ - $unv$ -форма збігається із самою  $\Psi$ .

$R$ -формули  $\Psi$  та  $\Xi$  назвемо  *$Rs$ - $unv$ -еквівалентними відносно  $Un$* , або  *$Rs$ - $Un$ -еквівалентними*, якщо  $\Psi$  та  $\Xi$  мають однакові  $Rs$ -форми, або ці  $Rs$ -форми мають однакові  $Un$ - $unv$ -форми.

Якщо  $\Psi$  та  $\Xi$   $Rs$ - $Un$ -еквівалентні, то  $\neg\Psi$  та  $\neg\Xi$  теж назвемо  $Rs$ - $Un$ -еквівалентними.

Будемо вважати, що кожна  $\Phi \in Fr$   $Rs$ - $Un$ -еквівалентна сама собі (рефлексивність).

**Твердження 3.** Якщо формули  $\Psi$  та  $\Xi$   $Rs$ - $Un$ -еквівалентні, то для кожної інтерпретації  $J$  із виділеною  $Un$  маємо  $T(\Psi_J) \cap {}^{Un}A = T(\Xi_J) \cap {}^{Un}A$  та  $F(\Psi_J) \cap {}^{Un}A = F(\Xi_J) \cap {}^{Un}A$ .

Це означає, що  $\Psi_J(d) = \Xi_J(d)$  для кожних  $J$  та  $d \in {}^V A$ , для яких  $\varepsilon u(d) = T$  для всіх  $u \in Un$  (тобто  $asn(d) \cap Un = \emptyset$ ).

На основі твердження 3 доводиться

**Теорема 1.** Нехай  $\Gamma \subseteq Fr$ ,  $Un = \{x \mid \varepsilon x \in \Gamma\}$ , нехай формули  $\Psi$  та  $\Xi$  –  $Rs$ - $Un$ -еквівалентні. Тоді  $\Phi$ ,  $\neg\Psi$ ,  $\Gamma$  невиконувана та  $\neg\Phi$ ,  $\Psi$ ,  $\Gamma$  невиконувана.

Поняття замкненої множини формул вводимо так, щоб виконувалась умова:

якщо множина формул  $\Gamma$  замкнена, то  $\Gamma$  невиконувана.

**Базові форми декомпозиції та умови замкненості множини формул.**  $QSF$ -числення задається базовими формами декомпозиції та умовами замкненості (невиконуваності) множини формул.

Умова замкненості множини формул  $\Gamma$  із множиною  $unv$ -змінних  $Un$  (ця умова індукована теоремою 1):

С) існують  $Rs$ - $Un$ -еквівалентні формули  $\Psi$  та  $\Xi$  такі:  $\Psi \in \Gamma$  та  $\neg\Xi \in \Gamma$ ;

зокрема, якщо існує формула  $\Phi$  така:  $\Phi \in \Gamma$  та  $\neg\Phi \in \Gamma$ .

Наведемо базові форми декомпозиції  $QSF$ -числень.

Допоміжні форми спрощення:

$$\begin{aligned} R \frac{\Phi, \Sigma}{R(\Phi), \Sigma}; & \quad \neg R \frac{\neg\Phi, \Sigma}{\neg R(\Phi), \Sigma}; \\ RI \frac{R_x^{\bar{y}}(\Phi), \Sigma}{R_{z,x}^{\bar{z},\bar{y}}(\Phi), \Sigma}; & \quad \neg RI \frac{\neg R_x^{\bar{y}}(\Phi), \Sigma}{\neg R_{z,x}^{\bar{z},\bar{y}}(\Phi), \Sigma}; \\ RU \frac{R_y^{\bar{z}}(\Phi), \Sigma}{R_{z,y}^{\bar{z},\bar{y}}(\Phi), \Sigma}, \text{ де } y \in v(\Phi); & \quad \neg RU \frac{\neg R_y^{\bar{z}}(\Phi), \Sigma}{\neg R_{z,y}^{\bar{z},\bar{y}}(\Phi), \Sigma}, \text{ де } y \in v(\Phi). \end{aligned}$$

Основні базові форми:

$$\begin{aligned} RR \frac{R_x^{\bar{y}} \circ R_y^{\bar{z}}(\Phi), \Sigma}{R_x^{\bar{y}}(R_y^{\bar{z}}(\Phi)), \Sigma}; & \quad \neg RR \frac{\neg R_x^{\bar{y}} \circ R_y^{\bar{z}}(\Phi), \Sigma}{\neg R_x^{\bar{y}}(R_y^{\bar{z}}(\Phi)), \Sigma}; \\ R_{\neg} \frac{\neg R_x^{\bar{y}}(\Phi), \Sigma}{R_x^{\bar{y}}(\neg\Phi), \Sigma}; & \quad \neg R_{\neg} \frac{\neg \neg R_x^{\bar{y}}(\Phi), \Sigma}{\neg R_x^{\bar{y}}(\neg\Phi), \Sigma}; \end{aligned}$$

$$\begin{array}{l}
R_{\forall} \frac{R_{\bar{x}}^{\forall}(\Phi) \vee R_{\bar{x}}^{\forall}(\Psi), \Sigma}{R_{\bar{x}}^{\forall}(\Phi \vee \Psi), \Sigma}; \quad \neg R_{\forall} \frac{\neg R_{\bar{x}}^{\forall}(\Phi) \vee \neg R_{\bar{x}}^{\forall}(\Psi), \Sigma}{\neg R_{\bar{x}}^{\forall}(\Phi \vee \Psi), \Sigma}; \\
R_{\exists s} \frac{\exists y R_{\bar{x}}^{\forall}(\Phi), \Sigma}{R_{\bar{x}}^{\forall}(\exists y \Phi), \Sigma}, \text{ де } y \notin \{\bar{V}, \bar{x}\}; \quad \neg R_{\exists s} \frac{\neg \exists y R_{\bar{x}}^{\forall}(\Phi), \Sigma}{\neg R_{\bar{x}}^{\forall}(\exists y \Phi), \Sigma}, \text{ де } y \notin \{\bar{V}, \bar{x}\}; \\
\neg \neg \frac{\Phi, \Sigma}{\neg \neg \Phi, \Sigma}; \\
\vee \frac{\Phi, \Sigma \quad \Psi, \Sigma}{\Phi \vee \Psi, \Sigma}; \quad \neg \vee \frac{\neg \Phi, \neg \Psi, \Sigma}{\neg(\Phi \vee \Psi), \Sigma}; \\
R_{\exists} \frac{\exists z R_{\bar{x}}^{\forall} \circ_{\bar{z}}^{\forall}(\Phi), \Sigma}{R_{\bar{x}}^{\forall}(\exists y \Phi), \Sigma}, \text{ де } z \in fu(R_{\bar{x}}^{\forall}(\exists x \Phi)); \quad \neg R_{\exists} \frac{\neg \exists z R_{\bar{x}}^{\forall} \circ_{\bar{z}}^{\forall}(\Phi), \Sigma}{\neg R_{\bar{x}}^{\forall}(\exists y \Phi), \Sigma}, \text{ де } z \in fu(R_{\bar{x}}^{\forall}(\exists x \Phi)); \\
\exists \frac{R_z^x(\Phi), \neg \varepsilon z, \Sigma}{\exists x \Phi, \Sigma} \text{ за умови: } z \in fu(\Sigma, \exists x \Phi); \quad \neg \exists \frac{\neg \exists x \Phi, \neg R_y^x(\Phi), \neg \varepsilon y, \Sigma}{\neg \exists x \Phi, \neg \varepsilon y, \Sigma} \text{ (тут } \varepsilon y \notin \Sigma).
\end{array}$$

Форма  $\varepsilon$ -розподілу (застосовується при елімінації кванторів перед застосуванням  $\neg \exists$ ):

$$\varepsilon d \frac{\varepsilon x, \Sigma \quad \neg \varepsilon x, \Sigma}{\Sigma} \text{ за умови: } \varepsilon x \text{ не входить до } \Sigma.$$

**Теорема 2** (основна властивість форм декомпозиції). Нехай  $\frac{\Sigma}{\Delta}$  та  $\frac{X \quad Y}{\Omega}$  – базові форми. Тоді:

1)  $\Sigma$  невиконувана  $\Leftrightarrow \Delta$  невиконувана;  $X$  невиконувана та  $Y$  невиконувана  $\Leftrightarrow \Omega$  невиконувана;

2)  $\Sigma$  має модель  $\Leftrightarrow \Delta$  має модель;  $\Omega$  має модель  $\Leftrightarrow X$  має модель або  $Y$  має модель.

Твердження п.1 та п.2 еквівалентні, тому доводимо для п.2.

Для однозасновкових форм  $R, \neg R, RI, \neg RI, RU, \neg RU, RR, \neg RR, R\neg, \neg R\neg, R_{\forall}, \neg R_{\forall}, R_{\exists s}, \neg R_{\exists s}, R_{\exists}, \neg R_{\exists}, \neg \neg$  твердження теореми впливає з наступного факту. Нехай  $\Phi$  та  $\Psi$  – виділені формули засновку та висновку такої форми; тоді для кожної інтерпретації  $J$  та  $d \in {}^V A$  маємо  $\Phi_J(d) = \Psi_J(d)$ .

Форма  $\vee$ . Покажемо:  $\Phi \vee \Psi, \Sigma$  має модель  $\Leftrightarrow \Phi, \Sigma$  має модель або  $\Psi, \Sigma$  має модель.

Для довільних інтерпретації  $J$  та  $d \in {}^V A$  маємо  $\Phi \vee \Psi_J(d) = T \Leftrightarrow \Phi_J(d) = T$  або  $\Psi_J(d) = T$ . Звідси отримуємо:  $\Phi \vee \Psi, \Sigma$  має модель  $\Leftrightarrow \Phi \vee \Psi_J(d) = T$  та  $\Sigma_J(d) = T$  для деяких інтерпретації  $J$  та  $d \in {}^V A \Leftrightarrow \Phi_J(d) = T$  та  $\Sigma_J(d) = T$  для деяких  $J$  та  $d \in {}^V A$  або  $\Psi_J(d) = T$  та  $\Sigma_J(d) = T$  для деяких  $J$  та  $d \in {}^V A \Leftrightarrow \Phi, \Sigma$  має модель або  $\Psi, \Sigma$  має модель.

Форма  $\neg \vee$ . Покажемо:  $\neg(\Phi \vee \Psi), \Sigma$  має модель  $\Leftrightarrow \neg \Phi, \neg \Psi, \Sigma$  має модель.

Для довільних інтерпретації  $J$  та  $d \in {}^V A$  маємо  $\neg(\Phi \vee \Psi)_J(d) = T \Leftrightarrow \neg \Phi_J(d) = T$  та  $\neg \Psi_J(d) = T$ . Звідси отримуємо:  $\neg(\Phi \vee \Psi), \Sigma$  має модель  $\Leftrightarrow \neg(\Phi \vee \Psi)_J(d) = T$  та  $\Sigma_J(d) = T$  для деяких інтерпретації  $J$  та  $d \in {}^V A \Leftrightarrow \neg \Phi_J(d) = T, \neg \Psi_J(d) = T$  та  $\Sigma_J(d) = T$  для деяких  $J$  та  $d \in {}^V A \Leftrightarrow \neg \Phi, \neg \Psi, \Sigma$  має модель.

Форма  $\exists$ . За умови  $z \in fu(\Sigma, \exists x \Phi)$  покажемо:  $\exists x \Phi, \Sigma$  має модель  $\Leftrightarrow R_z^x(\Phi), \neg \varepsilon z, \Sigma$  має модель.

Доведемо  $\Rightarrow$ . Нехай  $\exists x \Phi, \Sigma$  має модель, тобто існують інтерпретація  $J$  та  $d \in {}^V A$  такі, що  $\exists x \Phi_J(d) = T$  та  $\Sigma_J(d) = T$ . Але  $z \in fu(\Sigma, \exists x \Phi)$ , тому для кожного  $a \in A$  маємо  $\exists x \Phi_J(d \nabla z \mapsto a) = T$  та  $\Sigma_J(d \nabla z \mapsto a) = T$ . Із  $\exists x \Phi_J(d \nabla z \mapsto a) = T$  тоді  $\Phi_J(d \nabla z \mapsto a \nabla x \mapsto b)$  для деякого  $b \in A$ , звідки  $\Phi_J(d \nabla z \mapsto a \nabla x \mapsto b) = T$ . Тоді при  $a = b$  маємо  $\Phi_J(d \nabla z \mapsto b \nabla x \mapsto b) = T$  та  $\Sigma_J(d \nabla z \mapsto b) = T$  із останнього  $R_z^x(\Phi)_J(d \nabla z \mapsto b) = T$ . Проте  $\neg \varepsilon z(d \nabla z \mapsto b) = T$ , тому  $R_z^x(\Phi), \neg \varepsilon z, \Sigma$  має модель.

Доведемо  $\Leftarrow$ . Нехай  $R_z^x(\Phi), \neg \varepsilon z, \Sigma$  має модель, тобто існують інтерпретація  $J$  та  $d \in {}^V A$  такі:  $R_z^x(\Phi)_J(d) = T, \neg \varepsilon z(d) = T, \Sigma_J(d) = T$ . Із  $\neg \varepsilon z(d) = T$  маємо  $d(z) = b$  для деякого  $b \in A$ , із  $R_z^x(\Phi)_J(d) = T$  маємо  $\Phi_J(d \nabla x \mapsto d(z)) = T$ , звідки  $\Phi_J(d \nabla x \mapsto b) = T$ , що дає  $\exists x \Phi_J(d) = T$ . Отже,  $\exists x \Phi, \Sigma$  має модель.

Форма  $\neg \exists$ . Покажемо:  $\neg \exists x \Phi, \neg \varepsilon y, \Sigma$  має модель  $\Leftrightarrow \neg \exists x \Phi, \neg R_y^x(\Phi), \neg \varepsilon y, \Sigma$  має модель.

Твердження  $\Leftarrow$  очевидне. Доведемо твердження  $\Rightarrow$ . Нехай  $\neg \exists x \Phi, \neg \varepsilon y, \Sigma$  має модель, тобто існують інтерпретація  $J$  та  $d \in {}^V A$  такі, що  $\neg \exists x \Phi_J(d) = T, \neg \varepsilon y(d) = T$  та  $\Sigma_J(d) = T$ . Із  $\neg \varepsilon y(d) = T$  отримуємо  $d(y) = b$  для деякого  $b \in A$ . Із  $\neg \exists x \Phi_J(d) = T$  отримуємо  $\exists x \Phi_J(d) = F$ , звідки  $\Phi_J(d \nabla x \mapsto a) = F$  для кожного  $a \in A$ , зокрема,  $\Phi_J(d \nabla x \mapsto b) = F$ , що дає  $\Phi_J(d \nabla x \mapsto d(y)) = F$ . Звідси  $R_y^x(\Phi)_J(d) = F$ , тому  $\neg R_y^x(\Phi)_J(d) = T$ . Таким чином,  $\neg \exists x \Phi, \neg R_y^x(\Phi), \neg \varepsilon y, \Sigma$  має модель.

Форма  $\varepsilon d$ . Покажемо:  $\Sigma$  має модель  $\Leftrightarrow \varepsilon x, \Sigma$  має модель або  $\neg \varepsilon x, \Sigma$  має модель.

Твердження  $\Leftarrow$  очевидне. Доведемо твердження  $\Rightarrow$ .

Нехай  $\Sigma$  має модель, тобто існують інтерпретація  $J$  та  $d \in {}^V A$  такі, що  $\Sigma_J(d) = T$ . Можливі два випадки.

Якщо  $x \in asn(d)$ , то  $\neg \varepsilon x(d) = T$ , звідки  $\neg \varepsilon x, \Sigma$  має модель. Якщо  $x \notin asn(d)$ , то  $\varepsilon x(d) = T$ , звідки  $\varepsilon x, \Sigma$  має модель.

**Побудова семантичного дерева. Теорема коректності.** Опишемо побудову семантичного дерева для заданої множини формул  $\Sigma$  (скінченної або зліченної). При цій побудові форми застосовуються від висновку (множина формул під ризикою) до засновків (множина формул над ризикою), тобто ми спрощуємо виділену формулу чи розкладаємо її на компоненти. Побудова дерева розбита на етапи. Вона починається з кореня – множини  $\Sigma$ . Кожне застосування форми проводиться до скінченної множини доступних на даний момент формул.

Формули вигляду  $\varepsilon x$  та  $\neg \varepsilon x$  індукуються формами елімінації кванторів та формами  $\varepsilon$ -розподілу, тому вони не можуть бути в складі початкової множини формул  $\Sigma$ .

На початку виведення множини  $\Sigma$  виконуємо первісне означення – збагачуємо  $\Sigma$  формулою  $\neg \varepsilon z$  такою, що  $z \in fu(\Sigma)$ , тобто переходимо до множини  $\neg \varepsilon z, \Delta$ . Це дає непорожність множини означених імен для кожної вершини, окрім кореня дерева, що надалі гарантує застосовність форм  $\neg \exists$ . Коректність первісного означення гарантує властивість  $\varepsilon v$ .

На початку кожного етапу для кожної незамкненої множини-листа дерева робимо крок доступу: додаємо наступну формулу до списку доступних. На початку побудови доступна лише єдина формула.

Перед кожним виконанням форми перевіряємо, чи буде відповідна множина-лист замкненою.

При появі замкненої множини формул процес побудови семантичного дерева на цьому шляху обривається.

Якщо всі листи побудованого семантичного дерева замкнені, то ми отримали замкнене семантичне дерево, побудова завершена позитивно.

Якщо ні, то у випадку побудови семантичного дерева для скінченної множини формул додатково перевіряємо, чи буде хоч один із листів фінальною множиною. Незамкнена вершина  $\Omega$  будованого дерева для  $\Sigma$  – *фінальна*, якщо до неї вже незастосовна жодна форма, або якщо кожне застосування форми до  $\Omega$  не вводить нових формул, відмінних від формул вершин на шляху від  $\Sigma$  до  $\Omega$ . Поява фінальної множини означає ситуацію повторення незамкненої множини на даному шляху. Це сигналізує про наявність в семантичному дереві шляху (від кореня до даної фінальної множини), всі вершини якого незамкнені. Такий шлях назвемо *незамкненим*. Його поява засвідчує: побудова семантичного дерева завершена негативно.

Нехай на початок етапу після додавання до листа нової доступної формули маємо множину формул  $\eta$ . Активізуємо всі доступні (окрім примітивних) формули  $\eta$ . До кожної активної формули далі застосовуємо відповідну форму. Після застосування форми утворені нею формули на даному етапі пасивні, до таких формул на цьому етапі форми не застосовуються. При появі відповідної ситуації усуваємо тотожні реномінації, тотожні пари імен та пари імен реномінацій з неістотним верхнім іменем, застосовуючи належну допоміжну форму  $R, \neg R, RI, \neg RI, RU, \neg RU$ .

Застосування форм елімінації кванторів має особливості.

Нехай  $\Xi$  – множина доступних формул на шляху від кореня до даної  $\eta$ . Нехай  $VI = val(\Xi), Ud = ud(\Xi)$ .

При застосуванні форми  $\exists$  кожен раз беремо нове тотально неістотне ім'я  $z$ .

Кожна форма  $\neg\exists$  виконується багатократно, для кожного нового  $u \in VI$ .

Якщо в момент переходу до виконання форми  $\neg\exists$  маємо  $Ud \neq \emptyset$ , то наявні *нерозподілені* імена. Тому за допомогою  $\varepsilon d$ -форми робимо всеможливі розподіли імен із  $Ud$  на означені та неозначені. Це веде до побудови піддерева висоти  $m = |Ud|$  з вершиною  $\eta$ , звідки отримуємо  $2^m$  наступників множини  $\eta$ . Далі в кожному з цих наступників виконуємо відповідну форму  $\neg\exists$ .

Після виконання кожної форми перевіряємо множину-вершину на замкненість. При появі замкненої множини до неї незастосовна жодна форма, процес побудови дерева на цьому шляху обривається.

Таким чином, при побудові семантичного дерева можливі такі випадки:

- 1) побудову завершено позитивно, маємо скінченне замкнене дерево;
- 2) побудову завершено негативно, маємо скінченне незамкнене дерево;
- 3) побудова не завершується, маємо нескінченне дерево. За лемою Кеніга [8] нескінченне дерево зі скінченим розгалуженням має хоча б один нескінченний шлях.

У випадках 2) і 3) в семантичному дереві існує незамкнений шлях  $\wp$ , всі його вершини – незамкнені множини формул. Кожна з формул початкової множини-кореня  $\Sigma$  зустрінеться на шляху  $\wp$  і стане доступною.

**Теорема 3** (коректності). Нехай множина формул  $\Sigma$  вивідна. Тоді  $\Sigma$  невиконувана.

Нехай множина формул  $\Sigma$  вивідна, тоді для  $\Sigma$  побудовано замкнене семантичне дерево. Із процесу побудови дерева випливає: кожна множина-вершина  $\wedge$  цього дерева невиконувана. Для листів семантичного дерева це впливає з визначення замкненої множини формул. Збереження формами невиконуваності множини формул (від засновків до висновку) випливає з теореми 2. Таким чином, корінь дерева – множина формул  $\Sigma$  – теж невиконувана.

**Повнота числень виконуваних множин.** Теорема повноти QSF-числень опирається на теорему про існування моделі виконуваності для множини формул незамкненого шляху. Для доведення теореми про існування моделі використовується метод модельних (хіттікківських) множин.

**Теорема 4.** Нехай  $\wp$  – незамкнений шлях у семантичному дереві, збудованій для множини формул  $\Sigma$ , нехай  $H$  – множина всіх формул цього шляху. Тоді існують інтерпретація  $J = (A, I)$  та  $\delta \in {}^VA$  такі:

$$\Phi \in H \Rightarrow \Phi_A(\delta) = T \text{ та } \neg\Phi \in H \Rightarrow \neg\Phi_A(\delta) = T.$$

Таку пару  $(A, \delta)$  назвемо  $S$ -моделлю множини формул  $\Sigma$ .

**Доведення.** Задамо  $W = val(H) = \{y \in nm(H) \mid \neg \varepsilon y \in H\}$  та  $Un = unv(H) = \{y \in nm(H) \mid \varepsilon y \in H\}$ . Якщо  $Un = \emptyset$  (тоді форми  $\neg\exists$ , тому й форми  $\varepsilon d$ , не застосовувались), то візьмемо  $W = nm(H)$ .

Усі множини формул шляху  $\wp$  незамкнені, тому для них не виконується умова замкненості  $S$ . Застосування форм до формул шляху  $\wp$  відбувається до тих пір, поки це можливо, тому кожна непримітивна формула на шляху  $\wp$  буде розкладена чи спрощена згідно з відповідною формою. Отже, для  $H$  виконується така умова коректності:

НС) не існує  $Rs$ - $Un$ -еквівалентних формул  $\Psi$  та  $\Xi$  таких, що  $\Psi \in H$  та  $\neg\Xi \in H$ ;

зокрема, не існує формули  $\Phi$  (зокрема, примітивної) такої, що  $\Phi \in H$  та  $\neg\Phi \in H$ .

Переходи від нижчої вершини шляху  $\wp$  до вищої відбуваються згідно з відповідною формою декомпозиції, тому для  $H$  виконуються такі умови переходу:

$$HR) R(\Phi) \in H \Rightarrow \Phi \in H; \neg R(\Phi) \in H \Rightarrow \neg\Phi \in H;$$

$$HRI) R_{z,\bar{x}}^z(\Phi) \in H \Rightarrow R_{\bar{x}}^z(\Phi) \in H; \neg R_{z,\bar{x}}^z(\Phi) \in H \Rightarrow \neg R_{\bar{x}}^z(\Phi) \in H;$$

$$HRU) \text{ за умови } u \in v(\Phi) \text{ маємо: } R_{z,\bar{x}}^z(\Phi) \in H \Rightarrow R_{\bar{x}}^z(\Phi) \in H; \neg R_{z,\bar{x}}^z(\Phi) \in H \Rightarrow \neg R_{\bar{x}}^z(\Phi) \in H;$$

$$HRR) R_{\bar{x}}^z(R_{\bar{y}}^w(\Phi)) \in H \Rightarrow R_{\bar{x}}^z \circ_{\bar{y}}^w(\Phi) \in H; \neg R_{\bar{x}}^z(R_{\bar{y}}^w(\Phi)) \in H \Rightarrow \neg R_{\bar{x}}^z \circ_{\bar{y}}^w(\Phi) \in H;$$

$$HR\neg) R_{\bar{x}}^z(\neg\Phi) \in H \Rightarrow \neg R_{\bar{x}}^z(\Phi) \in H; \neg R_{\bar{x}}^z(\neg\Phi) \in H \Rightarrow \neg\neg R_{\bar{x}}^z(\Phi) \in H;$$

$$HR\vee) R_{\bar{x}}^z(\Phi \vee \Psi) \in H \Rightarrow R_{\bar{x}}^z(\Phi) \vee R_{\bar{x}}^z(\Psi) \in H; \neg R_{\bar{x}}^z(\Phi \vee \Psi) \in H \Rightarrow \neg(R_{\bar{x}}^z(\Phi) \vee R_{\bar{x}}^z(\Psi)) \in H;$$

$$HR\exists) \text{ за умови } u \notin \{\bar{v}, \bar{x}\} \text{ маємо: } R_{\bar{x}}^z(\exists u\Phi) \in H \Rightarrow \exists u R_{\bar{x}}^z(\Phi) \in H; \neg R_{\bar{x}}^z(\exists u\Phi) \in H \Rightarrow \neg\exists u R_{\bar{x}}^z(\Phi) \in H;$$

$HR\exists$  за умови  $z \in fu(R_{\bar{x}}^{\bar{v}}(\exists x\Phi))$  маємо:  $R_{\bar{x}}^{\bar{v}}(\exists y\Phi) \in H \Rightarrow \exists zR_{\bar{x}}^{\bar{v}} \circ_y^z(\Phi) \in H$ ;  $\neg R_{\bar{x}}^{\bar{v}}(\exists y\Phi) \in H \Rightarrow \neg \exists zR_{\bar{x}}^{\bar{v}} \circ_y^z(\Phi) \in H$ ;

$H\neg$   $\neg \Phi \in H \Rightarrow \Phi \in H$ ;

$H\vee$   $\Phi \vee \Psi \in H \Rightarrow \Phi \in H$  або  $\Psi \in H$ ;  $\neg(\Phi \vee \Psi) \in H \Rightarrow \neg\Phi \in H$  та  $\neg\Psi \in H$ ;

$H\exists$   $\exists x\Phi \in H \Rightarrow$  існує  $u \in W$  таке, що  $R_y^x(\Phi) \in H$ ;  $\neg \exists x\Phi \in H \Rightarrow \neg R_y^x(\Phi) \in H$  для всіх  $u \in W$ ;

Множину формул  $H$ , для якої виконуються наведені умови, назвемо  $S$ -модельною.

Побудуємо  $S$ -модель за  $S$ -модельною множиною  $H$ .

Візьмемо множину  $A$  таку, що  $|A| = |W|$ , та ін'єктивну  $\delta \in {}^VA$  з  $asn(\delta) = W$ . Така  $A$  дублює  $W$ .

Для предикатів-індикаторів маємо:  $\varepsilon y \in H \Rightarrow \varepsilon y(\delta) = T$ ;  $\neg \varepsilon y \in H \Rightarrow \neg \varepsilon y(\delta) = T$ ;

Задамо значення базових предикатів та їх заперечень на  $\delta$  та на іменних множинах вигляду  $r_{\bar{x}}^{\bar{v}}(\delta)$ :

$\neg p \in H \Rightarrow p_J(\delta) = T$ ,  $\neg \neg p \in H \Rightarrow \neg p_J(\delta) = T$ ;

$\neg R_{\bar{x}}^{\bar{v}}(p) \in H \Rightarrow p_J(r_{\bar{x}}^{\bar{v}}(\delta)) = T$ ;  $\neg \neg R_{\bar{x}}^{\bar{v}}(p) \in H \Rightarrow \neg p_J(r_{\bar{x}}^{\bar{v}}(\delta)) = T$ .

Для примітивних формул та їх заперечень твердження теорема впливає з визначення базових предикатів.

Далі доводимо традиційно: індукцією за складністю формули згідно з пунктами визначення множини  $H$ .

Наведемо для прикладу доведення для пп. HRI, HRR, HR $\exists$ , H $\neg$ , H $\vee$ , H $\exists$ .

Нехай  $R_{z,\bar{x}}^{\bar{v}}(\Phi) \in H$ . За HRI  $R_{\bar{x}}^{\bar{v}}(\Phi) \in H$ . За припущенням індукції для  $\delta$   $R_{\bar{x}}^{\bar{v}}(\Phi)_J(\delta) = T$ , звідки  $R_{z,\bar{x}}^{\bar{v}}(\Phi)_J(\delta) = T$ . Нехай  $\neg R_{z,\bar{x}}^{\bar{v}}(\Phi) \in H$ . За HRI  $\neg R_{\bar{x}}^{\bar{v}}(\Phi) \in H$ . За припущенням індукції для  $\delta$  тоді  $\neg R_{\bar{x}}^{\bar{v}}(\Phi)_J(\delta) = T$ , звідки  $\neg R_{z,\bar{x}}^{\bar{v}}(\Phi)_J(\delta) = T$ .

Нехай  $R_{\bar{x}}^{\bar{v}}(R_{\bar{y}}^{\bar{w}}(\Phi)) \in H$ . За HRR тоді  $R_{\bar{x}}^{\bar{v}} \circ_{\bar{y}}^{\bar{w}}(\Phi) \in H$ . За припущенням індукції для  $\delta$  маємо  $R_{\bar{x}}^{\bar{v}} \circ_{\bar{y}}^{\bar{w}}(\Phi)_J(\delta) = T$ , звідки  $R_{\bar{x}}^{\bar{v}}(R_{\bar{y}}^{\bar{w}}(\Phi))_J(\delta) = T$ . Нехай  $\neg R_{\bar{x}}^{\bar{v}}(R_{\bar{y}}^{\bar{w}}(\Phi)) \in H$ . За HRR тоді  $\neg R_{\bar{x}}^{\bar{v}} \circ_{\bar{y}}^{\bar{w}}(\Phi) \in H$ . За припущенням індукції для  $\delta$  маємо  $\neg R_{\bar{x}}^{\bar{v}} \circ_{\bar{y}}^{\bar{w}}(\Phi)_J(\delta) = T$ , звідки  $\neg R_{\bar{x}}^{\bar{v}}(R_{\bar{y}}^{\bar{w}}(\Phi))_J(\delta) = T$ .

Нехай  $R_{\bar{x}}^{\bar{v}}(\exists y\Phi) \in H$  та  $z \in fu(R_{\bar{x}}^{\bar{v}}(\exists x\Phi))$ . За HR $\exists$  маємо  $\exists zR_{\bar{x}}^{\bar{v}} \circ_y^z(\Phi) \in H$ . За припущенням індукції для  $\delta$  тоді  $\exists zR_{\bar{x}}^{\bar{v}} \circ_y^z(\Phi)_J(\delta) = T$ , звідки  $R_{\bar{x}}^{\bar{v}}(\exists y\Phi)_J(\delta) = T$ . Нехай  $\neg R_{\bar{x}}^{\bar{v}}(\exists y\Phi) \in H$  та  $z \in fu(R_{\bar{x}}^{\bar{v}}(\exists x\Phi))$ . За HR $\exists$  тоді  $\neg \exists zR_{\bar{x}}^{\bar{v}} \circ_y^z(\Phi) \in H$ . За припущенням індукції для  $\delta$  маємо  $\neg \exists zR_{\bar{x}}^{\bar{v}} \circ_y^z(\Phi)_J(\delta) = T$ , звідки  $\neg R_{\bar{x}}^{\bar{v}}(\exists y\Phi)_J(\delta) = T$ .

Нехай  $\neg \neg \Phi \in H$ . За H $\neg$  тоді  $\Phi \in H$ . За припущенням індукції для  $\delta$  маємо  $\Phi_J(\delta) = T$ , звідки  $\neg \neg \Phi_J(\delta) = T$ .

Нехай  $\Phi \vee \Psi \in H$ . За H $\vee$  тоді  $\Phi \in H$  або  $\Psi \in H$ . За припущенням індукції для  $\delta$  маємо  $\Phi_J(\delta) = T$  або  $\Psi_J(\delta) = T$ , звідки  $\Phi \vee \Psi_J(\delta) = T$ . Нехай  $\neg(\Phi \vee \Psi) \in H$ . За H $\vee$  тоді  $\neg\Phi \in H$  та  $\neg\Psi \in H$ . За припущенням індукції для  $\delta$  маємо  $\neg\Phi_J(\delta) = T$  та  $\neg\Psi_J(\delta) = T$ , звідки  $\neg(\Phi \vee \Psi)_J(\delta) = T$ .

Нехай  $\exists x\Phi \in H$ . За H $\exists$  тоді існує  $u \in W$  таке, що  $\exists x\Phi \in H$ . За припущенням індукції для  $\delta$  маємо  $R_y^x(\Phi)_J(\delta) = T$ . Звідси  $\Phi_J(\delta \nabla x \mapsto \delta(y)) = T$ . Однак  $\delta(y) \downarrow$  згідно з  $\delta \in {}^WA$  та  $u \in W$ , тому для  $a = \delta(y)$  отримуємо  $\Phi_J(\delta \nabla x \mapsto a) = T$ , звідки  $\exists x\Phi_J(\delta) = T$ .

Нехай  $\neg \exists x\Phi \in H$ . За H $\exists$  тоді для всіх  $u \in W$  маємо  $\neg R_y^x(\Phi) \in H$ . За припущенням індукції для  $\delta$  маємо  $\neg R_y^x(\Phi)_J(\delta) = T$  для всіх  $u \in W$ . Звідси  $R_y^x(\Phi)_J(\delta) = F$  для всіх  $u \in W$ , тому  $\Phi_J(\delta \nabla x \mapsto \delta(y)) = F$  для всіх  $u \in W$ . Згідно з  $\delta \in {}^WA$  маємо  $\delta(y) \downarrow$  для всіх  $u \in W$ . Позаяк  $\delta$  є бієкцією  $W \rightarrow A$ , то кожне  $b \in A$  має вигляд  $b = \delta(y)$  для деякого  $u \in W$ . Отже,  $\Phi_J(\delta \nabla x \mapsto b) = F$  для всіх  $b \in A$ , тому  $\exists x\Phi_J(\delta) = F$ , що дає  $\neg \exists x\Phi_J(\delta) = T$ .

На основі теореми про моделі отримуємо теорему повноти для QSF-числень.

**Теорема 5** (повноти). Нехай  $\Sigma$  невиконувана. Тоді  $\Sigma$  має замкнене семантичне дерево.

*Доведення.* Припустимо супротивне:  $\Sigma$  не виконувана, проте семантичне дерево  $\delta$  для  $\Sigma$  незамкнена. Отже, в  $\delta$  існує незамкнений шлях. Нехай  $H$  – множина всіх формул цього шляху. Зрозуміло, що  $\Sigma \subseteq H$ . За теоремою 4 існує  $S$ -модель  $(J, \delta)$  така:  $\Phi \in H \Rightarrow \Phi_J(\delta) = T$  та  $\neg\Phi \in H \Rightarrow \neg\Phi_J(\delta) = T$ . Згідно з  $\Sigma \subseteq H$  тоді  $\Phi_J(\delta) = T$  для всіх  $\Phi \in \Sigma$ . Це суперечить невиконаності  $\Sigma$ .

**Висновки.** В роботі побудовано спеціальні числення секвенційного типу для перевірки виконуваності множин формул в чистих першопорядкових логіках квазіарних предикатів – числення виконуваних множин. Виведення в цих численнях мають вигляд дерев, вершинами яких є множини формул. Такі дерева названо семантичними. Аксиомами пропонованих числень є замкнені (суперечливі) множини формул, правилами виведення – форми декомпозиції.

Проаналізовано застосовність традиційних секвенційних числень та числень виконуваних множин. Числення виконуваних множин застосовні лише для логік однозначних предикатів, адже методи перевірки виконуваності не працюють в логіках неоднозначних предикатів через тривіальність в цих логіках самого поняття виконуваності. Секвенційні числення використовуємо для перевірки наявності логічного наслідку, такі числення побудовано для різних відношень логічного наслідку в логіках як однозначних, так і тотальних чи часткових неоднозначних предикатів.

Для пропонованих числень виконуваних множин наведено умови замкненості множини формул та форми декомпозиції, описано побудову виведення. Для таких числень доведено теореми коректності й повноти.

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Handbook of Logic in Computer Science. Edited by S. Abramsky, Dov M. Gabbay and T. S. E. Maibaum. – Oxford Univ. Press. – Vol. 1–5, 1993–2000.
2. Нікітченко М. С., Шкільняк С. С. Математична логіка та теорія алгоритмів. – К., 2008.
3. Шкільняк С. С. Спектр секвенційних числень першопорядкових композиційно-номінативних логік // Пробл. програмування. – 2013, № 3.
4. Шкільняк С. С. Секвенційні системи виведення першопорядкових логік часткових предикатів // Комп'ютерна математика. – 2013, Вып. 2.
5. Beth E. W. Semantic Entailment and Formal Derivability // Mededelingen der Koninklijke Nederlandse Akademie van Wetenschappen (Amsterdam). – 1955, 18, No 13.
6. Кривий С. Л. Дискретна математика. – К., 2014.
7. Нікітченко М. С., Шкільняк С. С. Прикладна логіка. – К., 2013.
8. Клини С. Математическая логика. – М., 1973.
9. Nikitchenko M., Tymofiev V. Satisfiability and Validity Problems in Many-sorted Composition-Nominative Pure Predicate Logics // Comm. in Comp. and Inf. Science. – Springer, 2012. – V. 347.

Никитченко Н. С., д-р физ.-мат. наук, проф.  
Шкільняк С. С., д-р физ.-мат. наук, проф.  
КНУ імені Тараса Шевченка, Київ

## ИСЧИСЛЕНИЯ СЕКВЕНЦИАЛЬНОГО ТИПА ДЛЯ ПРОВЕРКИ ВЫПОЛНИМОСТИ В ЛОГИКАХ КВАЗИАРНЫХ ПРЕДИКАТОВ

Построены специальные исчисления для проверки выполнимости множеств формул в чистых первогопорядковых логиках квазиарных предикатов – исчисления выполнимых множеств. Проанализированы применимость традиционных секвенциальных исчислений и исчислений выполнимых множеств. Для предложенных исчислений доказаны теоремы корректности и полноты.

**Ключевые слова:** логика, предикат, исчисление, выполнимость, корректность и полнота

Nikitchenko M.S., Dr. Sci. Phys. Math., Professor  
Shkilniak S.S., Dr. Sci. Phys. Math., Professor  
Taras Shevchenko National University of Kyiv

## SEQUENT TYPE CALCULI FOR CHECKING SATISFIABILITY IN LOGICS OF QUASI-ARY PREDICATES

Special sequent calculi to check satisfiability of sets of formulas in pure first-order logics of quasiary predicates are constructed. They are called calculi of satisfiable sets. Axioms of the proposed calculi are closed (inconsistent) sets of formulas; inferences rules are forms of decomposition. Applicability of such calculi and traditional sequent calculi are analyzed. In logics of non-deterministic predicates the traditional notion of satisfiability becomes trivial therefore calculi of satisfiable sets are used only for logics of deterministic predicates. Moreover, sequent calculi are built for different relations of logical consequence both for deterministic and total and partial non-deterministic predicates. For the proposed calculi closedness conditions of sets of formulas and forms of decomposition are described. Soundness and completeness theorems are proved for the proposed calculi.

**Key words:** logic, predicate, calculus, satisfiability, validity and completeness

УДК 519.925

В. М. Петрович, канд. техн. наук, наук. співроб.,  
Н. М. Требіна, пров. інж.-математик, К. В. Двірничук, наук. співроб.  
КНУ імені Тараса Шевченка, Київ

## ПРО ОДИН ПІДХІД ДО РОЗВ'ЯЗАННЯ ЗАДАЧІ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ОДНОВИМІРНОЇ ДИНАМІЧНОЇ СИСТЕМИ З НЕПОВНО ВИЗНАЧЕНИМ КРАЙОВИМ СТАНОМ

Сформульовано та розв'язано задачу побудови інтегрального за Лапласом зображення функції стану довільної неповно за крайовими умовами спостережуваної одновимірної розподіленої динамічної системи за умов повної інформації про її початковий стан. Виконана оцінка точності отриманого розв'язку та сформульовані умови його однозначності.

**Ключові слова:** лінійно розподілені просторово-часові системи, функції Гріна, динамічний процес.

**Вступ.** Започаткована в [1] та розвинена в [2, 3] методика математичного моделювання впливу початково-крайових збурень на стан лінійно розподіленої просторово-часової системи дозволяє [2, 3] успішно розв'язувати прямі та обернені задачі динаміки таких систем за умов неповноти інформації про їх початково-крайовий стан. Обов'язковим елементом цієї методики є наявність функції Гріна досліджуваного процесу в необмеженій просторово-часовій області. Особливості побудови [4] цієї функції вимагають повторного контурного інтегрування в комплексно визначеній області, що супроводжується певними математичними труднощами. Нижче дається використання методики [4] побудови згаданої функції для неповноспостережуваного за крайовими умовами динамічного процесу, розподіленого в одновимірній просторовій області, за наявності повної інформації про його початковий стан. Розв'язуються задачі математичного моделювання функції стану процесу за неповноти інформації про його крайові умови.

1. Розглянемо динаміку розподіленого в області  $S_0^T = \{s = (x, t) : x_0 < x < x_1, 0 \leq t \leq T\}$  процесу, функція  $y(x, t)$  стану якого визначається співвідношеннями

$$L(\partial_x, \partial_t)y(x, t) = u(x, t), \quad (1)$$

$$L_r^0(\partial_t)y(x, t)|_{t=0} = Y_r^0(x) \quad (r = \overline{1, R_0}) \quad x \in [x_0, x_1], \quad (2)$$

$$L_p^\Gamma(\partial_x)y(x, t)|_{x=X^\Gamma} = Y_p^\Gamma(x, t) \quad (p = \overline{1, R_\Gamma}) \quad x \in X^\Gamma, \quad (3)$$

в яких  $L(\cdot)$ ,  $L_r^0(\cdot)$  ( $r = \overline{1, R_0}$ ),  $L_p^\Gamma(\cdot)$  ( $p = \overline{1, R_\Gamma}$ ) – лінійні диференціальні оператори, а  $X^\Gamma = \{x_0, x_1\}$ . Розглянемо той випадок, коли кількість  $R_0$  початкових співвідношень (2) узгоджена з порядком диференціального оператора  $L(\cdot)$  за змінною  $t$ . На кількість  $R_\Gamma$  крайових співвідношень (3) особливих обмежень не накладатимемо.

Зважаючи на те, що побудова точного розв'язку сформульованої вище задачі по знаходженню функції  $y(x, t)$  за відомих  $u(x, t)$ ,  $Y_r^0(x)$  ( $r = \overline{1, R_0}$ ) та  $Y_p^\Gamma(x, t)$  ( $p = \overline{1, R_\Gamma}$ ) є проблематичною, знайдемо  $y(x, t)$ , яке б, точно задовольняючи співвідношення (1), (2), з крайовими умовами (3) узгоджувалося за середньоквадратичним критерієм незалежно від кількості останніх. А це означає, що мусять виконуватися умови

$$\sum_{r=0}^1 \sum_{p=10}^{R_\Gamma} \int_0^T \int_{x_0}^{x_1} (L_p^\Gamma(\partial_x)y(x, t)|_{x=X^\Gamma} - Y_p^\Gamma(x, t))^2 dt \rightarrow \min_{y(x, t)} \quad (4)$$

при  $x \in X^\Gamma$ .

Зауважимо, що запропонована нижче методика розв'язання задачі (1)–(4) може бути успішно використана і у випадку, коли крайові спостереження за процесом, визначені нами в точках  $x_0, x_1$ , доповнюються спостереженнями в довільних точках  $x_i \in [x_0, x_1]$ .

Задача (1)–(4) розв'язувалася нами в [2, 3, 6]. При цьому покладалося, що

$$y(x, t) = y_\infty(x, t) + y_0(x, t) + y_\Gamma(x, t),$$

де

$$y_\infty(x, t) = \int_{S_0^\Gamma} G(x - x', t - t') u(x', t') dx' dt',$$

а

$$y_0(x, t) = \int_{S^0} G(x - x', t - t') u_0(x', t') dx' dt',$$

$$y_\Gamma(x, t) = \int_{S^\Gamma} G(x - x', t - t') u_\Gamma(x', t') dx' dt'$$

або

$$y_0(x, t) = \sum_{m=1}^{M_0} G(x - x_m^0, t - t_m^0) u_{0m},$$

$$y_\Gamma(x, t) = \sum_{m=1}^{M_\Gamma} G(x - x_m^\Gamma, t - t_m^\Gamma) u_{\Gamma m} \quad (5)$$

при  $S^0 = [x_0, x_1] \times (-\infty, 0]$ ,  $S^\Gamma = ((-\infty, x_0] \cup [x_1, \infty)) \times [0, T]$ ,  $x_m^0 \in S^0$ ,  $x_m^\Gamma \in S^\Gamma$ . Методика побудови моделюючих функцій  $u_0(x, t)$ ,  $u_\Gamma(x, t)$ , їх значень  $u_{0m} = u_0(x_m^0, t_m^0)$  ( $m = \overline{1, M_0}$ ),  $u_{\Gamma m} = u_\Gamma(x_m^\Gamma, t_m^\Gamma)$  ( $m = \overline{1, M_\Gamma}$ ) та функції  $G(x - x', t - t')$  викладена нами в [2, 3, 6]. Буде показано, що функція  $y(x, t)$ , яка визначена згідно (5), задовольняє рівнянню (1) точно для довільних  $u_0(x, t)$  та  $u_\Gamma(x, t)$ .

2. Зважаючи на проблематичність [4] побудови двохаргументної функції  $G(x, t)$  та враховуючи повну визначеність розглядуваної задачі за початковим станом, побудуємо часову координату у співвідношеннях (1)–(3), переводячи їх в клас інтегральних зображень Лапласа [5] за змінною  $t$ . Для спрощення викладок обмежимося випадком, коли

$$L(\partial_x, \partial_t) = L_x(\partial_x) + L_t(\partial_t),$$

де

$$L_x(\partial_x) = \sum_{j=1}^n a_j \partial_x^j \quad (6)$$

$$L_t(\partial_t) = \sum_{j=1}^m b_j \partial_t^j$$

лінійні диференціальні оператори степені  $n$  та  $m$  відповідно, а

$$L_t^0(\partial_t) = \partial_t^{r-1}.$$

Позначивши через

$$Y(x, p) = \int_0^\infty e^{pt} y(x, t) dt,$$

$$U(x, p) = \int_0^\infty e^{pt} u(x, t) dt$$

інтегральні зображення функцій  $y(x, t)$  та  $u(x, t)$  за змінною  $t$ , а також враховуючи, що [5]

$$\partial_t^k y(x, t) = p^k Y(x, p) - \sum_{j=1}^k p^{k-j} \partial_t^{j-1} y(x, t) \Big|_{t=0},$$

на базі рівнянь (1), (2) маємо

$$(L_x(\partial_x) + L_t(p)) Y(x, p) = \bar{U}(x, p), \quad (7)$$

де

$$\bar{U}(x, p) = U(x, p) + \sum_{k=1}^m b_k \sum_{r=1}^k p^{k-r} Y_r^0(x). \quad (8)$$

Співвідношення ж (3) при цьому запишуться у вигляді

$$L_p^\Gamma(\partial_x) Y(x, p) \Big|_{x=x_i \in X^\Gamma} = Y_p(x_i, p), \quad (p = \overline{1, R_\Gamma}), \quad (9)$$

де  $Y_p(x, p)$  – інтегральне зображення функції  $Y_p^\Gamma(x, t)$ . Співвідношення (2) при цьому будуть враховані. Тому моделююча функція  $u_0(x, t)$ , а отже і складова  $y_0(x, t)$  в (5) втрачають сенс. А це означає, що початково сформульована задача (1)–(3) зведеться до розв'язання звичайного диференціального рівняння (7) з крайовими умовами (9).

3. Побудуємо розв'язок задачі (7), (9) маючи на увазі, що в загальному випадку кількість  $R_\Gamma$  крайових співвідношень (9) не узгоджена з порядком диференціального оператора  $L_x(\partial_x)$ . За цих умов знайдемо розв'язок рівняння (7) такий, щоб

$$Y(x, p) = \arg \min_{Y(x, p)} \sum_{i=0}^1 \sum_{\rho=1}^{R_\Gamma} (L_p^\Gamma(\partial_x) Y(x, p) \Big|_{x=x_i} - Y_\rho(x_i, p))^2 \tag{10}$$

при

$$Y_\rho(x, p) = \int_0^{+\infty} e^{pt} Y_\rho^\Gamma(x, t) dt.$$

Будемо виходити з того, що розв'язком (7), побудованим без врахування (9), є [2, 3] функція

$$Y_\infty(x, p) = \int_{-\infty}^{+\infty} G_x(x-x', p) \bar{U}(x', p) dx', \tag{11}$$

в якій

$$G_x(x-x', p) = \frac{1}{2\pi i} \int_{-i\infty}^{+i\infty} \frac{e^{q(x-x')}}{L_x(q) + L_t(p)} dq \tag{12}$$

(тут  $i$  – уявна одиниця). Розв'язком же задачі (7)–(10) буде функція

$$Y(x, p) = Y_\infty(x, p) + Y_\Gamma(x, p), \tag{13}$$

в якій

$$Y_\Gamma(x, p) = \int_S G(x-x', p) U_\Gamma(x', p) dx' \tag{14}$$

при  $S = (-\infty, x_0) \cup (x_1, \infty)$  та  $U_\Gamma(x, p)$  такому, щоб

$$\sum_{i=0}^1 \sum_{\rho=1}^{R_\Gamma} (L_p^\Gamma(\partial_x) Y(x, p) \Big|_{x=x_i} - Y_\rho(x_i, p))^2 \rightarrow \min_{U_\Gamma(x, p)}. \tag{15}$$

З врахуванням визначення (6) оператора  $L_x(\partial_x)$  зауважимо, що подана в (12) функція

$$G(x-x', p) = \sum_{k=1}^n \text{Res} \left[ \frac{e^{q(x-x')}}{L_x(q) + L_t(p)}, q_k \right],$$

де  $\text{Res} \left[ \frac{e^{q(x-x')}}{L_x(q) + L_t(p)}, q_k \right]$  – інтегральний лишок, який визначається [5] співвідношенням

$$\text{Res} \left[ \frac{e^{q(x-x')}}{L_x(q) + L_t(p)}, q_k \right] = \frac{e^{q_k(x-x')}}{\partial_q L_x(q) \Big|_{q=q_k}},$$

якщо  $q_k$  – простий корінь рівняння

$$L_x(q) + L_t(p) = 0 \tag{16}$$

за змінною  $q$ , або

$$\begin{aligned} \text{Res} \left[ \frac{e^{q(x-x')}}{L_x(q) + L_t(p)}, q_k \right] &= \\ &= \frac{1}{(m_k - 1)!} \lim_{q \rightarrow q_k} \left\{ \frac{d^{m_k-1}}{dq^{m_k-1}} [(q - q_k)^{m_k} \frac{e^{q(x-x')}}{L_x(q) + L_t(p)}] \right\}, \end{aligned}$$

якщо  $q_k - m_k$  – кратний корінь рівняння, уже може бути легко побудована [3] з врахуванням умов її неперервності, затухання та наявності особливості при  $x' = x$ .

Функцію  $U_\Gamma(x, p)$ , визначену згідно (15), знайдемо середньоквадратично обертаючи рівняння

$$\int_S A(x', p) U_\Gamma(x', p) dx' = \bar{Y}(p), \tag{17}$$

в якому

$$\begin{aligned} A(x', p) &= \text{col}((L_p^\Gamma(\partial_x) G_x(x-x', p) \Big|_{x=x_i}), i = \overline{0, 1}, \rho = \overline{1, R_\Gamma}), \\ \bar{Y}(p) &= \text{col}(Y_\rho(x_i, p) - L_p^\Gamma(\partial_x) Y_\infty(x, p) \Big|_{x=x_i}), i = \overline{0, 1}, \rho = \overline{1, R_\Gamma}). \end{aligned}$$

Згідно [2, 3] знаходимо, що

$$\begin{aligned} U_\Gamma(x, p) &= \arg \min_{U_\Gamma(x, p)} \left\| \int_S A(x, p) u_\Gamma(x, p) dx - \bar{Y}(p) \right\|^2 = \\ &= A^T(x, p) P_1^+(p) \bar{Y}(p) + v(x, p) - A^T(x, p) P_1^+(p) A_v(p), \end{aligned} \tag{18}$$

де знаком "+" позначена операція псевдообернення матриці

$$P_1(p) = \int_S A(x, p) A^T(x, p) dx,$$

$$A_v(p) = \int_S A(x, p) v(x, p) dx$$

за довільної інтегровної в  $S$  функції  $v(x, p)$  такої, що  $v(x, p) \equiv 0$ , якщо

$$\lim_{N \rightarrow \infty} \det[A^T(x_i, p) A(x_j, p)]_{i, j=1}^N > 0.$$



Зауважимо, що точність розв'язання задачі (7)–(10) визначатиметься точністю псевдообернення рівняння (17), або, що еквівалентно, величиною

$$\begin{aligned} \varepsilon^2(p) &= \min_{Y(x,p)} \sum_{i=0}^{R_1} \sum_{p=1}^{R_2} (L_p^\Gamma(\partial_x)Y(x,p)|_{x=x_i} - Y_p(x_i,p))^2 = \\ &= \min_{U_\Gamma(x,p)} \left\| \int_S A(x,p)U_\Gamma(x,p)dx - \bar{Y}(p) \right\|^2 = \bar{Y}^T(p)\bar{Y}(p) - \bar{Y}^T(p)P_1(p)P_1^+(p)\bar{Y}^T(p). \end{aligned} \quad (19)$$

4. Розглянемо задачу побудови функції стану  $y(x,t)$  для випадку, коли її складова  $y_\Gamma(x,t)$  співвідношенням (5) визначається через вектор

$$\bar{u}_\Gamma = \text{col}(u_{m^\Gamma}, p), m = \overline{1, M_\Gamma}$$

значень  $u_{m^\Gamma}$  моделюючої функції  $u_\Gamma(x,t)$ .

Вектор

$$\bar{U}_\Gamma(p) = \text{col}(U_{\Gamma_m}(x_{m^\Gamma}, p)), m = \overline{1, M_\Gamma}$$

значень зображення  $\bar{U}_\Gamma(x,p)$  функції  $u_\Gamma(x,t)$  в точках  $x_{m^\Gamma}$  ( $m = \overline{1, M_\Gamma}$ ), визначений згідно (4), аналогічно розглянутому вище знайдено шляхом середньоквадратичного обернення лінійного алгебраїчного рівняння

$$AU_\Gamma(p) = \bar{Y}(p),$$

в якому  $\bar{Y}(p)$  визначено вище, а

$$A = \text{colstr}((L_p^\Gamma(\partial_x)G_x(x - x_{m^\Gamma}, p)|_{x=x_i}), i = \overline{0, 1}, p = \overline{1, R_1}), m = \overline{1, M_\Gamma}. \quad (20)$$

Розв'язком же (20) таким, що

$$\|AU_\Gamma(p) - \bar{Y}(p)\|^2 \rightarrow \min_{U_\Gamma(p)},$$

буде [2, 3]

$$\bar{U}_\Gamma(p) = A^+ \bar{Y}(p) + v(p) - A^+ A v(p), \quad (21)$$

де  $A^+ = A^T(AA^T)^{-1} = (A^T A)^{-1} A^T$ , а довільна  $M_\Gamma$  – вимірна вектор-функція  $v(p) \equiv 0$ , якщо  $\det A^T A > 0$ .

Визначена в (19) точність  $\varepsilon^2(p)$  виконання умови (9) при цьому буде визначатися співвідношенням

$$\varepsilon^2(p) = \min_{U_\Gamma(p)} \|AU_\Gamma(p) - \bar{Y}(p)\|^2 = \bar{Y}^T(p)\bar{Y}(p) - \bar{Y}^T(p)A A^+ \bar{Y}(p).$$

Знайдений згідно (21) вектор  $\bar{U}_\Gamma(p)$  значень зображення  $U_\Gamma(x,p)$  моделюючої функції  $u_\Gamma(x,t)$  з врахуванням того, що

$$\sum_{m=1}^{M_\Gamma} G_x(x - x_{m^\Gamma}, p - t_{m^\Gamma}^\Gamma) \bar{U}_\Gamma(x,p) = \bar{Y}_\Gamma(x,p),$$

співвідношенням (13) визначатиметься і зображення  $Y(x,p)$  функції  $y(x,t)$  стану досліджуваного процесу.

Таким чином, сформульована вище задача (1)–(4) розв'язана нами в класі інтегральних зображень Лапласа. Інтегральне зображення функції стану  $y(x,t)$ , подане нами співвідношенням (13), буде обернене по Лапласу, якщо будуть знайдені оригінали функцій  $Y_\infty(x,p)$  та  $Y_\Gamma(x,p)$ , визначених в (11), (12), (8) та (14), (12), (18) відповідно. Зауважимо, що процедура обернення останніх не є простою, однак на нашу думку може бути реалізована для конкретно визначеного процесу (співвідношення (1)–(3)). За інших же умов задача (1)–(4), будучи некоректно сформульованою за кількістю крайових умов, є нерозв'язною.

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Стоян В. А. Об одном подходе к исследованию начально-краевых задач математической физики // Проблемы управления и информатики. – 1998. – №1. – С. 79–86.
2. Скопецкий В. В., Стоян В. А., Кривонос Ю. Г. Математичне моделювання прямих та обернених задач динаміки систем з розподіленими параметрами. – К.: Наук. думка, 2002. – 361 с.
3. Стоян В. А. Математичне моделювання лінійних, квазілінійних і нелінійних динамічних систем. К.: ВЦП "Київський університет", 2011. – 320 с.
4. Стоян В. А., Двірничук К. В. До інтегрального еквіваленту лінійних диференціальних моделей // Доповіді НАН України. – 2012. – №9. – С. 36–43.
5. Мартыненко В. С. Операционное исчисление. – К.: Изд-во Киев. ун-та, 1968. – 103 с.

Стаття надійшла до редколегії 21.10.15

Петрович В. Н., канд. техн. наук, научн. сотр.,

Требина Н. Н., вед. инж.-математик,

Двірничук К. В., научн. сотр.

КНУ імені Тараса Шевченка, Київ

## ОБ ОДНОМ ПОДХОДЕ РЕШЕНИЯ ЗАДАЧИ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ ОДНОМЕРНОЙ ДИНАМИЧЕСКОЙ СИСТЕМЫ С НЕПОЛНО ОПРЕДЕЛЕННЫМ КРАЕВЫМ ГРАНИЧНЫМ СОСТОЯНИЕМ

Сформульовано та розв'язано задачу побудови інтегрального за Лапласом зображення функції стану довільної неавтономної системи за крайовими умовами спостережуваної одновимірної розподіленої динамічної системи за умов повної інформації про її початковий стан. Виконана оцінка точності отриманого розв'язку та сформульовані умови його однозначності.

**Ключевые слова:** линейно распределенные системы, функции Грина, динамический процесс.

Petrovich V. N., Ph.D. in Engineering Science  
Trebina N. N., leading engineer-mathematician,  
Dvirnichuk K. V., researcher,  
Taras Shevchenko National University of Kyiv

## ABOUT ONE APPROACH TO PROBLEM SOLVING OF ONE DECISION OF TASK OF THE MATHEMATICAL PROGRAMMING OF UNIDIMENSIONAL DYNAMIC SYSTEM WITH INCOMPLETE CERTAIN BORDER STATE

*The task of construction of integral after Laplace image of function of the state of arbitrary is set forth and untied incompletely on the regional terms of the looked after unidimensional distributed dynamic system at the terms of complete information about her initial state. Executed estimation of exactness of the got decision and laid down condition his unambiguity.*

**Key words:** linear systems, Grine functional, dynamical process.

УДК 519.1 + 681.518.

В. Г. Скобелев, д-р фіз.-мат. наук, д-р техн. наук, проф.  
Институт кибернетики имени В. М. Глушкова НАН Украины, Киев

## ПРОБЛЕМЫ АНАЛИЗА И СИНТЕЗА КРУПНОМАСШТАБНЫХ СЕТЕЙ (ОБЗОР)

*В настоящей работе содержится обзор состояния исследований некоторых актуальных проблем анализа и синтеза крупномасштабных информационных сетей. Подробно рассмотрены наиболее часто используемые методы выделения как непересекающихся, так и пересекающихся сообществ, основанные на анализе только топологии исследуемой сети. Охарактеризованы основные подходы, модели и методы, используемые в процессе анализа социальных сетей. Выделены некоторые актуальные проблемы, возникающие в процессе проектирования крупномасштабных информационных сетей, и кратко рассмотрены существующие подходы к их решению. Охарактеризованы основные модели и методы, применяемые для обеспечения безопасности крупномасштабных информационных сетей.*

**Ключевые слова:** крупномасштабные информационные сети, анализ, синтез, безопасность, сообщества, онлайн-социальные сети.

**Введение.** В настоящее время проникновение информационных технологий во все сферы жизнедеятельности человечества достигло "критической массы", давшей старт фундаментальным изменениям в общении людей, экономике, технике, военной деятельности, науке, медицине, образовании и т. д. Эти изменения напрямую связаны с развитием сетевых информационных технологий, где одно из центральных мест принадлежит разработке крупномасштабных информационных сетей (КИС), т. е. масштабируемых информационных сетей (ИС), покрывающих большие географические регионы, и включающих в себя ИС различных типов. Сказанное подтверждается значительными усилиями и средствами, направляемыми ведущими странами для формирования глобальной информационной инфраструктуры – GII, реализуемой на основе концепции открытых систем.

По уровню архитектуры КИС можно выделить коммуникационные ИС (т. е. системы физических каналов связи, реализующие протокол передачи данных между территориально разделенными пользователями и абонентскими системами) и информационно-вычислительные ИС, предоставляющие по запросам пользователей и систем информационные и вычислительные ресурсы и услуги. К ИС второго относятся корпоративные и военные телекоммуникационные сети, ИС управления энергетикой, авиа- и ж.д.-перевозками, транспортировкой нефти и газа, квантовыми физическими экспериментами и т. д. Свои особенности в решение проблем анализа и синтеза КИС вносит также тот фактор, что объекты исследования являются сложными взаимодействующими между собой аппаратно-программными комплексами, распределенными на значительных расстояниях.

Цель настоящего обзора состоит в том, чтобы рассмотреть существующие подходы к решению некоторых проблем, связанных с анализом и синтезом КИС.

**1. Выделение сообществ в КИС.** Известно, что до сих пор отсутствует общепринятое определение понятия "сообщество в сети" (см., напр., [1–4]), а для любого из существующих определений несложно построить контр-пример. Тем не менее, общепризнано, что для большинства КИС характерны общинные структуры. На их исследование направлены значительные усилия разработчиков КИС, государственных и коммерческих структур. Чтобы охарактеризовать методы выделения сообществ в КИС, достаточно считать, что для каждой КИС  $S$  можно построить (динамическую) сеть  $U_S$ , отражающую коммуникации ее пользователей, а под сообществом в сети  $S$  понимать ту или иную подсеть сети  $U_S$  с относительно редкими подключениями и уходом частей.

Выделение сообществ в КИС сводится к решению одной из следующих двух задач: 1) выделение сообществ только на основе топологии сети; 2) выделение и анализ сообществ пользователей (в этом случае обычно КИС называют социальной сетью, а сообщества – социальными группами), объединенных в группы либо явно за счет встроенных в КИС средств образования групп, либо неявно (т. е. за счет установления связей на основе общих интересов, деятельности, кругов общения и т. д.). Вторая задача намного сложнее первой, так как социальные группы, как правило, являются сильно перекрывающимися частями сообществ, выделенных только на основе топологии сети [5, 6]. Рассмотрим методы решения этих задач.

**1.1. Выделение сообществ на основе топологии КИС.** Из-за большой сложности алгоритмического решения этой задачи (известно, что даже разбиение сети на данное число подсетей примерно равных размеров при условии минимизации числа ребер между ними является NP-трудной задачей [7, 8]) единственным приемлемым на практике способом ее решения являются эвристические методы. Охарактеризуем эти методы (сравнительный анализ эвристических методов построения сообществ в сетях и их реализаций содержится в [9–11]).

**1.1.1. Выделение непересекающихся сообществ.** При решении этой задачи часто используется схема, основанная на следующих двух предположениях: 1) сеть  $U_S$  – это обычный граф; 2) число ребер внутри любого сооб-

щества существенно больше, чем число случайным образом построенных ребер между этим сообществом и любым другим сообществом, при условии, что сохраняются степени вершин. Вычисления состоят в последовательном измельчении разбиения (иными словами, схема организована по принципу "сверху вниз") множества вершин сети  $U_S$  [12–14], и осуществляются следующим образом.

Разбиение множества вершин  $V = \{1, \dots, n\}$  сети  $U_S$  на два непересекающихся сообщества сводится к вычислению наибольшего значения квадратичной формы (модульность (modularity))  $Q = (4m)^{-1} \sum_{i,j=1}^n (a_{ij} - (2m)^{-1} k_i k_j) s_i s_j$ , где  $m$  – число ребер сети,  $a_{ij}$  ( $i, j = 1, \dots, n$ ) – элементы матрицы смежности  $A$  сети,  $k_i$  ( $i = 1, \dots, n$ ) – степень вершины  $i$ , а  $s_i \in \{-1, 1\}$  ( $i = 1, \dots, n$ ) – переменная (если  $i$  принадлежит 1-му сообществу, то  $s_i = 1$ , а если 2-му сообществу, то  $s_i = -1$ ) (содержательно, значение квадратичной формы  $Q$  равно разности между долей ребер, связывающих вершины, принадлежащие сообществу и ожидаемой такой же долей для случая, когда ребра сети распределены случайным образом, при условии, что сохраняются степени вершин). Доказано, что все значения  $Q$  принадлежат отрезку  $[-0.5, 1]$ . При этом  $Q > 0$  на тех и только на тех наборах значений переменных, которые разбивают сеть  $U_S$  на два непустых сообщества.

Вычисление наибольшего значения квадратичной формы  $Q$  осуществляется следующим образом. Матрицей модульности сети  $U_S$  называется  $n \times n$ -матрица  $B$ , элементы которой имеют вид  $b_{ij} = a_{ij} - (2m)^{-1} k_i k_j$  ( $i, j = 1, \dots, n$ ) (отметим, что в терминах этой матрицы  $Q = (4m)^{-1} \mathbf{s}^T B \mathbf{s}$ , где  $\mathbf{s} = (s_1, \dots, s_n)^T$ ). Пусть  $\beta$  – наибольшее собственное число матрицы  $B$ , а  $\mathbf{u} = (u_1, \dots, u_n)^T$  – собственный вектор, отвечающий этому собственному значению. Если  $\beta \leq 0$ , то разбиения сети  $U_S$  на два сообщества не существует. В случае, когда  $\beta > 0$ , полагаем  $s_i = 1$ , если  $u_i > 0$ , и  $s_i = -1$  иначе (если  $\mathbf{u} = (u_1, \dots, u_n)^T$  – нормализованный собственный вектор, то число  $|u_i|$  может интерпретироваться как "вес" вершины  $i$  в 1-м сообществе, если  $s_i = 1$ , и как "вес" вершины  $i$  во 2-м сообществе, если  $s_i = -1$ ).

Если произошло разбиение сети  $U_S$  на два сообщества, то описанные выше вычисления применяются к подсетям, определяемым каждым из двух блоков полученного разбиения и т. д., до тех пор, пока дальнейшее измельчение разбиения невозможно.

В результате экспериментов были выявлены следующие два недостатка базовой схемы. Во-первых, не всегда обнаруживались относительно небольшие сообщества (по-видимому, это связано с расплывчатостью предположения о том, что "число ребер внутри любого сообщества существенно больше, чем случайным образом построенное количество ребер между этим сообществом и любым другим сообществом"). Во-вторых, некоторые сообщества нельзя было выделить в принципе, так как в базовой схеме заложено предположение о том, что каждая вершина сети принадлежит единственному сообществу. В терминах теории графов это означает, что структура сети  $U_S$  представляет собой иерархию вложенных друг в друга двудольных графов. Однако эксперименты показали, что значительное число реальных КИС содержат многодольные графы, в которых длина цикла между некоторыми долями – нечетное число (именно в таких сетях встречались сообщества, которые в принципе не могла выявить базовая схема). Для выявления сообществ в таких КИС в [15] была предложена следующая мера оценки сцепления (cohesion) подмножества вершин  $X$  сети  $U_S$ , основанная на использовании треугольников:

$$c(X) = \frac{\Delta_i(X)}{\binom{n}{3}} \cdot \frac{\Delta_i(X)}{\Delta_i(X) + \Delta_o(X)},$$

где  $\binom{n}{m}$  – число сочетаний из  $n$  по  $m$ ,  $\Delta_i(X)$  – число треугольников в подсети, определяемой множеством вершин  $X$ , а  $\Delta_o(X)$  – число треугольников в сети  $U_S$ , у которых в точности две вершины принадлежат множеству  $X$ . В [16] понятие "сцепление подмножества вершин" было обобщено для взвешенных и направленных графов (отметим, что понятие "мультиграф" является специальным случаем понятия "взвешенный граф").

Высокая трудоемкость базовой схемы стимулировала разработку эвристических методов, построенных как с учетом, так и без учета значений модульности. Рассмотрим некоторые из таких методов, применяемых на практике.

1. *Жадные методы, использующие значения модульности* [17, 18]. В них осуществляется иерархическая кластеризация КИС  $U_S$  (иными словами, эти методы основаны на принципе "снизу вверх") за счет ее преобразования в последовательность мультиграфов с петлями.

Вершинами очередного мультиграфа являются сообщества, построенные на данный момент. Кратные ребра, связывающие две вершины мультиграфа, соответствуют ребрам исходной сети, связывающим сообщества, соответствующие этим вершинам. Кратная петля в вершине мультиграфа соответствует ребрам исходной сети, находящимся внутри сообщества, представленного этой вершиной. Преобразование мультиграфа осуществляется за счет объединения его  $i$ -й и  $j$ -й вершин, т. е. в замене в матрице смежности  $i$ -й и  $j$ -й строк, а также  $i$ -го и  $j$ -го столбцов их суммами. Приращение значения модульности за счет такого объединения вершин вычисляется по формуле  $\Delta Q = 2(e_{ij} - a_{ij})$ , где  $e_{ij}$  – доля ребер мультиграфа, соединяющих его  $i$ -ю вершину с  $j$ -й, а  $a_{ij} = \sum_r e_{ir}$ . На каждом шаге объединяются две вершины (склеиваются в одну вершину), для которых происходит максимальный прирост значения модульности.

Эффективность программных реализаций жадных методов достигается за счет выбора структур данных, позволяющих устранять "ненужные" операции при построении последовательности мультиграфов (например, представле-

ние матрицы смежности массивом списков смежности вершин дает возможность при объединении 2-х вершин складывать только ненулевые значения).

2. Методы, основанные на случайных блужданиях [19, 20]. В них используется то обстоятельство, что случайные блуждания по графу имеют тенденцию оказываться в "ловушке" в плотно соединенных его частях. Иерархическая кластеризация графа осуществляется применением математического аппарата конечных цепей Маркова. С этой целью определяется матрица переходов  $P = D^{-1}A$  случайного блуждания, где  $D$  –  $n \times n$ -матрица, элементы которой имеют вид:  $d_{ij} = k_{ij}$  ( $i = 1, \dots, n$ ) и  $d_{ij} = 0$  ( $i \neq j$ ). В терминах матрицы  $P$  определяется расстояние между вершинами, используемое для построения возрастающей последовательности разбиений  $\pi_1, \pi_2, \dots, \pi_n$  сети на сообщества, где  $\pi_1 = \{\{i\} | i \in V\}$ ,  $\pi_n = \{V\}$ , а  $\pi_{h+1}$  ( $h = 1, \dots, n-1$ ) получается из  $\pi_h$  за счет объединения двух сообществ. Вычисления сводятся к поиску числа  $h$  ( $1 \leq h \leq n$ ), что  $\pi_h$  – результат выделения непересекающихся сообществ в сети. В [19] такой подход реализован следующим образом. Фиксируется достаточно большое натуральное число  $t$ . Расстояние между вершинами  $i$  и  $j$  определяется формулой  $\rho_{ij} = \sqrt{\sum_{r=1}^n k_r^{-1} (p_{ir}^{(t)} - p_{jr}^{(t)})^2}$  (это обычное расстояние между вероятностными распределениями  $p_i^{(t)}$  и  $p_j^{(t)}$ ), где  $p_{ij}^{(t)}$  ( $i, j = 1, \dots, n$ ) – элемент матрицы  $P^t$  (таким образом,  $p_{ij}^{(t)}$  ( $i, j = 1, \dots, n$ ) – это вероятность перехода из вершины  $i$  в вершину  $j$  за  $t$  шагов). Так как вероятность перехода из сообщества  $C$  в вершину  $j$  равна  $\rho_{Cj}^{(t)} = |C|^{-1} \sum_{i \in C} p_{ij}^{(t)}$ , то расстояние между сообществами  $C_1$  и  $C_2$  определено формулой  $\rho_{C_1, C_2} = \sqrt{\sum_{i \in V} k_i^{-1} (p_{C_1 i}^{(t)} - p_{C_2 i}^{(t)})^2}$ . На каждом шаге кластеризации объединяются два сообщества  $C_1$  и  $C_2$ , для которых достигается минимум величины

$$\Delta(C_1, C_2) = \frac{1}{n} \cdot \frac{|C_1| \cdot |C_2|}{|C_1| + |C_2|} \cdot \rho_{C_1, C_2}^2.$$

После объединения сообществ  $C_1$  и  $C_2$  пересчитываются расстояния между сообществами, принадлежащими полученному разбиению множества вершин сети. В качестве такого числа  $h$ , что разбиение  $\pi_h$  – результат выделения непересекающихся сообществ в КИС  $U_S$ , выбирается число  $r \in \{2, \dots, n-1\}$ , которому соответствует наибольшее значение величины  $\eta_r = \frac{\sigma_{r+1} - \sigma_r}{\sigma_r - \sigma_{r-1}}$ , где  $\sigma_r = n^{-1} \sum_{C \in \pi_r, i \in C} \rho_{iC}^2$ . Эффективность программной реализации достигается за счет того, что пересчитывать необходимо только расстояния между смежными сообществами.

3. Методы, основанные на спектральной кластеризации [21, 22]. Осуществляют поиск таких попарно ортогональных собственных векторов  $x_1, \dots, x_h$  (число  $h$  заранее неизвестно), соответствующих наименьшим собственным значениям матрицы Лапласа  $L = D - A$  сети  $U_S$ , что матрица  $X$ , столбцами которой являются эти векторы, минимизирует значение целевой функции  $Tr(X^T D^{-0.5} L D^{-0.5} X)$ . Каждый собственный вектор определяет двухблочное разбиение множества вершин сети. Пересечение этих разбиений – результат разбиения КИС  $U_S$  на непересекающиеся сообщества. Сложность применения таких методов обусловлена вычислением собственных векторов для достаточно больших сетей. Выходом из такой ситуации является многократное порождение "супер-вершин", соединенных с обычными вершинами, для преобразования сети в разреженный двудольный граф.

4. Методы, основанные на имитации отжига [23–25]. Являются вероятностными процедурами, направленными на максимизацию значения выбранной целевой функции  $f(s)$  за счет последовательного преобразования состояния  $s$  сети (например,  $f$  – модульность, а  $s$  – разбиение множества вершин сети). Для этого выбирается начальная "температура"  $T_0$  (эвристический параметр) и начальное состояние  $s_0$ . Переход из одного состояния в другое осуществляется в соответствии со следующей схемой:

Шаг 1. Вычисляется текущее значение "энергии"  $E_{current} = 1 - f(s)$ .

Шаг 2. В результате воздействия на состояние  $s$  строится новое состояние  $s'$  (в стандартной реализации воздействие строится из двух видов преобразований: локальных, при которых одна вершина перемещается из одного кластера в другой, взятый наугад, кластер и глобальных, состоящих из слияний и разбиений сообществ. На практике, как правило, воздействие представляет собой комбинацию, состоящую из  $l^2$  локальных и  $l$  глобальных преобразований, где  $l$  – фиксированное число).

Шаг 3. Вычисляется новое значение "энергии"  $E_{new} = 1 - f(s')$ .

Шаг 4. Принять решение, полученное на шаге 2, с вероятностью  $P = \exp(-T^{-1}(t) \cdot \max\{0, E_{new} - E_{current}\})$  (эксперименты показали целесообразность использования схемы охлаждения  $T(t) = T_0 \cdot (1 + r \cdot t)^{-1}$ , где  $r$  – скорость охлаждения (эвристический параметр), а  $t$  – номер итерации).

Шаг 5. Если энергия системы больше заданного значения  $c$  (критерий сходимости) или температура  $T(t)$  не опустилась до предельно возможной, то увеличить номер итерации  $t$  и перейти на шаг 1, иначе – конец.

5. Методы, основанные на муравьиных колониях [26–31]. Предназначены для построения на множестве  $V$  ( $|V| = n$ ) вершин сети  $U_S$  двухблочного разбиения  $\pi = \{V_1, V_2\}$  ( $|V_1| = n_1, |V_2| = n_2$ ) ( $n_1$  и  $n_2$  – такие заданные натуральные числа, что  $n_1 + n_2 = n$ ) с как можно меньшим числом ребер между блоками. В стандартной реализации задействовано  $n$  муравьев. Каждый из них в течение  $n_1 - 1$  итераций формирует свою строго возрастающую по включению последовательность множеств  $V_{i1}(1), \dots, V_{i1}(n_1)$ , где  $i$  – номер муравья,  $|V_{i1}(1)| = 1$  и  $|V_{i1}(n_1)| = n_1$ , при-

чем  $V_{ji}(1) \cap V_{ij}(1) = \emptyset$  ( $i \neq j$ ) (известны реализации, в которых задействовано  $n!$  муравьев, при этом каждая группа из  $l$  муравьев использует одно и то же множество  $V_{ji}(1)$ ). На каждой итерации моделирование поведения муравьев осуществляется в терминах распределения ими феромона на ребрах полного графа  $K_n = (V, E)$ . Вначале на каждом ребре графа  $K_n(1)$  откладывается количество феромона, равное  $Q|E|^{-1}$ , где  $Q$  – фиксированное число. В стандартной реализации метода  $t$ -я итерация ( $t = 1, \dots, n_t - 1$ ) состоит из трех этапов.

**Этап 1** (каждый муравей строит следующий элемент своей последовательности множеств вершин). Действия  $i$ -го муравья ( $i = 1, \dots, n$ ) определяются следующей вероятностной процедурой. Для каждой вершины  $v \in \overline{V_{ji}(t)}$  ( $\overline{V_{ji}(t)} = V \setminus V_{ji}(t)$ ) вычисляется число  $a_i(t, v)$  ребер сети  $U_S$ , соединяющих вершину  $v$  с множеством  $V_{ji}(t)$ , суммарный уровень феромона  $b_i(t, v)$  на ребрах графа  $K_n(t)$ , соединяющих вершину  $v$  с множеством  $V_{ji}(t)$ , стоимость  $f_i(t, v)$  связей вершины  $v$  с множеством  $V_{ji}(t)$  (стоимость  $f_i(t, v)$  представляется либо в мультипликативной форме  $f_i(t, v) = (b_i(t, v))^\alpha (a_i(t, v) + 1)^\beta$ , либо в аддитивной форме  $f_i(t, v) = (b_i(t, v))^\alpha + (a_i(t, v))^\beta$ , где неотрицательные параметры  $\alpha$  и  $\beta$  подбираются экспериментально. Если  $\alpha = 0$ , то каждый муравей реализует жадный метод. Если  $\beta = 0$ , то каждый муравей действует только на основании феромона, что, как правило, приводит к решениям, которые не являются оптимальными) и вероятность  $P_i(t, v) = f_i(t, v) \cdot (\sum_{v \in \overline{V_{ji}(t)}} f_i(t, v))^{-1}$  включения вершины  $v$  в множество  $V_{ji}(t+1)$ . Случайным образом выбирается одна вершина  $v \in \overline{V_{ji}(t)}$ , для которой эта вероятность – наибольшая, и полагается  $V_{ji}(t+1) := V_{ji}(t) \cup \{v\}$ .

**Этап 2** (муравьи откладывают феромон на ребрах графа  $K_n(t)$ ). Количество феромона, откладываемого  $i$ -м муравьем ( $i = 1, \dots, n$ ) на каждом ребре подграфа графа  $K_n(t)$ , определенного множеством ребер  $V_{ji}(t+1)$ , равно  $\Delta_i(t) = Q \cdot (c_i(t))^{-1}$ , где  $c_i(t)$  – число ребер сети  $U_S$ , соединяющих множества вершин  $V_{ji}(t+1)$  и  $\overline{V_{ji}(t+1)}$ .

**Этап 3** (испарение феромона). На каждом ребре графа  $K_n(t)$ , построенного на этапе 2, количество феромона пересчитывается в соответствии с формулой  $d := d \cdot (1 - \rho)$ , где  $\rho$  – подбираемый экспериментально коэффициент обновления. В результате построен граф  $K_n(t+1)$ .

По окончании описанной выше итеративной процедуры среди всех разбиений  $\{V_{ji}(n_t), \overline{V_{ji}(n_t)}\}$  ( $i = 1, \dots, n$ ) выбирается разбиение с наименьшим числом ребер между блоками.

### 1.1.2. Выделение пересекающихся сообществ.

1. **Методы, основанные на перколяции клик (CPM)** [32–35]. В них осуществляется поиск всех максимальных клик в КИС  $U_S$ . После этого объединяются клики, имеющие общие вершины и удовлетворяющие заданному метрическому или топологическому критерию. Недостатком этих методов является экспоненциальная сложность от параметров КИС  $U_S$ .

2. **Методы, основанные на распространении меток (SLPA)** [36–38]. Первоначально каждой вершине КИС  $U_S$  присваивается уникальная метка. Далее начинается итерационный процесс (количество итераций определяется пользователем), в котором каждая вершина принимает метку (или некоторое множество меток) по согласованию с большинством ее соседей. По окончании этого процесса множества вершин с одной и той же меткой, определяющие связанные подграфы, объявляются сообществами. Достоинство SLPA – простота реализации и быстрое исполнение (по этой причине SLPA часто используют в качестве составляющей части при построении различных методов выделения структур сообществ в реальных КИС [39–44]). Основным недостатком SLPA – различные их реализации приводят к различным решениям, некоторые из которых имеют очень плохое качество (это вызвано тем, что методы SLPA ориентированы на достижение некоторого локального минимума). В [45] показано, что SLPA эквивалентны применению упрощенной модели Поттса (модель Поттса используется в статистической механике для описания взаимодействия спинов в кристаллической решетке) для выделения структур сообществ в сети [46].

3. **Методы, основанные на хешировании, чувствительном к местоположению (LSH)** [47–51]. Основаны на уменьшении размерности данных за счет такого подбора хэш-функций, чтобы похожие объекты с высокой вероятностью попадали в одну и ту же корзину. Вычисления состоят в следующем. Заменяем КИС  $U_S$  взвешенным направленным графом, где  $w(v_j, v_i) \in [0, 1]$  ( $v_j, v_i \in V; v_j \neq v_i$ ) – вес того, что дуга идет из вершины  $v_j$  в вершину  $v_i$  ( $\sum_{v_i \in V \setminus \{v_j\}} w(v_j, v_i) = 1$  для всех  $v_j \in V$ ). Далее многократно применяется следующая схема.

Генерируем  $n$  случайных перестановок  $\pi_1, \dots, \pi_n$  множества  $V$ . Для каждой вершины  $v_j \in V$  вычислим  $n$  значений  $mh_1, \dots, mh_n$ , где  $mh_i$  ( $i = 1, \dots, n$ ) – это элемент множества  $N_j = \{v_j\} \cup \{v_i \mid w(v_j, v_i) > 0\}$ , с наименьшим индексом в перестановке  $\pi_i$ . Выберем случайным образом  $r$  ( $r < n$ ) попарно различных чисел  $l_1, \dots, l_r \in \mathbf{N}_n$ . Сопоставим с каждой вершиной сигнатуру, являющуюся конкатенацией значений  $mh_{l_1}, \dots, mh_{l_r}$ . Хешируем вершины на основе их сигнатур. Вершины с одним и тем же хэш-значением помещаются в одно сообщество (вероятность того, что вершины

$v_j$  и  $v_i$  согласованы на этой сигнатуре, равна  $\left( \frac{|N_j \cap N_i|}{|N_j \cup N_i|} \right)^r$ , так как вероятность того, что  $v_j$  и  $v_i$  согласованы по одному значению (т. е. индекс Жаккарда) равна  $\frac{|N_j \cap N_i|}{|N_j \cup N_i|}$ ).

4. Методы, использующие функцию локальной плотности (функция локальной плотности представляет собой один из вариантов функции полезности) [52–55]. Их суть состоит в следующем. В соответствии с выбранным критерием каждой вершине сети приписывается ранг. Последовательным удалением вершин (в направлении от максимального ранга к минимальному рангу) формируются попарно непересекающиеся ядра предполагаемых сообществ. Каждое ядро последовательно преобразуется за счет добавления или удаления вершин, обеспечивающего не убывания значения функции локальной плотности. Эта процедура выполняется до тех пор, пока невозможно осуществить дальнейшее

увеличение значения этой функции. В [52] использовалась функция локальной плотности  $f(C) = \frac{w_{in}^{(C)}}{w_{in}^{(C)} + w_{out}^{(C)}}$ , где  $w_{in}^{(C)}$  и  $w_{out}^{(C)}$ , соответственно, внутренний и внешний вес сообщества  $C$  (в частности, в качестве внутреннего веса  $w_{in}^{(C)}$  и внешнего веса  $w_{out}^{(C)}$  могут быть выбраны суммы, соответственно, внутренних и внешних степеней вершин, принадлежащих сообществу  $C$ ).

Ее недостаток в том, что могут порождаться сообщества, являющиеся несвязными подграфами (так как в процессе построения сообщества допускается удаление вершин). Для его устранения в [53] было предложено использовать функцию  $f(C) = \frac{w_{in}^{(C)}}{w_{in}^{(C)} + w_{out}^{(C)}} + \lambda e_p$ , где  $e_p = \frac{2w_{in}^{(C)}}{|C|(|C|-1)}$  – реберная вероятность (реберная вероятность отражает уровень связности в сообществе  $C$ ), а  $\lambda$  – параметр, и при построении сообщества сохранять только одну компоненту, имеющую максимальную плотность.

В [54] использовалась функция  $f(C) = \frac{k_{in}^{(C)}}{(k_{in}^{(C)} + k_{out}^{(C)})^\alpha}$ , где  $\alpha$  – параметр, управляющий размером сообществ. При этом после нахождения очередного сообщества случайным образом выбиралась вершина, не принадлежащая ни одному из построенных сообществ, которая использовалась для построения нового сообщества.

Предложенная в [55] функция  $f(C) = \frac{k_{in}^{(C)} + 1}{(k_{in}^{(C)} + k_{out}^{(C)})^\alpha}$  дает возможность рассматривать одновершинные сообщества, а также ускорить процесс объединения сообществ при их построении.

1.1.3. **Выделение сообществ на основе статистического вывода.** В этих методах используют стохастические блочные модели (SBM) (эти модели являются Байесовскими моделями для графов [56, 57]) в предположении, что известно число  $l$  сообществ КИС  $U_S$  (основная предпосылка этой модели состоит в том, что вершины, принадлежащие одному и тому же сообществу стохастически эквивалентны, т. е. для всех вершин в сообществе одинаковы вероятности их связей со всеми другими вершинами). Основная задача – максимизировать значение функции правдоподобия. Рассмотрим основные SBM.

SBM Бернулли применяется для выделения непересекающихся сообществ. В этом случае осуществляется вывод вектора  $\alpha = (\alpha_1, \dots, \alpha_n)$  ( $\alpha_i \in \{1, \dots, l\}$  ( $i \in V$ ) – номер сообщества, содержащего вершину  $i$ ) на основе  $l \times l$ -матрицы  $P$ , где  $p_{ij}$  ( $i, j \in \{1, \dots, l\}$ ) – вероятность связи между сообществами  $i$  и  $j$ . Функция правдоподобия имеет вид  $L(U_S | \alpha, P) = \prod_{i < j} p_{\alpha_i \alpha_j}^{a_{ij}} (1 - p_{\alpha_i \alpha_j})^{1 - a_{ij}}$  (таким образом, предполагается, что каждое ребро генерируется независимо, а величины  $a_{ij}$  распределены по закону Бернулли, где  $P(a_{ij} = 1) = p_{\alpha_i \alpha_j}$ ).

SBM Пуассона (эту модель часто рассматривают как базовую SBM) применяется для выделения непересекающихся сообществ. Предполагается, что величины  $a_{ij}$  распределены по закону Пуассона, т. е.  $a_{ij} \sim \text{Poi}(w_{\alpha_i \alpha_j})$  (таким образом,  $l \times l$ -матрица  $P$  заменяется  $l \times l$ -матрицей  $\Omega$  [58]). Для любых сообществ  $r$  и  $s$  оценка максимального правдоподобия величины  $w_{rs}$  имеет вид  $\hat{w}_{rs} = \frac{m_{rs}}{n_r n_s}$ , где  $m_{rs}$  – это число ребер между сообществами  $r$  и  $s$ , если  $r \neq s$ , и удвоенное число ребер внутри сообщества  $r$ , если  $r = s$ , а  $n_r$  – число вершин в сообществе  $r$ . Функция лог-правдоподобия имеет вид  $\log L(U_S | \alpha) = 0.5 \sum_{r,s} m_{rs} \log \frac{m_{rs}}{n_r n_s}$  [59].

В SBM Пуассона вершины сети, степени которых существенно отличаются друг от друга, имеют мало шансов попасть в одно и то же сообщество. Поэтому эта модель не дает возможность адекватно выделять сообщества в сетях, для которых характерна большая вариация распределения степеней вершин в сообществах. Был разработан ряд SBM, допускающих неоднородности степеней вершин в пределах сообщества. Наибольшее распространение получила DC SBM [58]. Для нее функция лог-правдоподобия имеет вид  $\log L(U_S | \alpha) = 0.5 \sum_{r,s} m_{rs} \log \frac{m_{rs}}{k_r k_s}$ , где  $k_r$  – сумма степеней вершин в сообществе  $r$ .

Существуют SBM предназначенные для выделения пересекающихся сообществ в сетях. Наиболее известной из них является MMSB [56]. В этой модели вычисления осуществляются следующим образом. Вначале с использованием распределения Дирихле каждой вершине  $v$  ставится в соответствие вектор длины  $l$ , определяющий распределение этой вершины по сообществам. Далее иницируется итерационный процесс стохастической оптимизации (число итераций определяется пользователем), преобразующий структуру сообществ в сети на основе вариационного вывода методом самосогласованного поля для приближенного вычисления апостериорных распределений принадлежности вершин сообществам.

1.1.4. **Нечеткие методы выделения сообществ.** В них первоначально для каждой вершины рассчитывается "мягкий" вектор ее принадлежности (этот вектор является стохастическим вектором, т. е. его компоненты неотрицательны и их сумма равна единице [62]) сообществам [60–62]. При этом естественно возникает проблема определения

размерности  $l$  этого вектора (это значение  $l$  может быть выбрано либо в виде параметра метода, либо определено на основе анализа данных), что существенно ограничивает успешное применение нечетких методов.

В [63] выделение пересекающихся сообществ сети сводится к решению нелинейной оптимизационной задачи (с ограничениями). Для этого предполагается, что задана непрерывная дифференцируемая функция подобия (similarity function) вершин исследуемой сети, а для решения задачи используется градиентный метод, т. е. вычисления заведомо направлены на поиск локального минимума.

Несмотря на то, что разработке нечетких методов выделения сообществ в сетях в последнее время уделяется значительное внимание, основной вопрос, на который пока нет удовлетворительного ответа, состоит в следующем: как сравнивать между собой четкие и нечеткие методы выделения сообществ в сетях.

**1.2. Анализ социальных сетей.** Проблема выделения сообществ пользователей в КИС значительно сложнее и многограннее, чем выделение сообществ на основе топологии КИС. Это обусловлено следующими причинами.

Понятие социальной сети как общественного явления сформировалось в 1-й половине XX века, а элементы сетевого анализа впервые были использованы именно в классической социологии в середине этого же века (благодаря этим исследованиям под социальной сетью стали понимать "набор социально-релевантных узлов, связанных одним или несколькими отношениями", что дало возможность основным социологическим понятиям "актор" и "взаимодействие" поставить в соответствие термин теории графов, а именно: вершину и ребро. Операционализация основных социологических понятий в терминах теории графов и использование широкого спектра математических методов привело к созданию методологии исследования социальных сетей, обладающей мощными инструментами анализа данных и визуализации результатов).

Работы в области формальной социологии привели к пониманию того, что представление объекта исследования в сетевом виде требует структурирования описывающих его переменных. Эта проблема была исследована в [64]. Были выделены следующие четыре основных типа свойств (иными словами, характеристик) члена коллектива (актора): 1) *абсолютные*, т. е. полученные без учета свойств коллектива или взаимоотношений в нем; 2) *относительные*, т. е. установленные путем обработки информации об отношениях между данным членом коллектива и другими его членами; 3) *сравнительные*, т. е. полученные при сопоставлении характеристики члена коллектива с ее средним значением по группе; 4) *контекстуальные*, т. е. выведенные из свойств самого коллектива. Были выделены следующие типы свойств (характеристик) группы (коллектива): 1) *аналитические*, т. е. основанные на данных о каждом члене группы; 2) *структурные*, т. е. основанные на сведениях об отношениях между членами группы; 3) *глобальные*, т. е. основанные на коллективных проявлениях.

Формирование формальной социологии в значительной мере осуществлялось в рамках общей теории идентификации [65]. Именно с этих позиций были выявлены следующие основные моменты. Во-первых, люди присоединяются к группам исходя из их идентичности (т. е. на основании интереса к обсуждаемым темам) или из социальных отношений (bond attachment). Во-вторых, поддержание кругов общения в явном виде громоздко и занимает слишком много времени. В-третьих, для многих людей по разным причинам неудобно явно объявлять свое членство в тех или иных группах. В-четвертых, некоторые типы групп (географические, образовательные, профессиональные и т. д.) основаны на сходстве атрибутов и не могут быть полностью восстановлены из-за неполноты информации об участниках. В-пятых, построение структуры неявных сообществ сети в явном виде является ценной информацией для различных аналитических центров и служб (именно это обстоятельство, во многом, стимулировало разработку методов кластеризации графов). В-шестых, одной из основных характеристик отношений между узлами социальной сети является "сила связи", определяемая частотой, интенсивностью, регулярностью и устойчивостью связей (поэтому можно выделить сильные, слабые и латентные связи).

На этом фоне в 90-х годах XX века и появились первые онлайн-сервисы социальных сетей (Facebook, Twitter, YouTube и т. д.). Практика показала, что методы анализа социальных сетей (методы анализа социальных сетей используются для исследования взаимодействий между участниками сети, прогнозирования их поведения, классификации, моделирования информационных потоков в сетях) могут успешно применяться при анализе онлайн-взаимодействий [66–69], что обусловлено следующими обстоятельствами. Онлайн-взаимодействия всегда имеют сетевую структуру (узлами сети могут быть люди, организации, веб-страницы, публикации, страны и т. д., а связями – потоки информации, различные виды ресурсов, взаимодействие, социальные отношения и т. д. Таким образом, классификации участников сети, анализ их взаимодействий и прогнозирование их поведения играют важную роль при решении исследовательских и бизнес-задач, для борьбы с сетевыми мошенничествами, отмыванием денег, кибератаками, а также для создания вспомогательных сервисов и приложений для пользователей социальных сетей), так как электронная коммуникация – сеть, состоящая из отправителей и получателей.

Представления о коммуникационных системах и отношениях были сформированы задолго до возникновения Интернета. Однако до сих пор сбор личных данных – сложная и громоздкая процедура, занимающая много времени. Кроме того, часто люди сами не понимают, кто находится в их личной сети или насколько сильна та или иная связь, а исследователю необходимо собрать точные данные о взаимодействиях. Эти сложности могут быть снижены за счёт онлайн-исследований, так как в них информация является цифровой и закодированной через акт отправки сообщения или добавления друга с помощью функционала Интернет-страницы. В электронных социальных сетях также несложно копировать сообщения для их дальнейшего анализа. Отметим, что источниками данных о структуре социальной сети могут служить API (application programming interface – интерфейс программирования приложений) популярных социальных сетей (Facebook, Twitter, LinkedIn), такие универсальные децентрализованные семантические сети, как Giant Global Graph на RDF2 протоколах, электронные данные, имеющиеся у государства, а также закрытые электронные данные банков и частных компаний.

Под онлайн-социальной сетью понимают Интернет-сервис, позволяющий пользователям публиковать на своих страницах ту или иную информацию, и служащий для упрощения коммуникации и обмена информацией между пользователями сети Интернет. Выделяют следующие типы онлайн-социальных сетей: 1) общего назначения (ВКонтакте, Мой Мир, Одноклассники, Facebook, и т. д.), предназначенные для поддержания существующих контактов из реального мира, обсуждения событий и развлечений; 2) контентные (Twitter, YouTube, и т. д.), предназначенные для обмена контентом, распространения новостей, создания и развития сообществ по интересам и раз-

влечений; 3) тематические (LinkedIn, Academia.edu, ResearchGate и т. д.), предназначенные для поддержания существующих и установления новых профессиональных контактов, дискуссий по общим интересам и решения профессиональных вопросов; 4) остальные (FourSquare – сеть с функцией геопозиционирования, Instagram – сеть с упором на выкладывание пользователем фотографий и т. д.).

Несмотря на многообразие онлайн-социальных сетей, в них естественно выделяются следующие типы данных: 1) сетевые данные, т. е. отношения связи между пользователями, а также между пользователями и объектами; 2) профили пользователей, т. е. их CV, биографии, взгляды, интересы и т. д.; 3) текстовые данные, т. е. сообщения, комментарии и т. д.; 4) мультимедийные данные, т. е. фото-, аудио- и видео-материалы; 5) объекты уровня сети, т. е. группы, сообщества, приложения и т. д.; 6) объекты внешнего мира, т. е. ссылки на ресурсы, расположенные за пределами сети; 7) журналы активности пользователей, т. е. записи о взаимодействии пользователей между собой и с различными объектами. Таким образом, онлайн-социальная сеть представляет собой граф (часто для онлайн-социальных сетей адекватным является представление размеченным ориентированным, либо направленным мультиграфом с петлями [70]). Некоторые онлайн-социальные сети естественно представлять гиперграфами (например, взаимодействие типа "назначение тега фотографии" включает пользователя, фотографию и тег), у которого вершины и ребра снабжены некоторыми наборами атрибутов.

Выделяют следующие четыре подхода к анализу онлайн-социальных сетей [71, 72]:

1. *Структурный подход.* Участники онлайн-социальной сети представляются вершинами графа, влияющими на конфигурацию ребер и других участников сети. Исследуется геометрическая форма сети и интенсивность взаимодействий (вес ребер), т. е. такие характеристики, как взаимное расположение вершин, центральность и транзитивность взаимодействий. Для интерпретации результатов применяются структурные теории и теории сетевого обмена.

2. *Ресурсный подход.* Исследуются возможности участников онлайн-социальной сети по привлечению индивидуальных и сетевых ресурсов (*индивидуальными ресурсами могут быть знания, престиж, богатство, раса, пол и т. д., а сетевыми ресурсами – влияние, статус, информация, капитал и т. д.*) для достижения поставленных целей. Осуществляется дифференциация участников, находящихся в идентичных структурных позициях социальной сети, в соответствии с их ресурсами.

3. *Нормативный подход.* Исследуется уровень доверия между участниками онлайн-социальной сети, нормы, правила и санкции, влияющие на их поведение и процессы их взаимодействий. Анализируются социальные роли, связанные с данным ребром сети (*например, отношения руководителя и подчиненного, дружеские или родственные связи и т. д.*). Комбинация индивидуальных и сетевых ресурсов участника с нормами и правилами, действующими в данной социальной сети, образует его "сетевой капитал", т. е. преимущества, которые участник может получить в произвольный момент времени для достижения поставленной цели.

4. *Динамический подход.* Исследуются изменения в структуре онлайн-социальной сети во времени, т. е. влияние внешних и внутренних воздействий на структуру сети, выделение тех или иных стационарных конфигураций в сети и т. д.

В [72, 73] эти подходы охарактеризованы в терминах следующих актуальных для анализа онлайн-социальных сетей задач и методов их решения.

1. *Статистический анализ* (*решение этой задачи для обычных социальных сетей было получено в 70-е годы XX столетия (см., напр., [74])*). На основе использования статистических методов исследуется изменение во времени основных характеристик типичных онлайн-социальных сетей, т. е. структуры, поведения и т. д. Обзор аналитических моделей, используемых в процессе событийного моделирования онлайн-социальных сетей содержится в [75].

2. *Выделение сообществ.* Модели и методы, рассмотренные в п.1.1, адаптируются для онлайн-социальных сетей с учетом следующих обстоятельств:

а) вершина может принадлежать нескольким сообществам с разной степенью принадлежности, т. е. осуществляется поиск нечетких кластеров (fuzzy clusters) [33, 60, 76, 77];

б) структура сообществ может быть иерархической [54, 78–80];

в) в предположении истинности гипотезы случайного распределения ребер при заданных значениях степеней вершин необходимо отсекают "псевдосообщества" на основе статистической значимости конфигурации [77, 81].

3. *Классификация вершин.* В онлайн-социальной сети вершины, у которых выделенные атрибуты удовлетворяют заданным условиям, снабжаются метками, которые в соответствии с тем или иным отношением толерантности (и, возможно, при наличии дополнительных ограничений) распространяются по сети (*такой подход используется в маркетинговых исследованиях заинтересованности участников сети в конкретном продукте [82, 83], в экспертных методах решения крупных проблем для выявления экспертов в группах специалистов в конкретных узких областях, а также при анализе текстовой [84, 85] и мультимедийной [86] информации в сети*).

4. *Анализ социального влияния.* Исследуется распространение по онлайн-социальной сети той или иной информации, влияющей на действия ее пользователей [87–90]. Для решения этой задачи чаще всего используются методы, получившие имя "вирусный маркетинг". Их суть состоит в том, чтобы главным распространителем информации являлись ее адресаты (*это достигается путем формирования содержания, способного привлечь новых получателей информации за счет яркой, творческой, необычной идеи или с использованием естественного или доверительного послания*). В результате информация распространяется по сети по закону, близкому к геометрической прогрессии. Следует отметить, что в последнее время у государственных органов различных стран вызывает большую озабоченность усовершенствованная разновидность вирусного маркетинга, получившая имя "террористический маркетинг". Такие методы используют современные медийные средства для того, чтобы достаточно быстро и эффективно осуществить подрывную деятельность небольшой инициативной группой лиц, внедренной в онлайн-социальную сеть для воздействия на целевую аудиторию.

5. *Визуализация социальных сетей.* Предназначена для наглядного представления структуры больших онлайн-социальных сетей в статическом и динамическом режимах [91]. Основная сложность состоит в разработке алгоритмов, сочетающих методы анализа сети и ее визуализации, обеспечивающих достижение поставленной цели. В настоящее время для визуализации и анализа данных в онлайн-социальных сетях создан ряд инстру-



ментальных средств. Одной из наиболее известных таких систем является SWIFT-3D [92], созданная в AT&T Labs. Эта система объединяет набор интерактивных инструментов, который включает пиксель-ориентированные 2D карты, интерактивные 3D-карты, статистические отображения, диаграммы сетевой топологии и интерактивный интерфейс запросов, организованный по принципу сверху-вниз.

Отметим, что при исследовании реальных онлайн-сетей приходится решать рассмотренные выше задачи в их совокупности. Поэтому дополнительно возникает проблема согласования между собой используемых моделей, структур данных и методов решения этих задач.

**2. Проектирование КИС.** Наряду с решением обычных проблем, возникающих при проектировании любой ИТ-системы (*обзор этих проблем представлен, например, в [93–95]*), возникают следующие проблемы, вызванные именно наличием сетевой структуры в КИС [96].

**2.1. Формулировка требований к проектируемой КИС.** Хотя требования существенно зависят от назначения КИС (*например, для КИС, используемых в цифровой экономике, типичны следующие требования: 1) сеть должна быть в рабочем состоянии все время, даже в случае несостоявшихся соединений, отказов оборудования и при условиях ее перегрузки; 2) сеть должна надежно доставлять приложения и обеспечить приемлемые сроки реагирования между любыми хостами; 3) сеть должна быть безопасной, т. е. должны быть защищены данные, которые передаются по сети и данные, хранящиеся на устройствах, которые подключаются к сети; 4) сеть должна легко адаптироваться к ее росту и общим изменениям бизнеса; 5) так как время от времени возникают неисправности в сети, то их поиск, локализация и устранение не должны занимать слишком много времени*), они направлены на достижение следующих четырех целей: 1) масштабируемость, т. е. возможность роста сети за счет подключения новых групп пользователей и удаленных объектов, поддержка новых приложений без снижения уровня обслуживания существующих пользователей; 2) доступность, т. е. обеспечение стабильного и надежного непрерывного функционирования, причем отказ одного звена или оборудования не должен существенно влиять на производительность сети; 3) безопасность, т. е. планирование местоположения устройств безопасности, фильтров, функций межсетевых экранов, а также формулирование требований к криптографической защите информации и системе доступа к сети; 4) управляемость, т. е. имеющийся в наличии персонал должен быть в состоянии управлять сетью и поддерживать ее (*сеть, которую слишком сложно поддерживать, не может функционировать эффективно и результативно*).

**2.2. Выбор структурной организации проектируемой КИС.** Современные КИС имеют иерархическую структурную организацию. Так как увеличение количества уровней упрощает процесс построения сети, но ухудшает временные характеристики, то целесообразно уменьшать количество уровней при одновременном использовании виртуальных каналов связи.

В настоящее время при проектировании КИС базовой считается 3-х уровневая иерархическая структурная организация, в которой основной уровень обеспечивает подключение к устройствам уровня распределения, уровень распределения соединяет локальные сети, а уровень доступа обеспечивает подключение сетевых хостов и конечных устройств. Для дальнейшего разделения такой структурной в модульных областях достаточно успешно используется методология Cisco Enterprise Architecture Model (*Cisco Systems – один из мировых лидеров по разработке и поставке на рынок сетевого оборудования*) [97, 98].

**2.3. Разработка сетевой топологии и архитектуры КИС.** Является сложной комбинаторной многофакторной оптимизационной проблемой, решение которой существенно зависит от типов подсетей (т. е. кабельных, беспроводных, оптических), входящих в КИС. Рассмотрим кратко ее основные составляющие.

Известно, что для кабельных сетей минимизация стоимости соединений и переключателей в сети при наличии ограничений на ее производительность является NP-трудной задачей [7]. Поэтому на практике для ее решения часто используют эвристические методы: табу-поиск [99], имитацию отжига [100], генетические алгоритмы [101], методы, основанные на муравьиной колонии [102]. Для реальных кабельных КИС ситуация усложняется еще и тем, что стоимости соединений и переключателей зависят от топологии сети, т. е. не являются фиксированными ни вначале, ни в процессе оптимизации. Такая ситуация исследована в [103], где на основе локального поиска и геометрического подхода предложены пять методов решения этой задачи начиная от исчерпывающего поиска и заканчивая быстрыми эвристическими. Приведен сравнительный анализ этих методов на основе результатов проведенных экспериментов.

Для беспроводных сетей (кроме геометрического размещения объектов) возникает необходимость назначения таких определенных стандартами дефицитных параметров, как мощность и частота с целью максимизации количества охваченных приемников или доходов, связанных с их охватом. Известно, что (см., напр., [104]) эта задача является NP-трудной. В [105] предложен генетический алгоритм решения этой задачи. При этом в дополнение к эмиссии мощности и частоты в качестве параметра, значение которого устанавливается в результате решения задачи, рассматривается схема передачи (взрыв профиля) (*таким образом, для беспроводной КИС решается задача назначения мощности, частоты и модуляции*). Это обстоятельство дает возможность применить предложенный в [105] метод к таким беспроводным сетям нового поколения, как WiMAX [106].

Для оптоволоконных сетей естественно выделяются следующие две задачи: 1) проектирование физической топологии и конфигурации сети; 2) маршрутизация и присвоение длин волн. Проектирование физической топологии состоит в определении числа и связности оптических кросс-соединений (*современные оптические кросс-соединения осуществляют мультиплексирование разделения длины волны, т. е. длина волны оптического сигнала на входном оптоволокне может быть переключена на любую длину волны в выходном оптоволокне. Протоколы управления такими кросс-соединениями представлены, например, в [107, 108]*), предназначенных для обеспечения непрерывной цепи прохождения луча света при наличии большого числа отмеченных коммутирующих маршрутизаторов. Проектирование конфигурации сети состоит в определении размеров оптических кросс-соединений, количества оптоволоконных соединений и множества путей прохождения света. Маршрутизация включает в себя отображение путей прохождения света на физическую топологию (*таким образом, мы приходим к виртуальной топологии, в которой узлы соответствуют фактическим физическим сетевым узлам, а связи соответствуют путям прохождения света*). Различные формулировки и решения задач маршрутизации и присваивания длин волн вдоль маршрутов в предположении, что зафиксирована оптоволоконная топология, представлены в [109]. Даже при фиксированной физической топологии проектирование конфигурации сети является NP-трудной задачей (*эта задача сводится к задаче целочисленного программирования*), а

для ее решения, как правило, применяется метод ветвей и границ. При фиксированном множестве маршрутов задача присваивания длин волн вдоль маршрутов эквивалентна задаче раскраски графа. В этом случае для ее решения применяют эвристические методы [110]. В [111] предложен генетический алгоритм для решения задачи маршрутизации и присвоения длин волн в предположении, что зафиксирована физическая топология. В [112] предложены эвристические методы решения задачи минимизации количества оптических кросс-соединений при заданных ограничениях на количество длин волн в волоконно-оптической линии, в предположении, что физическая топология оптоволоконной сети является двусвязной.

Сложной задачей является размещение средств, позволяющих обеспечить эффективное функционирование КИС при больших объемах сетевого трафика и большом количестве устройств, подключенных к сети. Для этой цели предназначены центры обработки данных (ЦОД), которые используются для хранения и обработки огромного количества данных. В настоящее время проектирование ЦОД становится одним из наиболее востребованных направлений (по прогнозу аналитиков, к 2018 году объем мирового рынка ЦОД вырастет до 35 млрд долларов, причем 40% всех расходов на КИС будут так или иначе связаны с ЦОД. В первую очередь ЦОД будут востребованы сервис-провайдерами, облачными коммерческими центрами обработки данных, крупными корпоративными КИС), и требует существенных изменений в подходах к построению (понятие ЦОД приводит к пересмотру сетевой архитектуры, а именно: к отделению управления от передачи данных, а также к автоматизации процесса администрирования сетевого оборудования), эксплуатации и управлению КИС. При построении ЦОД основной топологией является fat-tree (fat-tree (утолщенное дерево) – дерево, в котором связи становятся более производительными по пропускной способности с каждым уровнем по мере приближения к корню дерева (часто используют удвоение пропускной способности на каждом уровне)) [113–115]. Однако сама по себе эта топология не направлена на синтез оптимальных ЦОД (качество ЦОД оценивается по совокупности параметров: стоимость-задержка-энергопотребление). В [116] предложена архитектура Diamond для ЦОД, представляющая собой модификацию fat-tree. В этой модификации улучшение качества ЦОД достигается за счет того, что все агрегированные переключатели в fat-tree заменяются переключателями связей, которые непосредственно подсоединяются к базовым переключателям. Такой подход дает возможность снизить в среднем на 10% задержку между концевыми вершинами. В [117] показано, каким образом при проектировании ЦОД может быть использована dragon fly [118] топология, с тем, чтобы сократить задержки в сети за счет уменьшения количества переходов между узлами, и общую стоимость ЦОД за счет уменьшения количества оптических кабелей, используемых для подключения подсетей.

Большие усилия исследователей посвящены разработке методологии проектирования распределенных КИС (см., напр., [119–121]). В этом направлении особое место занимают исследования пиринговых сетей (P2P nets) [122–124], представляющих собой оверлейные компьютерные сети, основанные на равноправии участников. Часто в такой сети отсутствуют выделенные серверы, а каждый узел является как клиентом, так и выполняет функции сервера. Такая организация позволяет сохранять работоспособность сети при любом количестве и любом сочетании доступных узлов. По-видимому, одним из наиболее перспективных вариантов пиринговой сети является сеть Gnutella [121, 126, 127]. Она представляет собой полностью децентрализованную файлообменную сеть в рамках Интернета, в которой отсутствует центральный сервер. Сеть Gnutella формируется, когда один ее пользователь соединяется с другим пользователем, после чего может происходить обмен информацией (обмениваться можно файлами любых форматов, для этого предусмотрено большое количество удобных функций) между ними. В результате полной децентрализации сеть Gnutella практически невозможно уничтожить (так как для этого необходимо вывести из строя каждый ее узел).

**2.4. Эмуляция КИС.** Является одной из ключевых технологий в исследованиях компьютерных сетей. Дает возможность проводить эксперименты, которые являются более реалистичными, чем моделирование, и более дешевыми, чем эксперименты на сетях, построенных на основе аппаратных средств (кроме того, сети, реализованные аппаратно, трудно перенастраивать, что существенно ограничивает возможность крупномасштабных экспериментов, в которых необходимы несколько различных тестовых вариантов сети). Существует ряд инструментальных средств, предназначенных для эмуляции КИС (например, EINAR [128], VNUML [129], Netkit [130, 131]). Хотя эмуляция устраняет необходимость физического размещения соединительных проводов, она требует конфигурирования таких устройств, как маршрутизаторы и коммутаторы. Известно, что конфигурирование маршрутизаторов – сложная задача [132, 133], а ее решение вручную – источник "человеческих ошибок", которые трудно выявить. Поэтому разработка средств автоконфигурации маршрутизаторов при эмулировании КИС является актуальной задачей. В [134] предложен один из вариантов решения этой задачи – инструментальное средство автоконфигурации маршрутизаторов AutoNetkit, предназначенное для эмуляции КИС с использованием инструментальных средств Netkit.

**2.5. Управление КИС.** Представляет собой сложную многогранную проблему. Рассмотрим кратко некоторые ее основные составляющие.

Основные принципы конструирования Интернет-трафика, его онлайн и оффлайн технические возможности, системы его поддержки и варианты его развития систематически изложены в [135, 136]. В [137] исследована задача конструирования трафика в рамках одной автономной системы (такой как фирма, университетский городок, провайдер интернет услуг и т. д.). Показано, как адаптировать конфигурацию весов связей в позиции трафика всей сети и топологии внутри домена. В [138] предложен метод быстрого вывода матриц трафика в IP-сетях на основе измерений нагрузки связей, дополненных легкодоступной информацией о сетевой конфигурации и маршрутизации. Эффективность предложенного метода проиллюстрирована его применением к вычислению матриц трафика между магистральными маршрутизаторами достаточно большой Tier-1 IP сети (Tier-1 IP сеть – сеть, в которой оператор имеет доступ к сети Интернет исключительно через пиринговые соединения). В [139] содержится обзор состояния инжиниринга Интернет трафика с позиции оптимизации маршрутизации. Соответствующие алгоритмы классифицированы в следующих измерениях: одноадресный / мультиадресный, внутри-доменный / меж-доменный, IP / MPLS, оффлайн схема / онлайн схема. Охарактеризована надежность Интернет трафика и его совместимость с оверлейной маршрутизацией. Отмечены сложности, возникающие при применении инжиниринга Интернет трафика и некоторые проблемы, требующие дальнейшего исследования.

В п. 2.3 отмечено, что в настоящее время большое внимание уделяется разработке ЦОД. Однако известно мало результатов, связанных с анализом характеристик трафика сетевого уровня современных ЦОД. В [140] исследованы сетевые трафики десяти ЦОД различных типов организаций: университет, предприятие и облако. При этом в определении ЦОД облака кроме ЦОД крупных провайдеров, предлагающих интернет-приложения, включены также ЦОД, используемые для приложений с интенсивной обработкой данных (*иными словами, приложения, написанные с помощью MapReduce*). Была проанализирована статистика протокола сетевого управления, топология и следы на уровне пакетов. Развернутые в исследуемых ЦОД приложения рассмотрены с позиции их размещения, уровня потока данных, передачи на уровне следов пакетов, влияния на использование сети и связей в ней, загруженности сети и потери пакетов. Охарактеризовано влияние установленных закономерностей на инжиниринг внутреннего трафика, а также на архитектуру современных ЦОД. В [141] представлена система MicroTE, функционирующая поверх основных топологий ЦОД, предназначенная для смягчения последствий перегруженности сети, вызванной непредсказуемым трафиком, за счет использования частично предсказуемой матрицы краткосрочного трафика (*таким образом, система MicroTE адаптирует сеть к изменениям трафика*). Система MicroTE реализована в рамках OpenFlow с незначительными изменениями в концевых хостах. Показано, что система MicroTE реализует решение, близкое к оптимальному решению, при минимальных накладных расходах на сеть, что делает его подходящим для существующих и будущих ЦОД. Обзор существующих схем передачи потоков данных в ЦОД содержится в [142]. Отмечены преимущества и недостатки этих схем. Обосновано, почему некоторые версии транспортных протоколов не могут быть эффективными в ЦОД.

Значительное число реализованных в сетях алгоритмов и сетевых протоколов включают в себя механизмы распространения информации (*например, при маршрутизации в сети для пересылки сообщения осуществляется поиск адреса и отслеживание маршрута в его направлении*). Исследования, направленные на разработку механизмов перераспределения трафика в интернете из мест, менее релевантных запросу в места с большей релевантностью по этому запросу получили имя поисковый маркетинг. Эффективность механизма распространения информации в КИС оценивается по совокупности параметров: число сообщений и время завершения. В [143] исследованы две основные составляющие механизма распространения информации в КИС – поиск и маркетинг. Предложены критерии, характеризующие правильность, быстроту и честность (fairness) применения этого механизма.

Программно-конфигурируемые сети (SDN) передачи данных [144, 145] являются одной из форм виртуализации вычислительных ресурсов. Они характеризуется тем, что в них уровень управления отделен от устройств передачи данных и реализуется программно. В [146] содержится обзор современного состояния исследований SDN. Основное внимание уделено анализу управления потоками данных, отказоустойчивости сетей, обновлению топологии и средствам анализа трафика.

Эффективность КИС в значительной мере зависит от количества и расположения сервисных центров, размещенных на различных хостах. В [147] предложен новый подход к добавлению или удалению серверов в КИС на основе использования только локальной информации о топологии сети и запросах. Показано, что в течение одной-двух итераций достигается производительность КИС, сопоставимая с применением централизованных подходов, требующих полную информацию о топологии и запросах всей КИС.

**3. Безопасность КИС.** Является одной из ключевых проблем современных сетевых технологий. Рассмотрим кратко некоторые ее основные составляющие.

Практика показала, что к масштабным отказам, сбоям и разрушению маршрутизации КИС могут привести как стихийные бедствия [148–150], так и преднамеренные нападения [151–153]. Большинство существующих подходов к восстановлению КИС являются, по своей сути, проактивными и используют резервные предварительно вычисленные пути для перенаправления трафиков, пострадавших от неисправностей, во время конвергенции IGP (протокол внутреннего шлюза) [154–156]. Такие проактивные подходы построены на неявном предположении о том, что в КИС возникают только спорадические изолированные неисправности связей. По этой причине эти подходы не применимы для крупномасштабных неисправностей, так как связи и резервные пути могут быть разрушены одновременно. Кроме того, ни один маршрутизатор не имеет всю информацию о неисправностях; он только знает, достижимы ли его соседи. Для недостижимого соседа маршрутизатор не может различить ситуации: неисправен ли сосед или соединяющая его связь. В [157] решена задача быстрого восстановления маршрутизации внутри домена при крупных авариях. Идея предложенного подхода состоит в следующем. Вначале осуществляется сбор информации об отказе с помощью разработанного протокола пересылки специальных пакетов вокруг области неисправности и записи информации о неисправности в заголовке пакета. Затем вычисляются кратчайшие исправные пути, по которым направляются пакеты. В [158] исследуется эксплуатационная практика, стандарты и текущие исследования в области безопасности междоменной маршрутизации, сходства и различия в существующих подходах к построению более безопасной инфраструктуры Интернет.

Для обеспечения безопасности КИС большое значение имеет защита от нападений, осуществляемых на базе Интернет. Эта проблема особенно остро стоит перед корпоративными КИС. Одной из составляющих для ее решения является развертывание набора продуктов мониторинга безопасности, которые генерируют результаты "ситуационной разведки" в форме различных журналов. Такие журналы, как правило, содержат большие объемы информации о деятельности в сети, и являются одними из первых источников данных, которые анализируют специалисты по информационной безопасности, когда они подозревают, что произошла атака на сеть. Продукты мониторинга безопасности часто поступают от различных поставщиков, устанавливаются и администрируются несогласованно друг с другом. Как следствие, форматы генерируемых журналов отличаются друг от друга, журналы часто являются неполными, взаимно противоречивыми и очень большими по объему. В результате, собранная информация, хотя и является полезной, часто беспорядочна и содержит много "мусора". В [159] представлена система Veehive, осуществляющая автоматическое извлечение знаний из "мусорных данных" журналов, созданных для широкого спектра продуктов мониторинга безопасности корпоративных КИС. Эта система вместо выявления подозрительного поведения узла сообщает о потенциальных инцидентах, связанных с безопасностью. Далее инциденты могут быть проанализированы группами реагирования, чтобы определить, произошло ли нарушение политики безопасности, либо атака на сеть. На основе проведенных экспериментов уста-

новлено, что система Beehive в состоянии идентифицировать злонамеренные действия и нарушения политики безопасности, которые в противном случае остались бы незамеченными.

Известно, что в онлайн-социальных сетях существует достаточно много рисков, связанных с нарушением конфиденциальности и безопасности (см., напр., [160]). Эти риски возникают из-за неясного доверия, встроенного в заявленных социальных отношениях, что дает возможность для сбора личной информации о пользователях для тех или иных целей. Отметим, что в настоящее время в онлайн-социальных сетях принято выставлять информацию о пользователе, собранную из различных социальных сфер (например, личную информацию из Facebook, профессиональную деятельность из LinkedIn и т. д.), что приводит к излишне детализированным профилям [161]. Используя такую доступность информации, фирмы выискивают на Facebook и Twitter тенденции для создания вирусного контента для акций и лайков, работодатели проверяют на Facebook, LinkedIn и Twitter профили кандидатов на должность [162], правоохранительные органы подбирают в онлайн-социальных сетях доказательства при раскрытии преступлений [163]. Кроме того деятельность в онлайн-социальных платформах используется для изменения политических режимов [164] и для влияния на результаты выборов [165]. Пресечение масштабных вползаний (crawls) пользовательских профилей в такие онлайн-социальные сети, как Facebook и Renren отвечает интересам как пользователей, так операторов этих сайтов. Пользователи стремятся сохранить контроль над своей личной информацией, а операторы – защитить свои активы и бизнес-репутацию. Существующие методы ограничения скорости неэффективны против краулеров (*Краулер – программа, являющаяся составной частью поисковой системы и предназначенная для перебора страниц Интернета с целью занесения информации о них в базу данных поисковика. По принципу действия краулер напоминает обычный браузер. Он анализирует содержимое страницы, сохраняет его в некотором специальном виде на сервере поисковой машины, которой принадлежит, и отправляется по ссылкам на следующие страницы. Владельцы поисковых машин нередко ограничивают глубину проникновения краулера внутрь сайта и максимальный размер сканируемого текста, поэтому слишком большие сайты могут оказаться не полностью проиндексированными поисковой машиной*) с большим количеством аккаунтов, будь то фальшивые аккаунты (Sybils) или взломанные аккаунты реальных пользователей, полученные на черном рынке. В [166] представлена система Genie, предназначенная для защиты от краулеров в крупномасштабных онлайн-социальных сетях. Эта система использует тот фактор, что шаблоны просмотра честными пользователями и краулерами сильно отличаются друг от друга: даже краулеру с доступом ко многим аккаунтам необходимо просмотреть намного больше профилей на каждый аккаунт, чем честному пользователю. Краулер также просматривает профили пользователей, которые сильно отдалены друг от друга в социальной сети. Эксперименты с использованием реальных данных, собранных из популярных крупномасштабных онлайн-социальных сетей показали, что Genie расстраивает масштабные вползания и редко оказывает влияние на честных пользователей; пострадавших пользователей можно легко восстановить, добавив несколько ссылок о дружбе.

Критическую угрозу для сетевой безопасности представляют различные вредоносные программы (malware). Современные модели распространения вредоносных программ в КИС делятся на два класса. К первому классу относятся модели, построенные на основе теории управляющих систем [167]. На их основе разрабатываются методы обнаружения и ограничения распространения вредоносных программ. Ко второму классу относятся модели, построенные на основе эпидемиологии [168, 169]. Эти модели предназначены для исследования законов распределения количества зараженных хостов. Отметим, что на сегодняшний день нет полного понимания поведения вредоносных программ в КИС. Поэтому исследование законов их распространения в КИС является актуальной задачей. В [170] с глобальной точки зрения исследуется, каким образом вредоносные программы распространяются в КИС. Новизна предложенного подхода по сравнению с традиционным подходом, основанным на использовании эпидемиологических моделей, состоит в том, что анализ нарушений в модели осуществляется в соответствии со следующими двумя этапами. Вначале на основе восприимчивой к инфицированию модели [171] вычисляется количество сетей, скомпрометированных в течение заданного времени с момента прорыва КИС вредоносной программой. Далее для взломанной сети рассчитывается, сколько хостов было скомпрометировано с момента нарушения сети. Доказано, что распределение вредоносной программы в КИС подчиняется экспоненциальному закону на ранней стадии, степенному распределению с коротким экспоненциальным хвостом на поздней стадии, и сходится к степенному распределению. Достоверность теоретических результатов проиллюстрирована их применением к реальным данным. В [172] представлена система Nazca, предназначенная для обнаружения инфицирования КИС. Эта система выявляет явные признаки сетевых инфраструктур, которые руководят инсталляцией вредоносных программ. Процесс выявления этих признаков осуществляется на основе анализа коллективного трафика, производимого большим числом пользователей КИС (*именно такой подход обеспечивает системе Nazca большую скорость и эффективность по сравнению с системами, основанными на анализе соединений, свойств загруженных программ и репутации серверов*). На систему Nazca не влияет полнота зон покрытия пробелов в репутационных базах (т. е. черных списках). Кроме того, эта система не поддается на запутывание кода. Эффективность системы Nazca была проиллюстрирована результатами ее работы в течение семи дней с крупным Интернет провайдером. Были обнаружены ранее не выявленные вредоносные программы, причем число ложных срабатываний системы оказалось небольшим.

**Заключение.** В работе кратко рассмотрены некоторые актуальные проблемы в следующих направлениях, связанных с анализом и синтезом КИС: выделение сообществ на основе топологии сети, анализ социальных сетей, проектирование КИС, обеспечение безопасности КИС. Следует отметить, что существующие в настоящее время подходы к решению этих проблем весьма далеки от того, что принято называть проработанной технологией.

За рамками данного обзора осталось много актуальных проблем в области анализа и синтеза КИС, которые в настоящее время находятся только на стадии исследования, и анализ ситуации с каждой из которых является темой для соответствующего аналитического обзора. К ним, в частности, относится весь комплекс проблем анализа и синтеза сенсорных сетей, проблемы эффективной и безопасной организации облачных вычислений, проблемы эффективной криптографической защиты информации в КИС с критической областью применения и т. д.

В заключение отметим, что большинство проблем анализа и синтеза КИС настолько тесно взаимосвязаны, что попытки обеспечить оптимальное решение каждой из проблем при ее изолированном исследовании из-за возможных противоречий между этими решениями приведут к созданию КИС, которая окажется неэффективной и недостаточно защищенной.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.

1. Girvan M., Newman M. Community structure in social and biological networks // Proc. of the National Acad. of Sci. of the USA. – 2002. – Vol. 99. – P. 7821–7826.
2. Radicchi F., Castellano C., Cecconi F., et al. Defining and identifying communities in networks // Proc. of the National Acad. of Sci. of the USA. – 2004. – Vol. 101. – P. 2658–2663.
3. Hastings M. Community detection as an inference problem. – <http://arxiv.org/pdf/cond-mat/0604429.pdf>
4. Buzun N., Korshunov A. Innovative methods and measures in overlapping community detection // Proc. of International Workshop on Experimental Economics in Machine Learning. – 2012. – P. 20–32.
5. Yang J., Leskovec J. Defining and evaluating network communities based on ground-truth // Proc. of IEEE 12<sup>th</sup> International Conference on Data Mining. – 2012. – P. 745–754.
6. Chykhhradze K., Korshunov A., Buzun N., et al. Distributed generation of billion-node social graphs with overlapping community structure // Proc. of Complex Networks V. Studies in Computational Intelligence. – 2014. – Vol. 549. – P. 199–208.
7. Гари М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. – М.: Мир, 1982. – 416 с.
8. Tomita E., Tanaka A., Takahashi H. The worst-case time complexity for generating all maximal cliques // Computing and Combinatorics. – Berlin/Heidelberg: Springer. – 2004. – P. 161–170.
9. Fortunato S., Castellano C. Community structure in graphs. – <http://arxiv.org/abs/0712.2716.pdf>
10. Xie J., Kelly S., Szymanski B. Overlapping community detection in networks: the state of the art and comparative study. – <http://dx.doi.org/10.1145/2501654.2501657>
11. Harenberg S., Bello G., Gjeltema L., et al. Community detection in large-scale networks: a survey and empirical evaluation // WIREs Comput. Statistics. – 2014. – Vol. 6. – P. 426–439.
12. Newman M. Modularity and community structure in networks. – <http://www.pnas.org/content/103/23/8577.full>
13. Newman M., Girvan M. Finding and evaluating community structure in networks. – <http://link.aps.org/abstract/PRE/v69/e026113>
14. Ziv E., Middendorf M., Wiggins C. Information-theoretic approach to network modularity. – <http://journals.aps.org/pre/abstract/10.1103/PhysRevE.71.046117>
15. Friggeri A., Cheluis G., Fleury E. Triangles to capture social cohesion. – <http://arxiv.org/pdf/1107.3231.pdf>
16. Arenas A., Fernandez A., Fortunato S., Gomez S. Motif-based communities in complex networks. – <http://arxiv.org/pdf/0710.0059.pdf>
17. Clauset A., Newman M., Moore C. Finding community structure in very large networks. – <http://arxiv.org/abs/cond-mat/0408187>
18. Newman M. Fast algorithm for detecting community structure in networks. – <http://arxiv.org/abs/cond-mat/0309508>
19. Pons P., Latapy M. Computing communities in large networks using random walks // Proc. of the 20<sup>th</sup> International Conference on Computer and Information Sciences. – 2005. – P. 284–293.
20. De Meo P., Ferrara E., Fiumara G., Provetti A. Mixing local and global information for community detection in large networks // Journal of Computing Systems Sciences. – 2014. – Vol. 80. – P. 72–87.
21. White S., Smyth P. A spectral clustering approach to finding communities in graph // Proc. of SIAM Data Mining Conference. – 2005. – P. 76–84.
22. Newman M. Finding community structure in networks using the eigenvectors of matrices. – <http://arxiv.org/abs/physics/0605087>
23. Kirkpatrick S., Gelatt C.D., Vecchi M.P. Optimization by simulated annealing // Science. – 1983. – Vol. 220. – P. 671–680.
24. Guimera R., Sales-Pardo M., Amaral L. Modularity from fluctuations in random graphs and complex networks. – <http://dx.doi.org/10.1103/PhysRevE.70.025101>
25. Liu J., Liu T. Detecting community structure in complex networks using simulated annealing with k-means algorithms // Physica A.: Statistical Mechanics and its Applications. – 2010. – Vol. 389. – № 11. – P. 2300–2309.
26. Штовба С. Муравьиные алгоритмы // Экспонента Про. Математика в приложениях. – 2003. – №4. – С. 70–75.
27. Dorigo M., Stützle T. Ant colony optimization: overview and recent advances // IRIDIA Technical Report Series. – 2009. – Technical Report № TR/IRIDIA/2009-013. – 32 p.
28. Лебедев О.Б. Разбиение на основе муравьиной колонии // Материалы XV Международной конференции по нейрокибернетике. Т. 2. – Ростов-на-Дону: Изд-во ЮФУ. – 2009. – С. 102–105.
29. Liu Y., Luo J., Yang H., Liu L. Finding closely communicating community based on ant colony clustering model // Proc. of the International Conference on Artificial Intelligence and Computational Intelligence. – 2010. – P. 127–131.
30. Sadi S., Oguducu S., Uyar A. An efficient community detection method using parallel clique-finding ants // Proc. of IEEE Congress on Computational Intelligence. – 2010. – P. 1–7.
31. Jin D., Liu D., Yang B., et al. Ant colony optimization with markov random walk for clustering in complex networks // Proc. of the 15<sup>th</sup> Pacific-Asia Conference on Knowledge Discovery and Data Mining. – 2011. – P. 123–134.
32. Palla G., Derenyi I., Farkas I., Vicsek T. Uncovering the overlapping community structure of complex networks in nature and society // Nature. – 2005. – Vol. 435. – P. 814–818.
33. Lee C., Reid F., McDaid A., Hurley N. Detecting highly overlapping community structure by greedy clique expansion // Proc. of SNAKDD Workshop. – 2010. – P. 33–42.
34. Gregori E., Lenzini L., Orsini C. k-clique communities in the Internet AS-level topology graph // Proc. of ICDCS Workshops. – 2011. – P. 134–139.
35. Gregori E., Lenzini L., Mainardi S. Parallel k-clique community detection on large-scale networks // IEEE Trans. on Parallel and Distributed Systems. – 2012. – № 8. – P. 1651–1660.
36. Bagrow J., Bollt E. A local method for detecting communities. – [arXiv:cond-mat/0412482v2](http://arxiv.org/abs/cond-mat/0412482v2)
37. Raghavan U., Albert R., Kumara S. Near linear-time algorithm to detect community structures in large-scale networks. – <http://arxiv.org/abs/0709.2938v1>
38. Costa L. Hub-based community finding. – <http://arxiv.org/abs/cond-mat/0405022v1>
39. Xie J., Szymanski B. Towards linear time overlapping community detection in social networks // Proc. of PAKDD. – 2012. – P. 25–36.
40. Xie J., Szymanski B. Community detection using a neighborhood strength driven label propagation algorithm. – <http://arxiv.org/pdf/1105.3264>
41. Barber M. Detecting network communities by propagating labels under constraints. – <http://arxiv.org/abs/0903.3138>
42. Leung I., Hui P., Li P., Crowcroft J. Towards real-time community detection in large networks. – <http://arxiv.org/pdf/0808.2633.pdf>
43. Gregory S. Finding overlapping communities in networks by label propagation. – <http://arxiv.org/pdf/0910.5516>
44. Prabavathi G., Thiagarasu V. Design and development of overlapping community detection algorithm using multi-level propagation. – <http://www.ijarcsms.com/February2014.htm>
45. Tibely G., Kertesz J. On the equivalence of the label propagation method of community detection and a potts model approach // Physica A. – 2008. – Vol. 387. – P. 4982–4984.
46. Kumpula J., Saramaki J., Kaski K., Kertesz J. Limited resolution in complex network community detection with potts model approach // European Physical Journal B. – 2007. – Vol. 56. – P. 41–45.
47. Macropol K., Singh A. Scalable discovery of best clusters on large graphs // Proc. of VLDB Endowment. – 2010. – № 3. – P. 693–702.
48. Satuluri V., Parthasarathy S., Ruan Y. Local graph sparsification for scalable clustering // Proc. of the International Conference on Management of Data. – 2011. – P. 721–732.
49. Song H. Clustered embedding of massive social networks // ACM SIGMETRICS Performance Evaluation Review. – 2012. – № 1. – P. 331–342.
50. Tantawi A. A scalable algorithm for placement of virtual clusters in large data centers // Proc. of MASCOTS. – 2012. – P. 3–10.
51. Sui X., Lee T., Whang J., et al. Parallel clustered low-rank approximation of graphs and its application to link prediction // Languages and Compilers for Parallel Computing. – Berlin/Heidelberg: Springer. – 2013. – P. 76–95.
52. Baumes J., Goldberg M., Krishnamoorthy M., et al. Finding communities by clustering a graph into overlapping subgraphs // Proc. of the IADIS International Conference on Applied Computing. – 2005. – P. 97–104.
53. Kelley S. The existence and discovery of overlapping communities in large-scale networks. – Ph.D. thesis, Rensselaer Polytechnic Institute, Troy, NY. – 2009.
54. Lanchinetti A., Fortunato S., Kertesz J. Detecting the overlapping and hierarchical community structure of complex networks. New J. Phys. – 2009. – Vol. 11. – № 3. – 20 p.
55. Havemann F., Heinz M., Struck A., Glaser J. Identification of overlapping communities and their hierarchy by locally calculating community-changing resolution levels. – 2011. – J. Statist. Mech. – № 1. – P. 10123.
56. Airolidi E., Blei D., Fienberg S., Xing E. Mixed membership stochastic blockmodels // J. Mach. Learn. Res. – 2008. – № 9. – P. 1981–2014.
57. Wasserman S., Anderson C. Stochastic a posteriori blockmodels: Construction and assessment // Social Networks. – 1987. – № 1. – P. 1–36.
58. Karrer B., Newman M. Stochastic blockmodels and community structure in networks. – <http://dx.doi.org/10.1103/PhysRevE.83.016107>
59. Bickel P., Chen A. A nonparametric view of network models and Newman–Girvan and other modularities // Proc. of the National Acad. of Sci. of the USA. – 2009. – Vol. 106. – P. 21068–21073.

60. Zhang S., Wang R., Zhang X. Identification of overlapping community structure in complex networks using fuzzy c-means clustering // *Physica A*. – 2007. – Vol. 374: –P. 483–490.
61. Gregory S. Fuzzy overlapping communities in networks. – <http://arxiv.org/abs/1010.1523>
62. Liu J. Fuzzy modularity and fuzzy community structure in networks // *The European Physical Journal B*. – 2010. – № 4. – P. 547–557.
63. Nepusz T., Pécroci A., Négyessy L., Bazsó F. Fuzzy communities and the concept of bridgeness in complex networks. – <http://arxiv.org/abs/0707.1646v3>
64. Lasarsfeld P.F. On Social Research and Its Language. – University of Chicago Press. – 1993. – 342 p.
65. Prentice D. A., Miller D. T., Lightdale J. R. Asymmetries in attachments to groups and to their members: Distinguishing between common-identity and common-bond groups // *Personality and Social Psychology Bulletin*. – 1994. – № 5. – P. 484–493.
66. Garton L., Haythornthwaite C., Wellman B. Studying online social networks // *Journal of Computer Mediated Communication*. – 1997. – Vol. 3. – № 1. – P. 75–106.
67. Yang B., Liu D., Liu J. Discovering communities from social networks: methodologies and applications // *Handbook of Social Network Technologies and Applications*. – Berlin/Heidelberg: Springer, 2010. – P. 331–346.
68. Tang L., Liu H. Community detection and mining in social media // *Synthesis Lectures on Data Mining and Knowledge Discovery*. – 2010. – № 1. – P. 1–137.
69. Martin A., Wellman B. Social Network Analysis: An Introduction // *Handbook of Social Network Analysis* / ed. by P. Carrington, J. Scott, 2011. – Thousand Oaks, CA: Sage. – P. 11–25.
70. Bollobas B. Modern graph theory. – NY: Springer-Verlag, 1998. – 394 p.
71. Чураков А. Н. Анализ социальных сетей // *СоцИс*. – 2001. – № 1. – С. 109–121.
72. Батура Т.В. Методы анализа компьютерных социальных сетей // *Вестник НГУ*. – 2012. – Т.12. – Вып. 4. – С. 13–28.
73. Social network data analytics / Ed. Charu C. Aggarwal. – Springer US: Science+Business Media, LLC, 2011. – eBook ISBN 978-1-4419-8462-3. – 520 p.
74. Granovetter M. S. The strength of weak ties // *American Journal of Sociology*. – 1973. – № 6. – P. 1360–1380.
75. Scheidegger M., Baumgartner F., Braun T. Simulating large-scale networks with analytical models // *International Journal of Simulation*. – 2005. – Vol.6. – № 1–2. – P. 24–31.
76. Gregory S. An algorithm to find overlapping community structure in networks. – <https://www.cs.bris.ac.uk/~steve>
77. Lancichinetti A., Radicchi F., Ramasco J., Fortunato S. Finding statistically significant communities in networks. – <http://santo.fortunato.googlepages.com/inthepress2>
78. Clauset A., Moore C., Newman M. Hierarchical structure and the prediction of missing links in networks // *Nature*. – 2008. – Vol. 453. – P. 98–101.
79. Ahn Y., Bagrow J., Lehmann S. Link communities reveal multi-scale complexity in networks // *Nature*. – 2010. – Vol. 466. – P. 761–764.
80. Rosvall M., Bergstrom C. Maps of random walks on complex networks reveal community structure // *Proc. of the National Acad. of Sci. of the USA*. – 2008. – Vol. 105. – P. 1118–1123.
81. McDaid A., Hurley N. Using model-based overlapping seed expansion to detect highly overlapping community structure. – <http://sites.google.com/site/aaronmcdaid/amos>
82. Wang X., Sun J., Chen Z., Zhai C. Latent semantic analysis for multiple-type interrelated data objects // *Proceedings of the 29<sup>th</sup> Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*. – 2006. – NY: ACM. – P. 236–243.
83. Stern D., Herbrich R., Graepe T. Matchbox: large scale online bayesian recommendations // *Proceedings of the 18<sup>th</sup> International Conference on World Wide Web (WWW '09)*. – 2006. – NY: ACM. – P. 111–120.
84. Машечкин Д. В. Петровский И. В., Царев М. И. Методы вычисления релевантности фрагментов текста на основе тематических моделей в задаче автоматического аннотирования // *Вычислительные методы и программирование*. – 2013. – Т. 14. – С. 91–102.
85. Коршунов А. В. Гомзин А. Г. Тематическое моделирование текстов на естественном языке // *Труды Института системного программирования РАН*. – 2012. – Вып. 23. – С. 216–243.
86. Давыдов А. А. Системная социология: анализ мультимедийной информации в Интернете. – [http://www.isras.ru/index.php?page\\_id=988](http://www.isras.ru/index.php?page_id=988)
87. Прокофьев В. Ф. Тайное оружие информационной войны: атака на подсознание. – М.: СИНТЕГ, 2003. – 408 с.
88. Freeman L. The development of social network analysis: a study in the sociology of science. – Vancouver: Empirical Press, 2004. – 208 p.
89. Губанов Д. А., Новиков Д. А., Чхартишвили А. Г. Модели информационного влияния и информационного управления в социальных сетях // *Проблемы управления*. – 2009. – № 5. – С. 28–35.
90. Губанов Д. А., Новиков Д. А., Чхартишвили А. Г. Социальные сети: модели информационного влияния, управления и противоборства. – М.: Изд-во физико-математической литературы, 2010. – 228 с.
91. Ware C. Information visualization: perception for design. Waltham, MA: Morgan Kaufmann. – 2013. – 512 p.
92. Koutsofio E., North S., Truscott R., Keim D. Visualizing Large-Scale Telecommunication Networks and Services // *Proc. of IEEE Visualization*. – 1999. – P. 457–461.
93. Скобелев В. Г. Безопасность IT систем (обзор) // *Радиоэлектронні і комп'ютерні системи*. – 2013. – № 5. – С. 352–361.
94. Буй Д. Б., Скобелев В. Г. Безопасность программных средств: модели и методы (обзор) // *Радиоэлектронні і комп'ютерні системи*. – 2014. – № 1. – С. 42–54.
95. Буй Д. Б., Скобелев В. Г. Модели, методы и алгоритмы оптимизации запросов в базах данных (обзор) // *Радиоэлектронні і комп'ютерні системи*. – 2014. – № 1. – С. 43–58.
96. Ретана А., Слайс Д., Уайт П. Принципы проектирования корпоративных IP-сетей. – М.: Вильямс, 2002. – 368 с.
97. [http://www.netakademija.rs/pdf/ccna%20r&s/04.Connecting%20Networks/CN\\_instructorPPT\\_Chapter1\\_final.pdf](http://www.netakademija.rs/pdf/ccna%20r&s/04.Connecting%20Networks/CN_instructorPPT_Chapter1_final.pdf)
98. Hutton K., Schofield M., Teare D. Authorized self-study guide: designing Cisco network service architectures (ARCH), 2009. – Indianapolis, IN: Cisco Press. – 672 p.
99. Glover F., Laguna M. Tabu search // *Modern heuristic techniques for combinatorial problems* / Ed. C.R. Reeves. – 1993. – NY: John Wiley & Sons, Inc. – P. 70–150.
100. Aarts E., Lenstra J. Local search in combinatorial optimization. – NY: John Wiley & Sons, Inc., 1997. – 488 p.
101. Winter G., Periaux J., Galan M. Genetic Algorithms in Engineering and Computer Science. – NY: John Wiley & Sons, Inc., 1995. – 195 p.
102. Dorigo M., Di Caro G., Gambardella L. Ant Algorithms for Discrete Optimization // *Artificial Life*. – 1999. – № 5. – P. 137–172.
103. Grout V., Cunningham S., Picking R. Practical large-scale network design with variable costs for links and switches // *International Journal of Computer Science and Network Security*. – 2007. – № 7. – P. 113–125.
104. Mannino C., Rossi F., Smriglio S. The network packing problem in terrestrial broadcasting // *Operation Research*. – 2006. – № 6. – P. 611–626.
105. D'Andreagiovanni F. On improving the capacity of solving large-scale wireless network design problems by genetic algorithms // *LNCS*. – 2011. – Vol. 6625. P. 11–20.
106. Andrews J., Ghosh A., Muhamed R. Fundamentals of WiMAX: understanding broadband wireless networking. – Upper Saddle River: Prentice Hall, 2007. – 448 p.
107. Su D., Griffith D. Standards activities for MPLS over WDM networks // *Optical Networks*. – 2000. – № 3. – P. 6–69.
108. CISCO MPLS Web Page. – <http://www.cisco.com/warp/public/732/Tech/mpsl/>
109. Karasan E., Ayanoglu E. Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks // *IEEE Journal on Selected Areas of Communications*. – 1998. – Vol. 16. – P. 1081–1096.
110. Zhang Z., Acampora A. A heuristic wavelength assignment algorithm for multipath WDM networks with wavelength routing and wavelength reuse // *IEEE/ACM Trans. Networking*. – 1995. – Vol. 3. – P. 281–288.
111. Saha D., Purkayastha M., Mukherjee B. An approach to wide area WDM optical network design using genetic algorithms // *Computer Communications*. – 1999. – Vol. 22. – P. 156–172.
112. Xin Y., Rouskas G., Perros H. On the physical and logical topology design of large-scale optical networks // *Journal of Lightwave Technology*. – 2003. – № 4 – P. 904–915.
113. InfiniBand in the Enterprise Data Center: Scaling 10Gb/s Clustering at Wire-Speed. White Paper, Mellanox Technologies, 2006. – [http://www.mellanox.com/pdf/whitepapers/InfiniBand\\_EDS.pdf](http://www.mellanox.com/pdf/whitepapers/InfiniBand_EDS.pdf)
114. Al-Fares M., Loukissas A., Vahdat A. A Scalable, commodity datacenter network architecture // *ACM SIGCOMM Computer Communication Review*. – 2008. – № 4. – P. 63–74.
115. Mysore R., Pamboris A., Farrington N., et al. A scalable fault-tolerant layer 2 data center network fabric // *ACM SIGCOMM Computer Communication Review*. – 2009. – № 4. – P. 39–50.
116. Sun Y., Chen J., Liu Q., Fang W. Diamond: an improved fat-tree architecture for large-scale data centers // *Journal of Communications*. – 2014. – № 1. – P. 91–98.

117. Delimitrou C., Mohammadi M., Nothhaft F., Sharpless L. Datacenter network design: performance/power comparison of large-scale network configurations and a way to avoid it! – [https://stanford.edu/~milad/DC\\_Network\\_v1.pdf](https://stanford.edu/~milad/DC_Network_v1.pdf)
118. Kim J., Dally W., Scott S., Abts D. Technology-driven, highly-scalable dragonfly topology // Proc. of the International Symposium on Computer Architecture (ISCA). – 2008. – P. 77–88.
119. Menasce D., Goma H., Kerschberg L. A performance oriented design methodology for large-scale distributed data intensive information systems // Proc. of The First IEEE International Conference on Engineering of Complex Computer Systems. – 1995. – P. 72–79.
120. Kerschberg L., Weishar D. Conceptual Models and Architectures for Advanced Information Systems // Applied Intelligence. – 2000. – Vol. 13. – P. 149–164.
121. Risse T. Design and configuration of distributed job processing systems. – [http://tuprints.ulb.tu-darmstadt.de/665/1/thomas\\_risse\\_diss.pdf](http://tuprints.ulb.tu-darmstadt.de/665/1/thomas_risse_diss.pdf)
122. Saroiu S., Gummadi P., Gribble S. A measurement study of peer-to-peer file sharing systems // University of Washington Technical Report UW-CSE-01-06-02, July 2001.
123. Karagiannis T., Broido A., Brownlee N., et al. File-sharing in the Internet: A characterization of P2P traffic in the backbone – <http://www.cs.ucr.edu/~tkarag>
124. Karagiannis T., Broido A., Faloutsos M. Transport layer identification of P2P traffic // Proc. of the 4<sup>th</sup> ACM SIGCOMM Conference on Internet Measurement. – 2004. – P. 121–134.
125. Adar E., Huberman D. Free riding on Gnutella. – <http://firstmonday.org/ojs/index.php/fm/article/view/792>
126. DSS Group, Gnutella: To the bandwidth barrier and beyond. – <http://dss.clip2.com>
127. Ripeanu M., Foster I., Iamnitchi A. Mapping the Gnutella network: properties of large-scale peer-to-peer systems and implications for system design // IEEE Internet Computing Journal. – 2002. – № 1. – P. 50–57.
128. EINAR. Einar router simulator. – <http://www.isk.kth.se/proj/einar>
129. Galan F., Fernandez D., Ruiz A., et al. Use of virtualization tools in computer network laboratories // Proc. International Conference on Information technology Based Higher Education and Training. – 2004. – P. 209–214.
130. University of Roma Tre. Computer Networks Research Group. Netkit. – <http://www.netkit.org/>
131. Rimondini M. Emulation of Computer Networks with Netkit. – Roma Tre University: Technical Report RT-DIA-113-2007.
132. Enck W., McDaniel P., Sen S., et al. Configuration management at massive scale: system design and experience // Proc. of the 2007 USENIX Annual Technical Conference. – 2007. – P. 1–14.
133. Bellovin S., Bush R. Configuration management and security // IEEE Journal on Selected Areas in Communications. – 2009. – № 3. – P. 268–274.
134. Nguyen H., Roughan M., Knight S., Falkner N., et al. How to build complex, large-scale emulated networks. – <https://nguyentuanhung.files.wordpress.com/2014/07/autonetkit.pdf>
135. Awduche D., Chiu A., Elvalid A., et al. Overview and principles of Internet traffic engineering. Internet-Draft: draft-ietf-tewg-principles-00.txt. – <https://tools.ietf.org/html/draft-ietf-tewg-principles-00>
136. Choi T., Yoon S., Chung H., et al. Design and implementation of traffic engineering server for a large-scale MPLS-based IP Network // LNCS. – 2002. – Vol. 2343. – P. 699–711.
137. Fortz B., Rexford J., Thorup M. Traffic engineering with traditional IP routing protocols // IEEE Communications Magazine. – 2002. – Vol. 40. – P. 118–124.
138. Zhang Y., Roughan M., Duffield N., Greenberg A. Fast accurate computation of large-scale IP traffic matrices from link loads // Proc. of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems. – 2003. – P. 206–217.
139. Wang N., Ho K., Pavlou G., et al. An overview of routing optimization for Internet traffic engineering // IEEE Communications Surveys & Tutorials. – 2008. – № 1. – P. 36–56.
140. Benson T., Akella A., Maltz D. Network traffic characteristics of data centers in the wild // Proc. of the Internet Measurement Conference. – 2010. – P. 267–280.
141. Benson T., Anand A., Akella A., Zhang M. MicroTE: Fine grained traffic engineering for data centers // Proc. of the Seventh Conference on Emerging Networking Experiments and Technologies. – 2011. – P. 1–12.
142. Rojas-Cessa R., Kaymak Y., Dong Z. Schemes for fast transmission of flows in data center networks // IEEE Communications Surveys & Tutorials. – 2015. – № 3. – P. 1391–1422.
143. Oikonomou K., Kogias D., Tzevelekas L., et al. Investigation of information dissemination design criteria in large-scale network environments // Proc. of the 13<sup>th</sup> Panhellenic Conference on Informatics. – 2009. – P. 163–167.
144. Openflow switch specification v1.0–v1.4. – <https://www.opennetworking.org/sdn-resources/onf-specifications>
145. Agarwal S., Kodialam M., Lakshman T. Traffic engineering in software defined networks // Proc. of the 32<sup>nd</sup> IEEE International Conference on Computer Communications. – 2013. – P. 2211–2219.
146. Akyildiz I., Lee A., Wang P., et al. A roadmap for traffic engineering in SDN-OpenFlow networks // Computer Networks. – 2014. – Vol. 71. – P. 1–30.
147. Lautari N., Smarandakis G., Oikonomou K., et al. Distributed Placement of Service Facilities in Large-Scale Networks // Proc. of IEEE INFOCOM, 2007. – P. 2144–2152.
148. Neumayer S., Zussman G., Cohen R., Modiano E. Assessing the vulnerability of the fiber infrastructure to disasters // IEEE/ACM Transactions on Networking. – 2011. – № 6. – P. 1610–1623.
149. Neumayer S., Modiano E. Network reliability with geographically correlated failures // Proc. of the 29<sup>th</sup> Conference on Information Communications. – P. 1658–1666.
150. Ran Y. Considerations and suggestions on improvement of communication network disaster countermeasures after the wenchuan earthquake // IEEE Communications Magazine. – 2011. – № 1. – P. 44–47.
151. Ogielski A., Cowie J. Internet routing behavior on 9/11 and in the following weeks. – <http://www.rennesys.com/tech/presentations/pdf/rennesys-030502-NRC-911.pdf>
152. Bellovin S., Gansner E. Using link cuts to attack Internet routing. – <http://www.cs.columbia.edu/smb/papers/reroute.pdf>
153. Wilson C. High altitude electromagnetic pulse (HEMP) and high power microwave (HPM) devices: threat assessment. – <http://www.fas.org/man/crs/RL32544.pdf>
154. Kvalbein A., Hansen A., Cicic T., et al. Fast IP network recovery using multiple routing configurations. – <http://folk.uio.no/amundk/infocom06.pdf>
155. Kini S., Ramasubramanian S., Kvalbein A., Hansen A. Fast recovery from dual link failures in IP networks. – <http://www2.engr.arizona.edu/~srini/papers/Srini-2009-INFOCOM.pdf>
156. Wang Y., Wang H., A. Mahimkar A., et al. R3: Resilient routing reconfiguration. – <http://cs-www.cs.yale.edu/homes/yry/projects/reinforce/r3-sigcomm10.pdf>
157. Zheng Q., Cao G., La Porta T., Swami A. Optimal recovery from large-scale failures in IP networks // Proc. of the IEEE 32<sup>nd</sup> International Conference on Distributed Computing Systems. – 2012. – P. 295–304.
158. Butler K., Farley T., McDaniel P., Rexford J. A survey of BGP security issues and solutions // Proc. of the IEEE. – 2010. – № 1. – P. 100–122.
159. Yen T., Oprea A., Onarlioglu K., et al. Beehive: Large-scale log analysis for detecting suspicious activity in enterprise networks // Proc. of the 29<sup>th</sup> Annual Computer Security Applications Conference. – 2013. – P. 199–208.
160. Kayes I., Iamnitchi A. A Survey on Privacy and Security in Online Social Networks. – <http://arxiv.org/abs/1504.03342>
161. Nissenbaum H. A contextual approach to privacy online // Daedalus. – 2011. – № 4. – P. 32–48.
162. <http://goo.gl/kHJF15>
163. <http://www.cnn.com/2012/08/30/tech/social-media/fighting-crime-social-media/>
164. Lotan G., Graeff E., Ananny M., et al. The arab spring – the revolutions were tweeted: information flows during the 2011 tunisian and egyptian revolutions // International Journal of Communication. – 2011. – № 5. – P. 1375–1405.
165. <http://goo.gl/9A6FR>
166. Mondal M., Viswanath B., Clement A., et al. Defending against large-scale crawls in online social networks // Proc. of the 8<sup>th</sup> International Conference on Emerging Networking Experiments and Technologies. – 2012. – P. 325–336.
167. Dantu R., Cangussu J., Patwardhan S. Fast worm containment using feedback control // IEEE Transactions on Dependable and Secure Computing. – 2007. – № 2. – P. 119–136.
168. Sellke S., Shroff N., Bagchi S. Modeling and automated containment of worms // IEEE Transactions on Dependable and Secure Computing. – 2008. – № 2. – P. 71–86.
169. Willinger W., Alderson D., Doyle J. Mathematics and the internet: a source of enormous confusion and great potential // Notices of the American Mathematical Society. 2009. – № 5. – P. 586–599.
170. Yu S., Gu G., Barnawi A., et al. Malware propagation in large-scale networks // IEEE Transactions on Knowledge & Data Engineering. – 2015. – № 1. – P. 170–179.

171. Zou C., Gong W., Towsley D., Gao L. The monitoring and early detection of internet worms // IEEE/ACM Transactions on Networking. – 2005. – № 5. – P. 961–974.

172. Invernizzi L., Miskovic S., Torres R., et al. Nazca: Detecting malware distribution in large-scale networks // Proc. of the IEEE Network and Distributed System Security Symposium. – 2014. – 16 p.

Стаття надійшла до редколегії 25.09.15

Скобелев В. Г., д-р фіз.-мат. наук, д-р техн. наук, проф.  
Інститут кібернетики імені В. М. Глушкова НАН України, Київ

## ПРОБЛЕМИ АНАЛІЗУ ТА СИНТЕЗУ ВЕЛИКОМАСШТАБНИХ МЕРЕЖ (ОГЛЯД)

У даній роботі міститься огляд стану досліджень деяких актуальних проблем аналізу та синтезу великомасштабних інформаційних мереж. Детально розглянуті найбільш часто використовувані методи, що засновані на аналізі лише топології досліджуваної мережі та призначені для виділення в мережі або спільнот які не перетинаються, або спільнот які можуть перетинатися. Охарактеризовано основні підходи, моделі і методи, використовувані в процесі аналізу соціальних мереж. Виділено деякі актуальні проблеми, що виникають у процесі проектування великомасштабних інформаційних мереж, і коротко розглянуті існуючі підходи до їх вирішення. Охарактеризовано основні моделі і методи, застосовувані для забезпечення безпеки великомасштабних інформаційних мереж.

**Ключові слова:** великомасштабні інформаційні мережі, аналіз, синтез, безпека, спільноти, онлайн-соціальні мережі.

Skobelev V. G., Dr. Phys. Math. Sci., Dr. Tech. Sci., Prof.  
V. M. Glushkov Institute of Cybernetics NAS of Ukraine, Kyiv

## PROBLEMS OF ANALYSIS AND SYNTHESIS OF LARGE-SCALE NETWORKS (SURVEY)

Given paper consists some survey of the state of the art of research for some actual problems of analysis and synthesis of large-scale information networks. There are considered in detail the most frequently used methods based only on the analysis of the topology of the network and intended for extracting non-overlapping communities, as well as of overlapping communities. Main approaches, models and methods used in the analysis of social networks are described. There are extracted some actual problems arising in the design of large-scale information networks, and existing approaches to solving them are briefly discussed. Basic models and methods used to ensure the security and safety of large-scale information networks are characterized.

**Key words:** large-scale information networks, analysis, synthesis, security, safety, communities, online social networks.

УДК 512.7+512.9, 688. 321

Р. В. Скуратовський, асп.  
інституту математики НАН України, Київ

## МОДЕРНІЗОВАНИЙ АЛГОРИТМ ПОЛІГА-ХЕЛМАНА, ШЕНКСА

Не викликає сумніву, те що більшість з методів криптоаналізу можуть бути перевтілені завдяки застосуванню паралельних алгоритмів та алгебраїчного апарата, зокрема теорії груп. Одним з таких методів є метод Шенкса розв'язання ПДЛ. Ціллю даної роботи є побудова алгоритма, що паралельно знаходить всі значення з таблиць малого кроку і великого кроку, також зробити цей пошук більш спрямованим і впорядкованим для всіх значень елементів таблиць, що дозволить застосувати метод блокового пошуку і дасть можливість розбиття на впорядковані підблоки, прискорить застосування метода індексації значень (чи хеш від значень). Методом є паралельна оптимізація і блочна паралельне поразрядне сортування, яка стала можливою завдяки швидким пересилкам в дуплексном режимі і математичні моделі алгоритму. В даній роботі запропоновано метод паралельного обчислення векторів координатами яких є значення таблиць BS. Також знайдена оптимальна довжина малого кроку і як наслідок і великого кроку для методу. В роботі запропоновано метод покращення алгоритма Шенкса шляхом його композиції з методом Поліга-Хелмана.

**Ключові слова:** Алгоритм, метод Шенкса, криптоаналіз.

**Вступ.** За часів виникнення методу Шенкса (1973р.) для розв'язання проблеми дискретного логарифма його ефективність [1] була незначною з причини невисокої направленості перебору, який він використовує, і наявності великої кількості чисел для пошуку рівності. Це був один з перших методів, більш швидкий, ніж метод прямого перебору. Ним займалися такі відомі криптографами, як Л. Адлеман і А. Стеін [2, 3]. Завдяки розвитку комп'ютерної техніки з'явилась можливість вдосконалення методу. В наш час, коли можлива паралельна обробка великих масивів інформації, його ефективність може зрости в стільки разів, скільки комп'ютерів ми застосовуємо. Це можливо саме завдяки придатності методу до розпаралелювання обчислень, які в ньому проводяться, але цьому не приділялось достатньої уваги відносно цього методу.

За основу для подальшої розробки взято класичний метод Шенкса [1]. Також порівняти затрати часу на пошук у невпорядкованому наборі масивів – таблицю з методу Шенкса і у впорядкованих блоках, які відповідають тим же таблицям хіба дещо іншого розміру.

**1. Постановка задачі.** Мета роботи – створити алгоритм розв'язання проблеми дискретного логарифму (ПДЛ) із застосуванням паралельних обчислень і знайти його оптимальні параметри та оцінки складності обчислень.

**Основні поняття.** В роботі ми прагнемо зробити обчислення таблиць паралельним. Зменшення часу на пошук рівності досягаємо впорядкування елементів таблиць, що досягається швидкими пересилками; потім використовуємо пошук по лінійно впорядкованим множинам і по частково впорядкованим множинам. Також необхідно порівняти затрати часу на пошук у невпорядкованому наборі масивів – таблицю з методу Шенкса і у впорядкованих блоках, які відповідають тим же таблицям, хіба дещо іншого розміру.

Розв'язання ПДЛ за Даніелем Шенксом [1, 2] спирається на пошук рівностей в наборах чисел. В циклічній мультиплікативній групі  $C_n$  обчислюються два рядка чисел:  $(a, ga, g^2a, g^3a, \dots, g^{m-1}a) \bmod n$  цей набір (таблицю) назвемо BS,  $(g^m, g^{2m}, g^{3m}, g^{4m}, \dots, g^{(m-1)m}) \bmod n$  цей набір (таблицю) назвемо GS.

Знаходять такі  $i$  та  $j$ , для яких  $g^i a = g^{jm}$ . Тоді в  $g^x = a$  маємо  $x = jm - i$ .



Остання рівність випливає з перетворень за модулем  $n$ :  $g^x = g^{jm-i} = g^{jm}(g^i)^{-1} = g^{jm}aa^{-1}(g^i)^{-1} = g^{jm}a(g^i a)^{-1} = g^{jm}a(g^{jm})^{-1} = a$  перехід до третьої рівності можливий, бо група комутативна. Потрібно застосувати методу Шенкса алгоритм з використанням паралельних обчислень і знайти параметри при яких він має найбільшу обчислювальну ефективність.

**2. Основні результати.** Якщо додаткові параметри дозволяють, то до даних елементів  $g$  та  $a$  можна застосувати ізоморфізм  $\varphi(g^k) = k \pmod{n}$  в групу  $\mathbb{Z}_n$ , де груповою операцією є додавання за модулем  $n$ , яке є більш швидким ніж множення. Всього таких ізоморфізмів буде  $\varphi(n)$ . Далі знайдемо елемент такого ж порядку в  $\mathbb{Z}_n^*$  але з меншою вагою Хемінга для цього візьмемо  $l_i: (l_i, n) = 1$ , тоді  $g^{l_i}$  має такий же порядок як і  $g$  але знайдемо  $i: wh(g^{l_i}) < wh(g)$ . Складність знаходження такого  $l$  за алгоритмом Евкліда є  $O(\ln)$ . Якщо в  $\mathbb{Z}_n^*$  вдалося знайти канонічний розклад числа  $\varphi(n): \varphi(n) = p_1^{j_1} p_2^{j_2} \dots p_m^{j_m}$  а для  $\mathbb{C}_n$  розклад  $n: n = p_1^{i_1} p_2^{i_2} \dots p_m^{i_m}$ , то легше і швидше за  $O(p_j n)$  перевірити тестові степені  $l_i$  на подільність  $l_i$  на  $p_j, 1 \leq j \leq m$  якщо її нема, то одразу маємо, що на  $(l_i, n) = 1$ . Для знаходження порядку  $g$  для всіх  $p_j, 1 \leq j \leq m$  перевіримо виконання умови  $g^{\varphi(n)/p_i} = e$  в  $\mathbb{Z}_n^*$  чи  $g^{n/p_i} = e$  в  $\mathbb{C}_n$ . Знаходження  $ord(g)$  в  $\mathbb{C}_n$  дасть можливість і в цьому методі своєрідно оптимізувати [3] величини шуканих таблиць.

В мультиплікативній групі для знаходження порядку елемента  $g$  – основи дискретного логарифма (щоб замінити його на елемент такого ж порядку) достатньо піднести  $g$  до степенів рівних можливим порядкам підгруп  $\mathbb{C}_n$ , які відомо з розкладу  $n = p_1^{i_1} p_2^{i_2} \dots p_m^{i_m}$ , тобто перевірити виконання  $g^{n/p_i} = e$  для всіх  $i$ .

Перед запуском алгоритму доцільно запустити наступний передпроцес: факторизувати  $n$  замість рівняння  $g^x = b \pmod{n}$  за Китайською теоремою про лишки (КТЛ) розв'язати систему з рівнянь  $h_i^x = b_i \pmod{p_i}$ , де  $g \equiv h_i \pmod{p_i}$ ,  $b \equiv b_i \pmod{p_i}$ . Потім за формулами з (КТЛ) можна отримати розв'язок рівняння  $g^x = b \pmod{n}$ . Якщо після цього вдалося знайти  $\varphi(p_j) = q_1^{j_1} q_2^{j_2} \dots q_k^{j_k}, 1 \leq j \leq m$ , то для кожного  $q_j$  шукаємо розв'язок у відповідній підгрупі

порядку  $\varphi(p_j)/q_j$ , тобто розв'язуємо рівняння:  $h_i^{q_j x} = b_i^{q_j} \pmod{p_j}$ .

Після виконання такого препроцесу це можливо зробити за  $O(\sqrt{\max p_i} \log^2 n / k)$ . Для знаходження елементів таблиці BS можна застосовувати метод паралельно-конвеєрного множення [4], зробивши перед цим обчислення базових степенів виду  $g^{n/k}, g^{2n/k}, \dots, g^{(k-1)n/k}$ , де  $k$  – кількість процесорів. Для швидкого обчислення набору базових степенів можна використати паралельний метод бінарного піднесення до степеня і адитивні ланцюжки [1, 4]. В результаті складність обчислення всіх таблиць BS  $O(\sqrt{p} \log^2 p / k)$ , яку далі буде покращено застосувавши спеціальний вибір елемента  $g$ .

**2.1. Модель методу Шенкса для  $k$  – процесорного кластера без впорядкування лишків.** Нехай маємо кластер з  $k$  нодами (Нода – незалежний обчислювальний вузол у мультипроцесорній синхронній обчислювальній системі типу кластер, може містити як один так і кілька процесорів). Позначимо:  $m$  – величина малого кроку (BS), в звичайному методі Шенкса  $m = \sqrt{n}$ , де  $n$  – порядок групи в якій вирішується ПДЛ. У BS розподілимо елементи рівномірно між нодами, тобто на кожному комп'ютері буде обчислено  $\frac{m}{k}$  елементів з таблиці, BS (тобто з таблиці яка відповідає малому кроку); всього буде  $\frac{n}{m}$  таблиць BS з  $m$  елементами в кожній і вони будуть рівномірно розподілені на  $k$  комп'ютерах. Розмірність таблиці GS якраз дорівнює  $\frac{n}{m}$ .

Тоді кількість піднесень до степеня в BS дорівнює  $\frac{m}{k}$ . Далі маємо сумарну кількість кроків-елементарних обчислень у таблиці GS і в кожному екземплярі таблиці BS:  $S = \frac{n}{m} + \frac{m}{k}$ .

Наша задача мінімізувати суму  $S$  за величиною  $m$ , тобто:  $\frac{n}{m} + \frac{m}{k} \rightarrow \min$ .

Для того, щоб вибрати оптимальне  $m$  подивимося на цю суму з точки зору нерівності Коші. Позначимо  $a = \frac{n}{m}$ ,  $b = \frac{m}{k}$ . Як відомо, для невід'ємних дійсних  $a, b$  виконується  $a^2 + b^2 \geq 2ab$ , чи  $a + b \geq 2\sqrt{ab}$ , а рівність досягається тільки при  $a = b$ . Отже найменше значення, яке сума може досягти це  $2\sqrt{ab}$ . Тоді в нерівності  $\frac{n}{m} + \frac{m}{k} \geq 2\sqrt{\frac{n}{k}}$  має місце рівність тоді і тільки тоді, коли  $\frac{n}{m} = \frac{m}{k}$ ,  $m^2 = nk$ ,  $m = \lceil \sqrt{nk} \rceil$ . Тоді загальна складність одночасного обчислення усіх таблиць:

$\frac{n}{\sqrt{nk}} + \frac{\sqrt{nk}}{k} = \frac{\sqrt{n}}{\sqrt{k}} + \frac{\sqrt{n}}{\sqrt{k}}$ . Позначимо  $l = \log n$ ,  $m' = \lceil \sqrt{nk} \rceil$  – величина нового BS. Тоді в цілому на створення таблиць з врахуванням тривалості піднесення до степеня треба  $O(m'^l / k) = O(\sqrt{nk}((\log^2 n) / k + \log k)) = O(\sqrt{n/k}(\log^2 n))$ ,  $k \ll n$ , тут  $\log k$  це час на генерування базової послідовності степенів.

Спочатку бінарно-паралельним методом згенеруємо базову послідовність степенів [12]  $\{g^{2^k}\}_1^s$ ,  $g^{2^s} > m > g^{2^{s-1}}$  і перешлемо її на кожну ноду. Далі на кожній з  $k$  нод згенеруємо послідовні степені з інтервалу  $[\frac{(j-1)m}{k}; \frac{jn}{k}]$ ,  $j \in 1, \dots, k$ , використовуючи паралельний метод адитивних ланцюжків. На це буде потрібно  $O(\frac{m}{kc})$ , оскільки всі  $m$  чисел ділимо на  $k$  підмножин і на кожній ноді є  $c$  процесорів.

Якщо спочатку паралельно генерувати бінарним методом послідовність  $1, g, g^2, g^4, \dots, g^{\lceil \log \sqrt{nk} \rceil}$  і  $g^{\lceil \log \sqrt{n} \rceil}, g^{2\lceil \log \sqrt{n} \rceil}, g^{4\lceil \log \sqrt{n} \rceil}, \dots, g^{\lceil \log^2 \sqrt{n/k} \rceil}$ , то на кожній ноді можна породжувати всі степені з BS всередині кожного діапазона (там їх лише  $m/k = \sqrt{nk}/k$ ) бінарно-паралельним методом піднесення до степені, використовуючи всі процесори і всі їхні ядра, що має нода. Нехай вона має  $c$  ядер в сукупності, тобто ширина платформи ноди рівна  $c$ , тоді висотою платформи (висотою бінарного дерева) є  $\frac{\sqrt{n/k}}{c}$ . Тобто найдовший адитивний ланцюг для обчислення довільної степені з BS не більше ніж  $\frac{\sqrt{n/k}}{c}$ . Отже загальна оцінка покращиться до  $O((\log^2 n)\sqrt{n/c^2k})$ . Також слід врахувати при розбитті на  $t$  діапазонів те, що перші  $t$  біт всередині кожного діапазона однакові, тому обчислювати їх кожного разу не потрібно, лишається частина числа довжиною  $l' = l - t = l - k$ . Отже, маємо  $O((l')^2 \sqrt{n/c^2k})$ . Застосування бінарно-паралельного методу піднесення до степеня є оптимальним також з точки зору паралельної архітектури, бо жоден процесор на протязі всієї процедури не простоє (окрім початкового етапа, де обчислюються малі степені, але цього неможливо уникнути в будь-якому разі).

Обґрунтовуючи оптимальність застосування паралельно бінарного методу піднесення в степінь важливо підкреслити не лише те, що тут використовується мінімальна кількість множень, а головне те, що використовується максимально можлива ширина обчислювальної платформи, тобто проводиться одночасно найбільша кількість множень, при цьому загальна кількість множень найменша, а саме  $\frac{m}{k}$ .

Зауважимо, що ми розділяли шуканий степінь на дві частини. Перша – префікс, який є постійним для всього інтервалу чисел (при розбитті на  $t$  діапазонів те, що перші  $t$  біт), що лежить на даному процесорі і друга – кінцева, яка точно обчислюється для кожного числа побудовою адитивних ланцюгів, що включають числа з усіх інтервалів для отримання мінімальної кількості множень в добутку рівному шуканому числу.

Назвемо префіксом префіксну частину кожного діапазону. Помітимо, що доцільно по адитивним ланцюгам рахувати тільки степені для частини показника, що йде після префіксу відповідного діапазону тобто після перших  $t$  біт.

1) Очевидно, що для обчислення кожного степеня необхідно принаймні 1 множення, яке не виконувалось для обчислення інших степенів.

2) Кожне множення в даному методі дає якийсь новий, раніше не обчислений степінь.

З 1) і 2), очевидно, слідує, що кількість множень в методі рівне мінімальній кількості множень тобто  $\frac{m}{k}$  ( $m = \lceil \sqrt{n} \rceil$ ), необхідному для обчислення отриманого набору степенів – в цьому сенсі метод оптимальний (безвідносно паралельності).

Він оптимальний також з точки зору паралельної архітектури, т.я. жоден процесор протягом всієї процедури не простоє (окрім початкового етапу, коли обчислюються малі степені, але від цього нікуди не подітися).

Висновок: малий крок  $m' = \lceil \sqrt{nk} \rceil$  став більшим за рахунок розподілення обчислень, що є подібним результату [3],

де  $m' = \lceil \sqrt{nE_k} \rceil = \lceil \sqrt{n\alpha(g)L} \rceil$  і відповідно  $\frac{E_k}{m'}$  (в позначеннях [3]  $\alpha(g)$  – оцінка порядку елемента  $g$ ) великих кроків.

**2.2. Модель з впорядкуванням і пересилками по блоках.** Спочатку запустимо передпроцес: якщо розв'язуємо ПДЛ за основою  $g$  в  $C_p$ , то знайдемо серед  $g^i$ ,  $1 < i < \log_2 p$  такий, що вага Хемінга [5]  $wt(g^i)$  у нього є найменшою.

Далі визначаємо,  $g$  через  $g^i$  і застосовуємо метод для  $g^i$  а потім знаходимо потрібну степінь для  $g$ . У випадку коли  $g$  має щільні серії з одиничок, то застосувати перехід до представлення у формі  $na_f$  числа [3, 6] за формулою виду  $g = 11111 = 100000 - 000001$ , при цьому будемо виділяти в числі блоки з одиниць для такого виду заміни. Лінійно впорядковані множини завжди є придатнішими для аналізу та порівняння, зокрема до них зручно застосувати метод блочного пошуку. Доцільно застосувати метод паралельного сортування чисел на  $k$  комп'ютерах:

Крок 1. Нехай  $m = \lceil \frac{n}{k} \rceil$ . Розподілимо по  $m$  чисел на кожній ноді\* (процесор). На кожній ноді робимо сортування чисел, що знаходяться на ньому.

Крок 2. Після обчислення кожною ногою табличок BS і GS лишки за модулем  $p$  розподілені нерівномірно на нодах. Треба розбити всі лишки на множини  $M_{i,i+1} = \{r: t_i < r < t_{i+1}, |M_{i,i+1}| = m\}$  (множину  $M_{i,i+1}$  назовемо  $i$ -им діапазоном  $-D_i$ ,  $t_i, t_{i+1}$  – границі інтервалу, що відповідає  $D_i$ ). Якщо на  $j$ -ій ноді лежить найбільша частина з множини  $M_{i,i+1}$ , то на цій же  $j$ -ій ноді і збиратимемо всі елементи з множини  $M_{i,i+1}$  (аналогічно на інших нодах). Тоді для збору  $M_{i,i+1}$  на одній ноді, наприклад на  $j$ -му, кількість пересилок з кожного комп'ютера буде якомога менше.

Знаходимо таке  $t_1$ , що кількість всіх чисел (на всіх нодах), які не більші ніж  $t_1$  є рівно  $m$ . Це робиться за допомогою **двійкового пошуку**. Тобто вибір границь першого діапазону  $[0; t_1)$  робимо вибираючи  $t_1$  половинним діленням до досягнення потрібної кількості  $m$ . Оскільки аргумент дискретний, і лежить на відрізку довжини  $n$ , то пошук займе  $1 + \log_2 n$  часу. Аналогічно знаходимо всі  $t_i$ ,  $1 < i \leq k$ .

**2.3. Дводольна модель пересилок.** В процесі пересилки кожна нода приймає і передає числа але тільки від одної ноди чи тільки одній ноді. Побудуємо модель пересилок у вигляді дводольного графа, для цього продублюємо всі комп'ютери в вигляді другої долі з усіх комп'ютерів але активними в цій долі є ті, які знаходяться в стані прийому. А в першій (верхній) долі є тіж самі ноди, але активними з них є лише ті, які перебувають у стані відправки. Іншими словами побудуємо модель у вигляді дводольного графа  $K_{k,k}$ , де  $k$  – кількість нод. В другій долі активні ті, які приймають, тобто в них входять стрілки; перша доля – ті, які передають, з них виходять стрілки.

**Означення 1.** Множину чисел, що лежить на  $i$ -ому ноді і належить  $j$ -ому діапазону позначимо як  $A_{ij}$ . Матрицю  $(A_{ij})_{1 \leq i, j \leq k}$  елементами якої є кількості  $A_{ij}$  назовемо матрицею пересилок  $A$ .

З'єднаємо  $i$ - нод зверху та  $j$ - нод знизу на якому збираємо числа з  $D_j$ , якщо  $A_{ij} \neq \emptyset$ . Знайдемо паросполучення долей і зробимо сеанс пересилки, в якому на  $j$ -ий ноді знизу висилаються числа з  $i_j$ - нод зверху, де всі  $i_j$  різні, вони як раз і шукаються паросполученням. На  $j$ - нод знизу приходять числа з  $i$ -го комп'ютера верхньої долі з діапазону  $D_j$  із  $M_{j-1,j}$ , тобто вони з  $A_{ij}$ , на рис. 1 X, Y, Z, W – ноди з верхньої долі а A, B, C, V – ноди з нижньої долі, де збираються лишки з відповідних діапазонів.

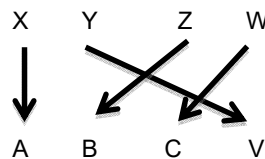


Рис. 1. Паросполучення-пересилка між нодами

Тоді для довільного  $i$  з  $0 < i < k+1$ ,  $\sum_{j=1}^k \#A_{ij} = m_0$ , при  $n:k$  або  $\sum_{j=1}^k \#A_{ij} = m_0 + 1$ , якщо  $n \neq sk$ , де  $\#$  позначає потужність множини. Також для довільного  $j$  з  $0 < j < k+1$ ,  $\sum_{i=1}^k \#A_{ij} = m_0$ , при  $n:k$ , або  $\sum_{i=1}^k \#A_{ij} = m_0 + 1$ , якщо  $n \neq sk$ .

$$\sum_{j=1}^k \#A_{ij} = \sum_{i=1}^k \#A_{ij} = m_0 \quad (1)$$

для кожного  $i: 1 \leq i \leq k$  в лівій частині і в правій частині для кожного  $j: 1 \leq j \leq k$ ,  $m_0 > 0$ .

За час сеансу пересилки на кожній ноді передається наступна множина чисел:

$$X = \min \{ \#A_{i_1}, \#A_{i_2}, \dots, \#A_{i_k} \}, i_j \neq j$$

Тобто на кожний комп'ютер надійшло в сукупності  $X$  чисел з інших комп'ютерів.

Для даного  $k$  будемо дводольний граф  $K_{k,k}$ . Для цього введемо на вершинах  $v$  з  $K_{k,k}$  відношення суміжності  $\omega(v_i, v_j) = 1$  тоді і тільки тоді коли  $A_{ij} \neq \emptyset$  тобто з'єднаємо  $i$ -у ноду зверху з  $j$ -ю ногою знизу ребром, якщо  $A_{ij} \neq \emptyset$ .

Назовемо матрицю бістохастичною, якщо вона квадратна, всі її елементи невід'ємні, а сума елементів в кожному рядку і в кожному стовпчику рівна одному і тому ж додатному числу.

**Лема 1.** Паросполучення долей (або трансверсаль в матриці суміжності для  $K_{k,k}$ ) існує як тільки виконується умова (1), яка характеризує відношення суміжності ребер дводольного графа.

Дійсно довільні  $k$  нод має переслати  $km_0$  чисел, причому кожна передає і приймає рівно по  $m_0$  чисел, тому прийняти за сеанс пересилки ці  $km_0$  чисел можуть не менше ніж  $k$  нод, ця умова виконується для кожного  $1 \leq j \leq k$ , тому умова  $\#A' \leq \#f(A')$  з Наслідку 10.4.1 [7] виконана, де  $f(A')$  – множина вершин інцидентних ребрам, що виходять з множини вершин  $A'$ .

Позначимо  $\omega(v_i, v_j)$  – відношення суміжності вершин в графі, матрицю суміжності вершин дводольного графа як  $A$ .

**Теорема 1.** Якщо в матриці суміжності  $A$  замінити 1-ки в клітинах, де  $\omega(v_i, v_j) = 1$  на значення  $A_{ij}$ , то отримаємо матрицю пересилок  $\tilde{A}$ , з нод нижньої долі в ноди верхньої долі, між якими існує трансверсаль без 0-вих елементів матриці.

Доведення. Оскільки для матриці  $\tilde{A}$  виконується умова (1), то можемо поділивши кожен її стовпець отримаємо бістохастичну в традиційному розумінні матрицю для якої застосовна згадана в лемі умова і умови лемі виконані, то трансверсаль в матриці пересилок  $\tilde{A}$ , що відповідає пересилкам між комп'ютерами існує, тому шляхом помноження кожного коефіцієнта матриці  $A$  на суму пересилок по рядку  $m_0$  (чи можна  $X$ ), ми встановлюємо відповідність між матрицями суміжності і матрицею пересилок обернену відповідність встановлюємо шляхом ділення на  $m_0$ . Отже отримали взаємозначну відповідність між матрицями пересилок і матрицею суміжності. Тобто це впливає з Лемі і доведено в [9].

**Наслідок.** Поки на кожній ноді існують числа, що потрібно переслати, одночасна пересилка з усіх нод можлива навіть без використання дуплексного режиму обміну даними. Наявність же такого режиму тільки зменшує час на пересилку. Опишемо основні суть дій алгоритму пересилок:

1. Знайти перше паросполучення з максимально можливим мінімальним ребром.

Перше паросполучення доцільно вибрати таким:  $X = \max_T \{ \min \{ \#A_{i_1}, \#A_{i_2}, \dots, \#A_{i_k} \} \}$ , де  $T$  – множина всіх трансверсалей, щоб кожен комп'ютер зробив формальну пересилку самому собі якомога більшої кількості чисел. Тому час на цю пересилку і пошук такої комбінації ребер буде виключено з затраченого часу на пересилки, саме тому бажано щоб їх час був найбільшим. Тобто це мінімум з максимально можливих пересилок, що будуть зроблені за петлею. Для того щоб знайти таке паросполучення з максимально можливим мінімальним ребром слід [9] використати динамічний алгоритм приєднання ребер, у порядку спадання їх ваги, у випадку неможливості реалізації часткового паросполучення з більшою кількістю ребер. Вийде оцінка –  $O(k)$  кількості фаз збільшення потужності множини можливих в паросполученні ребер. Кожна фаза вимагає  $O(k^2)$  переглядів різних ребер а є  $k$  фаз [9], тому всього  $O(k^3)$ , що менше за  $O(k^5)$  – час пошуку повним перебором з застосуванням алгоритму Куна. Але якщо при цьому використати алгоритм Хопкрофта-Карпа [9], то всього  $O(k^2 \sqrt{k} \log k)$ . Хоча це не суттєво, бо  $k \ll n$ .

Далі можна шукати паросполучення в дводольному графі вже не з  $Max$  ребром серед  $Min$  ваг ребер з різних паросполучень. Оскільки кількість пересланих чисел одна і та сама, то піклуватись про максимальність мінімального ребра вже не потрібно. Швидше знайти доповнений ланцюг з ребер, що буде новим паросполученням модифікувавши старе паросполучення, що було зменшене на кілька ребер (зазвичай на одне ребро) а це можна зробити методом пошуку "в ширину" за  $O(k^2)$  він описаний нижче в адаптованому вигляді.

Після пересилки шукаємо нове паросполучення знову з урахуванням зменшених множин  $A_j$ , позначимо їх  $A_j^1$ ,  $|A_j^1| = |A_j| - X$ . Воно знову знайдеться згідно з лемою, бо матриця знову володіє властивістю (1), бо сума по всім рядкам і стовпчикам цієї матриці однакова і не рівна 0. Дійсно для довільного  $i$  з  $0 < i < k+1$ ,  $\sum_{j=1}^k \#A_{ij} = m_0 - X$ , при  $n:k$  або  $\sum_{j=1}^k \#A_{ij} = m_0 + 1 - X$ , якщо  $n \neq sk$ . Також для довільного  $j$  з  $0 < j < k+1$ ,  $\sum_{i=1}^k \#A_{ij} = m_0 - X$ , при  $n:k$ , або  $\sum_{i=1}^k \#A_{ij} = m_0 + 1 - X$ , якщо  $n \neq sk$ . Помітимо, що умова (1) після цього не порушена і матриця суміжності лишилась бістохастичною, бо від кожної суми по рядку і стовпцю відняли однакове  $X$ . Далі переходимо до нового сеансу пересилки.

2. Знайти нове паросполучення методом подовжуючого ланцюга. Подовжуючий ланцюг ефективно будується методом пошуку "в ширину" модифікуючи старе паросполучення.

3. Після завершення всіх пересилок числа будуть розподілені по відповідним діапазонам. Їх лишилось відсортувати методом порозрядного сортування чи індексувати на кожному комп'ютері, який містить рівно один діапазон  $D_i$ , для такого індексування можна використати хешування.

4. Знайти рівність шляхом побітового порівняння, що буде найшвидшим у даному випадку. Можна застосувати алгоритм [10] оптимального розбиття кожної таблиці індексів на блоки при якому середнє число порівнянь необхідних для пошуку елемента в індексі є мінімальним. Якщо кількість елементів індекса рівна  $m$ , то оптимальний розмір блока буде  $\sqrt{m}$ . При цьому число блоків і середня кількість порівнянь теж  $\sqrt{m}$ . Кінець.

Знайдемо оцінку середньої кількості порівнянь біт чисел з різних таблиць для знаходження рівності. Для цього покажемо, що якщо  $g$  – породжуючий елемент мультиплікативної групи поля  $F_q$ , то двійкові знаки чисел з BS будуть розподілені рівномірно-дискретно. Введемо випадкову величину –  $X$  як кількість послідовно порівняних знаків  $\alpha_i, \beta_j$  чисел до знаходження першої пари різних знаків, тобто  $\alpha_j \neq \beta_j, 1 \leq i, j \leq n$ . Нехай маємо  $k = 2^t, t \in \mathbb{Z}$ , тоді в таблиці номер  $k$  будуть зібрані числа які починаються на  $t$  одиниць. Якщо  $g$  – твірний групи  $F_q^*$ , то усі елементи породжені в процесі експоненціювання  $g$  і зведення за модулем  $q$  взаємно однозначно відповідають елементам з  $F_q^*$ , бо лишків за модулем  $q$  є стільки ж як елементів в  $F_q^*$ . Тому у останньому інтервалі будуть елементи  $h: [(k-1)/k]q < h \leq q$  з префіксом з  $t$  одиниць а потім мають таке ж заповнення знаками 0 і 1 як і числа у двійковій с.ч., які мають  $[\log_2 m] - t$  розрядів, тобто на початку інтервалу більше 0-ів а у симетричних їм числам відносно середини інтервалу настільки ж більше 1-ць.

Підрахуємо математичне сподівання знаків, які можуть відрізнятися. Спочатку порівнюємо перші знаки, якщо вони співпали (ймовірність чого рівна  $p_1 = \frac{1}{2}$ ), то порівнюємо другі знаки ймовірність одночасного співпадіння двох знаків в обох числах є  $p_2 = \frac{1}{4}$ . Тоді  $M(X) = 1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{4} + 3 \cdot \frac{1}{8} + \dots + (\log p_i^{-1}) p_i + \dots$ .

Помітимо, що цей ряд є сумою наступних геометричних прогресій:

$$1 \cdot \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \dots = 1 \text{ – така є одна;}$$

$$\frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \dots = \frac{1}{2} \text{ – таких є дві, але одну з них вже врахували;}$$

$$\frac{1}{8} + \frac{1}{16} + \dots = \frac{1}{4} \text{ – таких є 3 але 2 з них вже порахували.}$$

І т. д. в сумі матимемо 2. Отже математичне сподівання числа порівнянь розрядів дорівнює 2.

**2.4. Принцип пошуку доповненого ланцюга.** Мета пошуку доповненого ланцюжка – це знайти нову трансверсаль після занулення ребра з найменшою вагою. Позначимо перше побудоване паросполучення як  $r_0$ ,  $r_1$  – часткове паросполучення, що утворилось з  $r_0$  після вилучення ребра між  $v_0$  і  $u_0$  з мінімальною вагою в результаті пересилки,  $u_i$  – вершини нижньої долі,  $v_i$  – вершини верхньої долі. Нехай  $u_0$ ,  $v_0$  – вершини, що випала з паросполучення  $r_0$  після чергової пересилки між нодами, яким відповідають вершини. Добудуємо  $r_1$  до повного паросполучення починаючи з вершини  $u_0$  за алгоритмом:

1) з можливих ребер вибираємо всі, які виходять з  $u_0$  далі рухаємось кожному з них (пошук в ширину) потрапляємо в якусь  $u_{i_1} \dots u_{i_j}$ , опишемо такий рух по одному з них (по іншим аналогічно), нехай потрапляємо в якусь  $v_{i_1}$ .

2) переходимо з  $v_{i_1}$  в  $u_{i_1}$  по ребру з  $r_1$ , яке є ребром з  $r_0$  і видаляємо це ребро з  $r_1$ .

3) беремо ребра, що виходить з  $v_{i_1}$  і повторюємо процедуру описану в пункті 1.

4) якщо потрапили у вершину, якій інцидентне ребро з  $r_1$  утворене в п. 1 то цей ланцюжок перестаємо розглядати, розглядаємо інші ланцюжки які таким же чином почали будувати з  $v_0$ .

5) як тільки потрапили у  $v_0$  зупиняємо алгоритм і включаємо пройдені ребра з нижньої долі у верхню в нове паросполучення а решту доповнюємо ребрами з  $r_0$ .

6) Для пошуку подовженого ланцюга ефективно використати метод пошуку в ширину або метод пошуку в глибину. Тому складність пошуку подовженого ланцюжка є  $O(e)$ , де  $e$  – кількість наявних ребер. Ці методи доцільні, бо дозволяють добудувати редукований ланцюжок на 1 ребро щоб отримати нове паросполучення. Хоча можна побудувати нове паросполучення за методом Хопкрофта-Карпа.

**Коректність.** Нове  $r_1$  обов'язково буде знайдене, воно існує за лемою, а тому його ребра мають входити і в  $u_0$  і в кожен вершину в яку ми потрапили на кроці 1 з  $u_0$ , тому запропонований пошук початий з вершини  $u_0$  пройде і по цій трансверсалі ланцюг не був пропущеним. Довжина знайденого ланцюжка не може перевищити  $k$ , бо це довжина трансверсалі. Оскільки на кроці 1 ми могли потрапити як максимум в  $k$  вершин і при кожному повторі кроку 1 теж як максимум в  $k$  вершин, то сукупна кількість кроків по ребрам рівна  $k^2$ . Отже оцінка кількості пройдених ребер при пошуку  $O(k^2)$ .

Після завершення всіх пересилок числа будуть розподілені по відповідним діапазнам. Їх лишилось відсортувати методом половинного ділення чи індексувати на кожному комп'ютері, який містить рівно один діапазон  $D_i$ . Для цього ефективно використати порозрядне сортування, бо числа мають однакову довжину і довжина чисел зростає разом з величиною масиву  $O(\sqrt{n} \log n / k)$ . Як альтернативний алгоритм можна застосувати алгоритм оптимального розбиття [10] кожної таблиці індексів на блоки при якому середнє число порівнянь необхідних для пошуку елемента в індексі є мінімальним. Якщо кількість елементів індекса рівна  $m$ , то оптимальний розмір блока буде  $\sqrt{m}$ . При цьому число блоків і середня кількість порівнянь теж  $\sqrt{m}$ .

Можливо також зробити пошук паросполучення  $r_1'$  заново, замість пошуку доповнючого ланцюга за  $O(k^2)$  до  $r_1$ , алгоритмом пошуку Хопкрофта-Карпа складність якого  $O(\sqrt{kk^2})$ , чи алгоритм Куна  $O(k^3)$ .

Позначимо  $k$  нодів, що відповідають  $k$  діапазнам як  $n_1, n_2, \dots, n_k$ .

Задамо послідовність пересилок так: перша пересилка множини  $T_1 = (A_{11}, A_{22}, \dots, A_{kk})$  друга  $T_2 = (A_{21}, A_{32}, \dots, A_{k,k-1}, A_{1k})$  містить піддіагональні елементи і доповнючий елемент  $A_{1,k}$  далі  $T_3 = (A_{31}, A_{42}, \dots, A_{k,k-2}, A_{1k-1}, A_{2,k})$  і т. д. остання  $T_k = (A_{k1}, A_{12}, \dots, A_{k-2,k-1}, A_{k-1,k})$ . За кожну пересилку  $T$  будемо передавати усі множини з  $T$  в тому числі  $X = \max_T \{ \#A_{i_1,1}, \#A_{i_2,2}, \dots, \#A_{i_k,k} \}$  при цьому поки не закінчиться пересилка  $X$  інші ноди очікують але час на обчислення  $T_i$  не витрачається.

Якщо кількість нод занадто велика а швидкість пересилок не менша ніж по оптоволоконним провідникам, то доцільно виконувати направлені пересилки без розрахунку повного паросполучення. В такому випадку використовуємо наявні трансверсалі у матриці пересилок по детермінованому графіку.

Ефективним є комбіноване застосування побудованого алгоритму з модифікованим Стінсоном алгоритмом Шенкса, що використовує розщиплювальну систему множин (SS), описану в [11, 5], складність якого, в залежності від версії ал-

горитму є або  $O\left(C_{\lceil \log_2 n \rceil}^{t/2}\right)$  або  $O\left(NC_{\lceil \log_2 n \rceil}^{t/2}\right)$ , де  $t$  – вага Хемінга дискретного логарифма елемента  $g$ ,  $N$  – кількість блоків  $B_i$  в  $(N; m, t)$ -SS. Побудований алгоритм доцільно застосовувати якщо  $k \ll |B_i|$ . В результаті застосування нашого алгоритму складність стане  $O\left(C_{\lceil \log_2 n \rceil}^{t/2} / k\right)$  або  $O\left(NC_{\lceil \log_2 n \rceil}^{t/2} / k\right)$  відповідно, необхідні обчислювальні ресурси  $O\left(kC_{\lceil \log_2 n \rceil}^{t/2}\right)$ . Також доцільно здійснювати паралельну обробку даних для різних вірогідних  $t$ , помітимо, що особливо ефективним метод SS є при  $t: t/2 < [0,45 \log_2 n]$ , бо значення  $C_n^{t/2}$  тим менше чим більше різниця  $n - t/2$ .

**Аналіз алгоритму і коректність роботи алгоритму.** Після розбиття на діапазони і завершення їх наповнення однакові числа можуть існувати лише в межах одного діапазону, саме тому пошук пари співпадаючих чисел робиться не з множини  $\#M = C_k^2 \sqrt{n/k} \sqrt{n/k} = C_k^2 n/k$  а за допомогою сортування  $n/k$  елементів  $O(\sqrt{n} \log n / k)$  і паралельної перевірки наявності рівності серед сусідніх елементів. Звідси економія кількості операцій на пошук збігів дорівнює  $C_k^2(n/k) - (n/k)$ . А одне порозрядне порівняння [10] може в гіршому випадку зайняти  $\log_2 n$ .

Дійсно при методі без впорядкованих діапазонів треба кожне число  $z$ , що обчислено на ноді, якій відповідає діапазон  $D_i$  порівнювати з усіма іншими числами з усіх діапазонів в тому числі з самого  $D_i$ .

**Складність методу впорядкованих блоків з пересилками.** Позначимо:  $m$  – кількість чисел,  $l$  – довжина числа. Зрозуміло, що  $l = \log n$ ,  $m = \lceil \sqrt{\max p_i} \rceil$ .

На створення таблиць треба  $O(m l^2 / k) = O(\sqrt{n} (\log n \log g) / k)$ ,  $wt(g) < wt(n)$  елементарних операцій, більш точно це  $O(m l \cdot wt(g) / k)$ , ця ж дія з використанням паралельного множення на основі багатосмугових матричних операцій займе  $O(m l \cdot \log(\log(wt(g)))) / k$ , де  $\log n$  – час на піднесення до степеня. Порозрядне сортування:  $O(m l / k) = O(\sqrt{n} \log n / k)$  елементарних операцій.

Пересилка:  $O(m l / k) = O(\sqrt{\max p_i} \log n / k)$ . Тут слід врахувати, що максимальна кількість пересилок становить  $m^2$ , бо маємо  $m$  при кожному проходженні в гіршому випадку шлемо  $m - 1$ . Сумарна тривалість сеансу –  $m$  (бо з кожної ноди пересилається рівно  $m$  чисел).

Всього  $O(\sqrt{\max p_i} \log^2 n / k)$ . Порівняно з варіантом Стінсона [11,6] цього алгоритму  $O((N)C_{(l+1)/2}^{t/2})$ ,  $l = \lceil \log n \rceil$ ,  $0 < t < l$ ,  $t, l \in \mathbb{N}$  є кращим для багатьох значень параметрів  $l$  і  $t$ . Таким чином цей варіант методу Шенкса є дещо ефективнішим по складності обчислень ніж той, що описано у [6, 12], хоча теж вимагає великих витрат пам'яті, що є пропорційною до  $\sqrt{n/2}$ , але вже не вимагає покрокового порівняння координат, що здійснюється по всім точкам, що зберігаються в пам'яті. Отже, функція складності роботи алгоритму  $f(n, k)$  обернено пропорційно залежить від  $k$  і при фіксованих значеннях параметрів  $n, t = wt(g)$  має вигляд  $f(n, k) = C/k$ . Стосовно реальної ширини обчислювальної платформи тобто  $k$ , то сучасна відео карта може мати більше ніж 2000 (так для NVIDIA GeForce GTX Titan  $k=2668$ ) процесорних ядер, тому  $k > 2^{11}$  навіть при використанні однієї такої відео карти а китайський кластер [13] Tianhe-2 (MilkyWay-2) – TH-IVB-FEP має  $k = 3\,120\,000$  нод.

**Складність методу невпорядкованих блоків без пересилок.**

Тут, як доведено, оптимальним є  $m = \lceil \sqrt{\max p_i k} \rceil$  – довжина малого кроку.

На створення таблиць треба  $O(m l^2 / k) = O(\sqrt{nk} (\log n \log g) / k + \log k) = O(\sqrt{nk} (\log^2 n / k))$ ,  $k \ll n$ ,  $wt(g) < wt(n)$  множень. Також врахуємо порозрядне сортування яке має складнішу оцінку:  $O(m l / k) = O(\sqrt{nk} \cdot \log n / k)$ .

Порівнюючи складності цих методів  $O(m \cdot l^2 / k) + O(m l / k)$  і  $O(\lceil \sqrt{\max p_i} \rceil l^2 / k) + O(k^2)$ , де  $l = \log n$ ,  $m = \lceil \sqrt{\max p_i k} \rceil$  домножимо на  $k$  і маємо  $O(m \cdot l^2) + O(m l)$  і  $O(\lceil \sqrt{\max p_i} \rceil l^2) + O(k^3)$ . Для спрощення порівняння  $O(\lceil \sqrt{\max p_i k} \rceil \cdot l) + O(\lceil \sqrt{\max p_i} \rceil l) < O(\lceil \sqrt{\max p_i} \rceil l^2) + O(k^3)$  можна оцінити ліву частину знизу як  $O(\lceil \sqrt{\max p_i k} \rceil \cdot l^2) + O(\lceil \sqrt{\max p_i} \rceil l) > O(\lceil \sqrt{\max p_i} \rceil \cdot l^2) + O(\lceil \sqrt{\max p_i} \rceil l)$ , підставивши цю оцінку у нерівність отримуємо шукану умову  $O(\lceil \sqrt{\max p_i} \rceil l) > O(k^3)$ , яка виконається при  $\lceil \sqrt{\max p_i} \rceil l > (k^3)$  або  $\max p_i \log^2 n > k^6$ . Тобто для того, щоб другий метод став більш витратним треба щоб кількість нод перевищила  $\sqrt[6]{\max p_i \log^2 n}$ .

**Висновок.** Метод паралельного обчислення впорядкованих блоків Шенкса має в  $\sqrt{k}$  раз менше операцій ніж його аналог з невпорядкованими блоками. Отже збільшуючи число нод можна значно збільшити ефективність. Порівняно з паралельним алгоритмом, що застосовує не менш ніж  $\left(n^{2/3} : (\log n)^{1/3}\right)$  процесорів, при однаковому часі  $t'$  доступу до спільної пам'яті [12] і виконується не менше ніж за  $O\left(n^{1/3} (\log n)^{4/3}\right)$ , даний алгоритм виконується не більше ніж за

$O(\sqrt{nk} \cdot \log n / k)$ , тобто при  $k = \left(n^{2/3} \cdot (\log n)^{1/3}\right)$  дає краще значення –  $O(n^{1/6} (\log n)^{2/3})$ . Крім того він не вимагає одночасного, постійного доступу процесорів до спільної пам'яті і не змінює закон оцінки складності і для  $k < \left(n^{2/3} \cdot (\log n)^{1/3}\right)$ .

Перспективою продовження є комбінування цього методу з алгоритмом Поліга – Хелмана. Доцільно також розглянути модель пересилок на декілька комп'ютерів одночасно з однієї ноди в дуплексному режимі з виконанням умови (1).

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Василенко О. Н. Теоретико-числовые алгоритмы в криптографии. – М.: МЦНМО, 2003. – 328 с.
2. Adleman L. A subexponential algorithm for the discrete logarithm problem with applications to cryptography // Proc. 20th Ann. IEEE Symp. Found. Comput. Sci. 1979. P. 55–60.
3. Stein A., Teske E., Optimized baby step-giant step methods, Journal of the Ramanujan Mathematical Society 20, 2005, CRC PRESS, no. 1, 1–32.
4. Задирака В. К., Олексюк О. А. Компьютерная арифметика многоарядных чисел. Тернопіль, Підручники і посібники. 2003., с. 247.
5. Coron J. S. A new baby – step giant – step algorithm and some applications to cryptanalysis. Universit'e de Luxembourg // Proceedings of the 7th international conference on Cryptographic hardware and embedded system. P. 47–60.
6. Бессалов А. В., Телиженко А. Б. Криптосистемы на эллиптических кривых. – Киев: 2004, 203 с.
7. Оре О. Теория графов. – 2-е изд. – М.: Наука, 1980, 336 с.
8. Костюкова. Н. И. Графы и их применение. Комбинаторные алгоритмы для программистов. Москва. Издательство: Интернет-университет информационных технологий ИНТУИТ.ру 2007. С. 231.
9. Hand Book of Graph theory edited by Kenneth H. Rosen. Discrete Mathematics and its application. CRC Press., London, P. 1139.
10. Цегелик Г. Г. Методы автоматической обработки информации // Львов. Вища школа. 1981., с. 132.
11. Stinson D. R. Some baby step giant-step algorithm for low hamming weight discrete logarithm problem. Mathematics of Computation, 2002. 71(237), P. 379–391.
12. Michael J. Wiener. The Full Cost of Cryptanalytic Attacks. Journal of Cryptology 2004, vol. 17, no. 2, P. 105–124.
13. Т. Кормен, Ч. Лейзерсон, Р. Ривест. К. Штайн Алгоритмы. Построение и анализ. Издание 3-е. 2013. – 1324 с. 13. J. M. Pollard Kangaroos, Monopoly and Discrete Logarithms. J. Cryptology, 2000, 13: P. 437–447.
14. <http://top500.org/lists/2014/11/>

Стаття надійшла до редколегії 03.03.15

Скуратовський Р. В., асп.  
Института математики НАН України, Київ

### МОДЕРНИЗИВАННИЙ АЛГОРИТМ ПОЛИГА-ХЕЛМАНА, ШЕНКСА

Без сомнения многие криптоаналитические методы могут быть переоплощены благодаря параллельным алгоритмам и применению алгебраического аппарата. Одним из них является метод Шэнкса решения ПДЛ. Целью работы является построение алгоритма позволяющего параллельно находить все значения из таблиц малого и большого шага, также сделать этот поиск более направленным и упорядочить все значения элементов таблиц, что позволит применить метод блочного поиска даст возможность разбиения на упорядоченные подблоки, ускорит применение метода индексации значений (или хэш от значений). Методом является параллельная оптимизация и блочная параллельная поразрядная сортировка. В данной работе предложен метод параллельного вычисления векторов, координатами которых являются значения таблицы BS. Также найдена оптимальная длина малого шага и как следствие и большого шага для метода. В работе предложено метод улучшения алгоритма Шэнкса путем его композиции с методом Полига-Хелмана.

**Ключевые слова:** алгоритм, метод Шенкса, криптоанализ.

Skuratovskii R. V., PhD Stud.  
Institute of mathematics NAN of Ukraine, Kiev

### MODERN ALGORITHM OF POHLIG-HELLMAN AND SHANKS

There is no any doubts that most of cryptanalysis methods can be recreated owing to using parallel algorithms and algebraic apparatus in particular theory of groups. One of them is method of Shanks of solving discrete logarithms problem. The main goal of this article is to construct algorithm, which affording calculate all values of giant and baby steps tables in parallel mode also allowing to do this search for values of table odered and oriented. It enables to apply method of blocs searching. It makes acceleration in using of indexation of values method or calculating hex function. Method of solving problem is parallel optimization and blocs bitwise sorting. In given work method of parallel counting of vectors wich corresponds to tableaux are suggested. Also optimal length of baby and giant steps were found for this method. In given work method of improvement of Shanks algorithm via using algorithm of Pohlig-Hellman was suggested.

**Key words:** algorithm, method of Shanks, cryptanalysis.

УДК 519.7: 621.8

R. M. Trokhymchuk, Ph.D. Physics and Mathematics, Associate Professor  
K. E. Kozlov, master's degree in Cybernetics  
Taras Shevchenko National University of Kyiv

### APPLICATION OF MODULAR NEURAL NETWORKS FOR ANALYSIS AND PREDICTION OF SOCIO-ECONOMICAL AND ECOLOGICAL INDICATORS OF THE REGION

The objective of this work is the use of modular neural networks for analysis and forecasting of environmental indicators and their impact on the socio-economic condition of the region, as well as the comparison of the results of the model used with the results of other methods of data analysis (methods of mathematical modeling and mathematical statistics).

**Key words:** artificial neural network (ANN), modular neural network, municipal solid waste (MSW), forecasting model

**Introduction.** Presently, the problem of analysis and forecasting of various environmental indicators for particular areas is extremely important. The solution to this problem provides an opportunity to plan efficiently and reasonably and to manage optimally and effectively the process of collecting, transporting and recycling or disposal of municipal solid waste (MSW). In particular, the relationship between socio-economic indicators and the composition and volume of MSW in a specific area is very important to research. Seasonal fluctuations in the values of parameters and a large number of factors affecting this is the feature that greatly complicates the construction and use of appropriate formal models for these processes.

Traditionally, the classical methods of mathematical statistics and systems analysis, expert systems, fuzzy models, etc. are used to solve this problem and similar ones [1–3]. Artificial neural networks are a powerful and effective tool for this

analysis provided a small amount of training samples and the lack of a clear relationship between the input and output parameters. Application of an artificial neural network (ANN) to predict the characteristics of the waste according to the changing socio-economic factors described in the paper [4] cannot be called successful, as it uses a very simple small-sized ANN-model. Therefore, the resulting forecast differs little from the results of approximation of the input data by classical methods of mathematical statistics and numerical methods, as a result, the analyzed (extrapolated) data increase quickly and without bound outside the sample.

In this paper, the study of above-mentioned dependencies is done by means of the modular ANN [5, 6]. Modular neural networks are a new generation of artificial neural networks. Modular ANN is a very promising model, which allow during the training stage to combine different traditional architecture and algorithms for training neural networks in order to optimize and improve the effectiveness of the model, taking into account the adaptation of the model to a given situation. In particular, in this paper we have implemented a modular ANN using the following architectures: Hopfield network, Rosenblatt perceptron and neocognitron.

Various other methods of data analysis, such as regression and correlation analysis, interpolation, mathematical (analytical) modeling were tested for comparison and evaluation of the obtained results. Comparative analysis has shown that the classical methods are not effective in this concrete case: first, because of the small amount of time available data set provided a large variety of indicators. This fact makes it impossible to determine dependencies and sort data indicators on the degree of influence. Second, the training set for the proposed analysis is not sufficiently complete and representative.

Given the highly subjective and chaotic nature of the data provided by the environmental experts for the analysis pre-treatment of the data was performed. This allowed a number of ways to divide the source data into the training and the control samples to verify the adequacy of the model. The studies clearly confirmed that the modular ANN-model showed the best results in the analysis and prediction of seasonal changes in the composition and volume of MSW.

**1. Data analysis.** At the first stage of the study regression and correlation analysis of input data was conducted to test the representativeness of the sample and the presence of the interdependence of these data. It was determined that the initial sample does not contain substantially interdependent data and it is well suited for the detection and prediction of seasonal fluctuations.

Training sample consists of data for the following regions (Table 1):

- Georgia, Kutaisi – 05.2010–04.2011;
- Lithuania, Kaunas – 01.2009–12.2010;
- Russia, Saint-Petersburg – 01.2010–12.2010;
- Ukraine, Boryspil – 01.2010–12.2010.

Statistical data used for prediction, were divided into the following categories:

- The composition of solid waste;
- Fullness of garbage bins;
- The total calorie;
- The result of recycling.

In this case, the results of recycling include a substance, energy and heat emitted directly during recycling, and after it.

In turn, MSW has been divided into separate categories in order to increase the efficiency of the system, since the studied parameters for these categories differ considerably among themselves and thus vary independently for each category.

Among these major categories were chosen:

- Paper and paperboard;
- Plastic;
- Black Metal;
- Other metals;
- Glass;
- TetraPack;
- Food waste;
- Garden waste;
- Other organic waste;
- Other inorganic waste.

We should also note the possible better efficiency of analysis by further separating these categories into sub-categories, namely: separate examination of paper and cardboard, the allocation of a separate category of packaging, office paper, etc. However, such an analysis requires a much more complete sample. In our case, such data were not available for most regions. Also, one should take into account that such an expansion will significantly increase the number of projected indicators and may reduce the accuracy of the forecast.

At the preliminary stage the version of analysis and prediction of deviations from the mean values for the year for each indicator has been considered since the sample does not show dynamics for long-term changes (Table 2). However, this approach was recognized as non-optimal, since it increases the amount of work during the preliminary examination of the samples and the obtained results while the data obtained as the result of the analysis do not differ significantly.

When constructing a model we also took into account the results of previous studies, namely – the interdependence of the composition and volume of solid waste and socio-economic indicators [7]. As a result, for each indicator we actually allocated two modules for the prediction: the first – for prediction based on the values for the previous periods, the second – for prediction on the basis of the values of the socio-economic parameters for the current period.

**2. Forecasting model.** On the basis of the above and of the previous studies four main groups of indicators were suggested: 1) Composition of MSW; 2) Fullness of garbage bins; 3) The total calorie content; 4) The result of recycling.

Also we indicated three main areas of prediction:

1. Based on the value of indicators for the current period determining the value of the same indicators for the follow-up period;
2. Based on the socio-economic indicators for the next period of time, to determine the composition of MSW in the next period of time;
3. According to the composition of MSW for the next period of time to determine the results of their recycling in the next period of time.

Thus, for example, we have two values of the results of recycling: firstly, the values resulting from prediction based on the values of the previous periods, and secondly, "associated" values obtained from the composition of MSW for the predicted time taking into account the socio-economic and environmental situation in the region. The resulting value is taken as a kind of average value of these two values, and the corresponding coefficients are selected in the learning process.









Table 2

## Average values of the volume of MSW and garbage bins fullness

	Input			Landfill				
	2009.00	2010.00	2011.00	2009.00	2010.00	2011.00		
Georgia, Kutaisi	Population number, thousands		193.62	194.99	Population number, thousands		193.62	194.99
	GDP, EUR per capita		162.00	199.75	GDP, EUR per capita		162.00	199.75
	GDP, USD per capita		232.75	266.50	GDP, USD per capita		232.75	266.50
	Average air relative humidity		68.63	73.75	Average air relative humidity		68.63	73.75
	Average air temperature, oC		20.88	7.95	Average air temperature, oC		20.88	7.95
	<i>Paper and cardboard</i>		8.97	14.23	<i>Paper and cardboard</i>		1.73	2.41
	<i>Plastics</i>		12.63	15.24	<i>Plastics</i>		2.46	2.58
	<i>Ferrous metals</i>		0.84	2.16	<i>Ferrous metals</i>		0.16	0.36
	<i>Other metals</i>		0.28	1.52	<i>Other metals</i>		0.05	0.26
	<i>Glass</i>		3.16	5.76	<i>Glass</i>		0.61	0.98
	<i>Tetrapaks</i>		0.41	0.31	<i>Tetrapaks</i>		0.08	0.05
	<i>Food waste</i>		50.04	31.02	<i>Food waste</i>		9.79	5.28
	<i>Yard waste</i>		7.55	5.26	<i>Yard waste</i>		1.48	0.88
	<i>Wood</i>		3.08	2.92	<i>Wood</i>		0.62	0.49
	<i>Other organic</i>		5.25	11.58	<i>Other organic</i>		1.01	1.95
	<i>Other inorganic</i>		7.39	9.72	<i>Other inorganic</i>		1.44	1.64
	<i>Hazardous</i>		0.39	0.30	<i>Hazardous</i>		0.08	0.05
<b>Average</b>		100.00	100.00	<b>Total:</b>		19.51	16.93	
Lithuania, Kaunas	Population number, thousands	352285.00	343693.25	Population number, thousands	352285.00	343693.25		
	GDP, EUR per capita	694.94	694.94	GDP, EUR per capita	694.94	694.94		
	GDP, USD per capita	920.46	920.46	GDP, USD per capita	920.46	920.46		
	Average air relative humidity	77.88	77.88	Average air relative humidity	77.88	77.88		
	Average air temperature, oC	4.72	4.72	Average air temperature, oC	4.72	4.72		
	<i>Paper and cardboard</i>	11.16	13.46	<i>Paper and cardboard</i>	4.03	4.25		
	<i>Plastics</i>	12.15	12.89	<i>Plastics</i>	4.41	4.61		
	<i>Ferrous metals</i>	1.49	1.40	<i>Ferrous metals</i>	0.53	0.50		
	<i>Other metals</i>	0.67	1.16	<i>Other metals</i>	0.24	0.41		
	<i>Glass</i>	9.35	9.80	<i>Glass</i>	3.42	3.14		
	<i>Tetrapaks</i>	0.78	0.77	<i>Tetrapaks</i>	0.29	0.29		
	<i>Food waste</i>	36.29	31.75	<i>Food waste</i>	13.23	11.68		
	<i>Yard waste</i>	4.82	6.69	<i>Yard waste</i>	1.83	1.11		
	<i>Wood</i>	1.32	0.57	<i>Wood</i>	0.48	0.21		
	<i>Other organic</i>	14.30	15.62	<i>Other organic</i>	5.27	5.82		
	<i>Other inorganic</i>	7.65	5.47	<i>Other inorganic</i>	2.78	2.09		
	<i>Hazardous</i>	0.02	0.41	<i>Hazardous</i>	0.01	0.15		
<b>Average</b>	100.00	100.00	<b>Total:</b>	36.53	34.24			

Table 2 (end)

Russia, SPB	Population number, thousands	4600.26		Population number, thousands	4600.26	
	GDP, EUR per capita	833.00		GDP, EUR per capita	833.00	
	GDP, USD per capita	1117.00		GDP, USD per capita	1117.00	
	Average air relative humidity	76.75		Average air relative humidity	76.75	
	Average air temperature, oC	5.83		Average air temperature, oC	5.83	
	<i>Paper and cardboard</i>	18.01		<i>Paper and cardboard</i>	4.56	
	<i>Plastics</i>	14.29		<i>Plastics</i>	3.62	
	<i>Ferrous metals</i>	2.17		<i>Ferrous metals</i>	0.54	
	<i>Other metals</i>	0.67		<i>Other metals</i>	0.17	
	<i>Glass</i>	8.74		<i>Glass</i>	2.21	
	<i>Tetrapaks</i>	2.35		<i>Tetrapaks</i>	0.61	
	<i>Food waste</i>	6.12		<i>Food waste</i>	1.53	
	<i>Yard waste</i>	0.62		<i>Yard waste</i>	0.16	
	<i>Wood</i>	0.74		<i>Wood</i>	0.19	
	<i>Other organic</i>	22.61		<i>Other organic</i>	5.75	
	<i>Other inorganic</i>	22.92		<i>Other inorganic</i>	5.83	
<i>Hazardous</i>	0.75		<i>Hazardous</i>	0.19		
<b>Average</b>	100.00		<b>Total:</b>	25.35		
Ukraine, Borispol	Population number, thousands	58.12	58.49	Population number, thousands	58.12	58.49
	GDP, EUR per capita	662.32		GDP, EUR per capita	662.32	
	GDP, USD per capita	910.68		GDP, USD per capita	910.68	
	Average air relative humidity	72.65		Average air relative humidity	72.65	
	Average air temperature, oC	9.28		Average air temperature, oC	9.28	
	<i>Paper and cardboard</i>	8.34	11.31	<i>Paper and cardboard</i>	2.32	
	<i>Plastics</i>	12.49	14.47	<i>Plastics</i>	3.50	
	<i>Ferrous metals</i>	1.45	2.56	<i>Ferrous metals</i>	0.41	
	<i>Other metals</i>	0.25	0.53	<i>Other metals</i>	0.07	
	<i>Glass</i>	20.00	19.12	<i>Glass</i>	5.60	
	<i>Tetrapaks</i>	0.62	0.71	<i>Tetrapaks</i>	0.17	
	<i>Food waste</i>	22.95	20.26	<i>Food waste</i>	6.35	
	<i>Yard waste</i>	12.08	10.09	<i>Yard waste</i>	3.42	
	<i>Wood</i>	1.50	1.30	<i>Wood</i>	0.42	
	<i>Other organic</i>	11.46	10.66	<i>Other organic</i>	3.22	
	<i>Other inorganic</i>	8.79	8.76	<i>Other inorganic</i>	2.48	
<i>Hazardous</i>	0.07	0.24	<i>Hazardous</i>	0.02		
<b>Average</b>	100.00	100.00	<b>Total:</b>	27.99		

Table 3

## Data on the composition of MSW obtained by prediction (2011)

		2010.01	2010.02	2010.03	2010.04	2010.05	2010.06	2010.07	2010.08	2010.09	2010.1	2010.11	2010.12	2011.01	2011.02	2011.03	2011.04	2011.05	2011.06	2011.07	2011.08	2011.09	2011.1	2011.11	2011.12	
Lithuania, Kaunas	Population number, thousands	348624	347728	346831	345935	345038	344142	343245	342349	341452	340556	339659	338763	337866	336970	336073	335177	334280	333384	332487	331591	330694	###	328901	328005	
	GDP, EUR per capita	548	609	645	670	690	706	720	731	742	751	760	767	547.90	609.09	644.89	670.28	689.98	706.08	719.68	731.47	741.87	751.17	759.59	767.27	
	GDP, USD per capita	726	807	854	888	914	935	953	969	983	995	1006	1016	725.73	806.77	854.17	887.81	913.90	935.21	953.24	968.85	982.62	994.94	1006.08	1016.26	
	Average air relative humidity	81.02	84.49	80.74	70.03	73.51	71.47	70.24	74.18	78.49	78.68	85.75	86.00	81.02	84.49	80.74	70.03	73.51	71.47	70.24	74.18	78.49	78.68	85.75	86.00	
	Average air temperature, °C	-11.46	-5.63	-1.65	5.43	11.36	14.02	18.69	17.65	9.68	2.83	1.84	-6.14	-11.46	-5.63	-1.65	5.43	11.36	14.02	18.69	17.65	9.68	2.83	1.84	-6.14	
	Paper and cardboard	13.70	11.86	17.42	8.76	11.79	10.12	11.10	21.84	15.20	15.61	13.48	10.63	17.74	14.87	20.68	10.56	14.10	12.43	16.34	10.91	17.22	18.57	20.01	20.28	
	Plastics	13.84	15.28	12.19	6.42	18.10	9.87	9.83	10.61	14.91	14.78	16.60	12.27	15.59	17.34	13.48	6.97	19.05	10.34	15.71	9.09	9.91	13.95	14.64	14.43	
	Ferrous metals	1.93	1.28	3.32	2.09	1.53	1.05	1.27	0.53	0.73	1.09	0.58	1.44	1.93	1.29	3.33	2.07	1.51	1.02	1.56	0.46	0.71	0.81	0.83	0.86	
	Other metals	0.50	1.85	3.76	0.93	1.28	0.70	1.40	0.92	0.86	0.36	1.11	0.21	0.80	2.96	5.97	1.49	2.04	1.11	3.13	0.79	1.41	0.68	0.71	0.71	
	Glass	10.01	12.55	10.54	7.39	12.58	13.21	11.54	9.21	6.90	5.95	7.40	10.36	9.94	12.26	10.45	7.47	12.89	13.46	7.13	10.00	10.27	8.31	8.26	8.78	
	Tetrapaks	0.66	0.66	1.22	0.72	1.69	0.40	0.95	0.71	0.45	0.54	0.81	0.41	0.65	0.65	1.19	0.70	1.65	0.39	0.92	0.68	0.44	0.52	0.78	0.40	
	Food waste	36.47	37.12	29.77	32.21	24.41	27.71	25.26	26.41	32.77	26.20	35.05	47.59	31.54	33.06	27.20	29.97	22.71	25.38	25.88	38.86	32.19	28.30	28.86	28.92	
	Yard waste	0.90	0.00	0.25	9.68	7.49	5.19	8.58	10.11	13.44	15.31	9.38	0.00	2.32	1.07	1.37	11.31	8.43	6.19	10.03	3.92	10.68	11.65	7.87	6.59	
	Wood	0.83	0.76	1.05	0.73	0.48	2.43	0.00	0.55	0.00	0.00	0.05	0.00	0.05	0.09	0.27	0.28	0.23	2.68	0.13	1.11	1.70	1.32	1.69	1.69	
	Other organic	11.70	14.27	15.44	16.19	17.57	26.29	23.81	17.06	8.22	11.30	12.82	12.81	10.95	13.45	14.98	16.09	17.65	26.86	20.70	21.88	14.33	13.22	13.05	13.39	
	Other inorganic	9.32	4.15	3.16	14.85	3.04	2.40	6.26	1.96	6.11	8.55	2.36	3.50	6.51	3.30	2.57	6.77	2.14	1.81	2.10	1.22	1.46	1.36	0.94	0.50	
	Hazardous	0.15	0.22	1.87	0.04	0.05	0.62	0.01	0.08	0.40	0.31	0.35	0.79	0.11	0.16	1.17	0.04	0.05	0.35	0.02	0.03	0.03	0.06	0.06	0.06	
Total:	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	98.13	100.48	102.67	93.72	102.45	102.04	103.66	98.97	100.36	98.76	97.70	96.61		

Thus, this approach allows to combine the main currently existing approaches for predicting the composition of MSW and the results of their recycling: namely – classical forecasting based on time series [1–3], and an approach that takes into account the interrelation between MSW and the socio-economic and environmental condition of the region. Implementation of the balance between these two approaches is performed automatically in the course of training the neural network. The result is a much more accurate forecast, as both of these approaches are not without drawbacks, and carried out balance allows to compensate for these shortcomings.

**3. The obtained results.** Modular neural network proposed in this paper, showed high efficiency and accuracy in comparison with traditional algorithms and approaches for solving this problem. Mainly as a result of the use of data dependency and the nature of changes in the composition of MSW and the results of their recycling.

Figure 1 shows a diagram of MSW for Lithuania (Kaunas), where the data for the year 2011 (the last third) display the values obtained from the prediction using the proposed modular neural network.

Numerical data for this diagram are shown in Table 3. As can be seen, for 2011, the values of which were predicted, the total amount of MSW obtained as the sum of the values for a particular month, ranges that can be interpreted as fluctuations in the amount MSW. Unfortunately, due to the fact that for the analysis were taken completely different data regions, the consideration of absolute values will strong error in assay results.

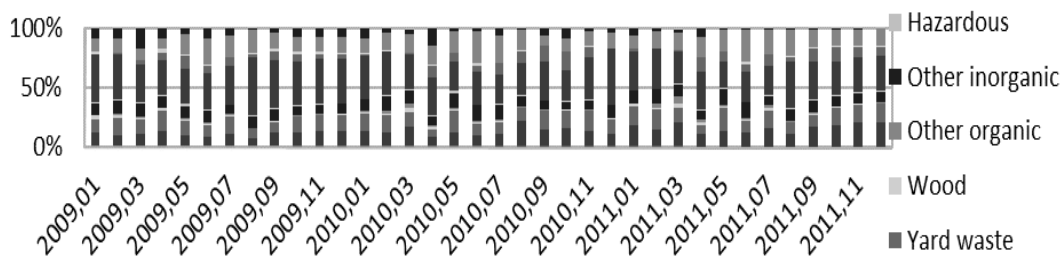


Figure 1: Diagram of oscillations the composition of MSW (Lithuania, Kaunas)

In Figure 2, we can see fluctuations in the area of waste bins for each category of waste.

As can be seen from the results shown in Figure 2 and Table 3, the proposed model takes into account the general downward trend in the amount of MSW and, accordingly, the territory occupied by them, which is observed in the region.

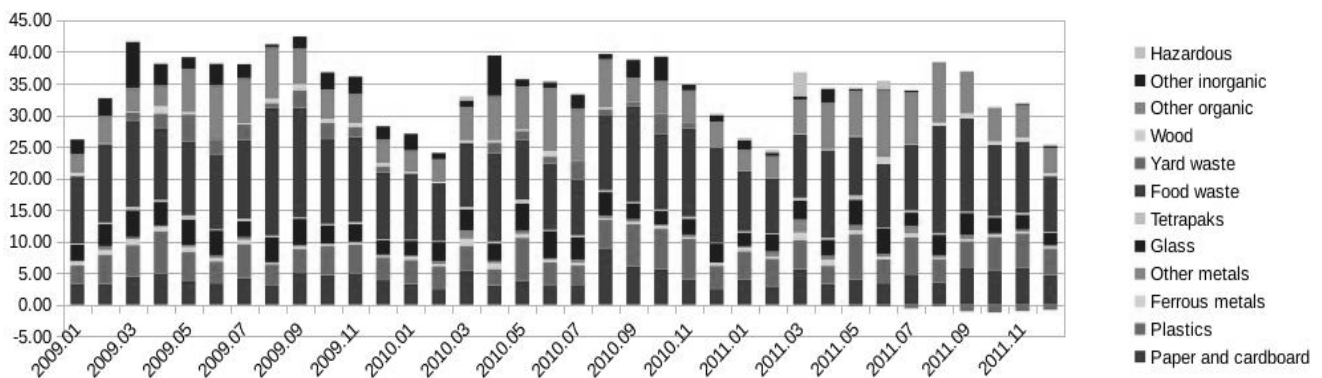


Figure 2: Diagram of fluctuations in the garbage bins fullness (Lithuania, Kaunas)

**4. Conclusions.** From these data it is evident that the modular neural network was able to accurately detect fluctuations and dependencies in the sample, and to reflect these fluctuations in the predicted data.

Analysis of the results also indicates that for long-term forecasting it is necessary to expand the sample so that the presented data displays information about the long-term fluctuations. This extension of the sample will not only get the forecast for the period of time sufficient to use the results in the optimization of storage and recycling MSW, but also will improve the accuracy of prediction of cyclical fluctuations (eg, seasonal).

**REFERENCES**

1. Rimaityte I., Ruzgas T., Denafas G., Racys V., Martuzevičius D. Application and evaluation of forecasting methods for municipal solid waste generation in an eastern-European city // *Waste Management & Research*.– 2012, 30(1).– P. 89–98.
2. Denafas G., Zavauskas K., Martuzevičius D. and others Seasonal aspects of municipal solid waste generation and composition in east-european countries with respect to waste management system development // *Proceedings Venice 2010, Third International Symposium on Energy from Biomass and Waste Venice, Italy*; 8–11 November 2010.
3. Denafas G., Ruzgas T., Martuzevičius D. and others Seasonal variation of municipal solid waste generation and composition in four East European cities // *Resources, Conservation and Recycling*, (89) 2014; P. 22–30.
4. Bojan Batinić, Srđan Vukmirović, Goran Vujčić and others // Using ANN model to determine future waste characteristics in order to achieve specific waste management targets-case study of Serbia // *Journal of Scientific & Industrial Research*, Vol.70, July 2011, pp.513–518.
5. Schmidt A. A modular neural network architecture with additional generalization abilities for high dimensional input vectors // *Manchester Metropolitan University, Department of Computing*, September 1996. – 113 p.

6. Резник А. М., Куссуль М. Э., Сычов А. С., Садовая Е. Г., Калина Е. А. Система автоматизированного проектирования модульных нейронных сетей CAD MNN. // Математичні машини і системи.– 2002, № 3.– С. 28–36.

7. Trokhymchuk R. M., Kozlov K. E. Application of modular neural networks for analysis and prediction of environmental indicators. – // Праці міжнародної наукової конференції "Питання оптимізації обчислень (ПОО-LX)", Київ: Інститут кібернетики НАНУ, 2013.– с. 264.

Стаття надійшла до редколегії 03.03.15

Трохимчук Р. М., канд. фіз.-мат. наук, доц.,  
Козлов К. Є., магістр  
КНУ імені Тараса Шевченка, Київ

### **ВИКОРИСТАННЯ МОДУЛЬНИХ НЕЙРОННИХ МЕРЕЖ ДЛЯ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ СОЦІАЛЬНО-ЕКОНОМІЧНИХ ТА ЕКОЛОГІЧНИХ ПОКАЗНИКІВ РЕГІОНУ**

*Метою даної роботи є використання модульних нейронних мереж для аналізу і прогнозування екологічних показників та їхнього впливу на соціально-економічний стан регіону, а також порівняння результатів використаної моделі з результатами інших методів аналізу даних (методи математичного моделювання та математичної статистики).*

*Ключові слова:* штучні нейронні мережі (ШНМ), модульні нейронні мережі, тверді побутові відходи (ТПВ), модель прогнозування.

Трохимчук Р. Н., канд. фіз.-мат. наук, доц.,  
Козлов К. Е., магістр  
КНУ імени Тараса Шевченка, Киев

### **ИСПОЛЬЗОВАНИЕ МОДУЛЬНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АНАЛИЗА И ПРОГНОЗИРОВАНИЯ СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ И ЭКОЛОГИЧЕСКИХ ПОКАЗАТЕЛЕЙ РЕГИОНА**

*Целью данной работы является использование модульных нейронных сетей для анализа и прогнозирования экологических показателей и их влияния на социально-экономическое положение региона, а также сравнение результатов использованной модели с результатами других методов анализа данных (методы математического моделирования и математической статистики).*

*Ключевые слова:* искусственные нейронные сети (ИНС), модульные нейронные сети, твердые бытовые отходы (ТБО), модель прогнозирования.



Наукове видання



# ВІСНИК

КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ІМЕНІ ТАРАСА ШЕВЧЕНКА

## КІБЕРНЕТИКА

Випуск 1(15)

Друкується за авторською редакцією

Оригінал-макет виготовлено ВПЦ "Київський університет"

Автори опублікованих матеріалів несуть повну відповідальність за підбір, точність наведених фактів, цитат, економіко-статистичних даних, власних імен та інших відомостей. Редколегія залишає за собою право скорочувати та редагувати подані матеріали. Рукописи та дискети не повертаються.



Формат 60x84<sup>1/8</sup>. Ум. друк. арк. 8,4. Наклад 300. Зам. № 215-7437.

Гарнітура Arial. Папір офсетний. Друк офсетний. Вид. № К1.

Підписано до друку 15.02.16

Видавець і виготовлювач

ВПЦ "Київський університет"

01601, Київ, б-р Т. Шевченка, 14, кімн. 43

☎ (38044) 239 3222; (38044) 239 3172; тел./факс (38044) 239 3128

e-mail: vpc@univ.kiev.ua

http: vpc.univ.kiev.ua

Свідоцтво суб'єкта видавничої справи ДК № 1103 від 31.10.02