

Інститут кібернетики імені В. М. Глушкова
Національної академії наук України
Кам'янець-Подільський національний університет
імені Івана Огієнка

МАТЕМАТИЧНЕ ТА КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ

Серія: Технічні науки

Збірник наукових праць

Випуск 21

Кам'янець-Подільський національний університет
імені Івана Огієнка
2020

УДК 004.94:53.072

ББК 30

М34

Свідцтво про державну реєстрацію друкованого засобу масової інформації:
Серія КВ № 14522-3493Р від 25.06.2008 р.

Збірник включено до Переліку наукових фахових видань України категорії Б
галузь «Технічні науки» (спеціальність 012 «Комп'ютерні науки»)
наказ Міністерства освіти і науки України № 409 від 17.03.2020 р.

Друкується згідно з рішенням вченої ради Кам'янець-Подільського
національного університету імені Івана Огієнка,
протокол №10 від 29 жовтня 2020 року.

Редакційна колегія:

- О. М. Хіміч**, член-кореспондент НАНУ,
доктор фізико-математичних наук, професор (*відповідальний редактор*);
А. Ф. Верлянь, член-кореспондент НАПНУ,
доктор технічних наук, професор (*заст. відповідального редактора*);
В. А. Федорчук, доктор технічних наук, професор (*відповідальний секретар*);
Т. Бокалруд, доктор філософії, професор, Норвегія;
В. П. Боюн, член-кореспондент НАНУ, доктор технічних наук, професор;
Д. Г. Вєлев, доктор філософії, професор, Болгарія;
А. А. Верлянь, доктор філософії, професор, Норвегія;
В. К. Задірака, академік НАНУ, доктор фізико-математичних наук, професор;
В. А. Заславський, доктор технічних наук, професор;
О. С. Коваленко, кандидат технічних наук, доцент;
І. М. Конет, доктор фізико-математичних наук, професор;
С. А. Положаєнко, доктор технічних наук, професор;
Б. А. Худаяров, доктор технічних наук, професор, Узбекистан.

Математичне та комп'ютерне моделювання. Серія: Технічні науки : зб.
М34 наук. праць / Інститут кібернетики імені В. М. Глушкова Національної
академії наук України, Кам'янець-Подільський національний університет
імені Івана Огієнка ; [редкол.: О. М. Хіміч (відп. ред.) та ін.]. Кам'янець-
Подільський: Кам'янець-Подільський національний університет імені Івана
Огієнка, 2020. Вип. 21. 144 с.

У збірнику друкуються результати досліджень, що стосуються проблем
застосування математичних моделей у різних галузях людської діяльності.

Збірник включений до бази даних наукових журналів Норвегії.

Для наукових та інженерно-технічних працівників, докторантів, аспірантів,
студентів вищих навчальних закладів.

УДК 004.94:53.072

ББК 30

ISSN 2308-5916

DOI: 10.32626/2308-5916.2020-21

© Інститут кібернетики імені В. М. Глушкова НАН України, 2020

© Кам'янець-Подільський національний
університет імені Івана Огієнка, 2020

V. M. Glushkov Institute of Cybernetics
of National Academy of Sciences of Ukraine
Kamianets-Podilskyi National Ivan Ohienko University

MATHEMATICAL AND COMPUTER MODELLING

Series: Technical sciences

Scientific journal

ISSUE 21

Kamianets-Podilskyi National Ivan Ohienko University
2020

Editorial board:

O. Himich, Corresponding Member of the NAS of Ukraine, Doctor of Physical and Mathematical Sciences, Professor (*Executive Editor*);

A. F. Verlan, Corresponding Member of the NAPS of Ukraine, Doctor of Technical Science, Professor (*Vice Executive Editor*);

V. Fedorchuk, Doctor of Technical Science, Professor (*Responsible Secretary*);

T. Bokalrud, Associate Professor, Norway;

V. Boyun, Corresponding Member of the NAS of Ukraine, Doctor of Technical Science, Professor;

B. Khudayarov, Doctor of Technical Science, Professor, Uzbekistan;

I. Konet, Doctor of Physical and Mathematical Sciences, Professor;

O. Kovalenko, PhD, Associate Professor;

S. Polozhaenko, Doctor of Technical Science, Professor;

D. Velev, PhD, Professor, Bulgaria;

A. A. Verlan, PhD, Professor, Norway;

V. Zadiraka, Academician of the NAS of Ukraine, Doctor of Physical and Mathematical Sciences, Professor;

V. Zaslavskiy, Doctor of Technical Science, Professor.

Mathematical and computer modelling. Series: Technical sciences: scientific journal / V. M. Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine, Kamianets-Podilskiy National Ivan Ohiienko University ; [Editorial Board: O. Himich (*Executive Editor*) and others]. Kamianets-Podilskiy: Kamianets-Podilskiy National Ivan Ohiienko University, 2020. ISSUE 21. 144 p.

The journal publishes results of studies on the mathematical models' application problems in various areas of human activity.

Joint with NTNU the journal has been included to the database of Norwegian Register for Scientific Journals, Series and Publishers.

Intended for scientific and engineering staff, researchers, undergraduate, graduate and PhD students, post-graduates.

© V. M. Glushkov Institute of Cybernetics
of NAS of Ukraine, 2020

© Kamianets-Podilskiy National
Ivan Ohiienko University, 2020

ISSN 2308-5916

DOI: 10.32626/2308-5916.2020-21

УДК 519.6

DOI: 10.32626/2308-5916.2020-21.5-24

С. В. Барановський*, канд. техн. наук,
А. Я. Бомба**, д-р техн. наук, професор

*Рівненський державний гуманітарний університет, м. Рівне,

**Національний університет водного господарства
та природокористування, м. Рівне

УЗАГАЛЬНЕННЯ МАТЕМАТИЧНОЇ МОДЕЛІ ПРОТИВІРУСНОЇ ІМУННОЇ ВІДПОВІДІ МАРЧУКА-ПЕТРОВА З УРАХУВАННЯМ ВПЛИВУ МАЛИХ ПРОСТОРОВО РОЗПОДІЛЕНИХ ДИФУЗІЙНИХ ЗБУРЕНЬ

Для дослідження закономірностей розвитку процесів імунного захисту організму від вірусів та хвороботворних бактерій розроблено досить різноманітний спектр математичних моделей. Відома модель противірусної імунної відповіді Марчука-Петрова, що описує механізми імунного захисту клітинного та гуморального типів, побудована у припущенні, що середовище «організму» є однорідним і усі компоненти процесу в ньому миттєво перемішуються.

У статті узагальнено математичну модель Марчука-Петрова з метою урахування малих просторово розподілених дифузійних впливів на розвиток вірусного захворювання. Відповідну сингулярно збурену модельну задачу із запізненнями зведено до послідовності задач без запізнення, для яких отримані відповідні асимптотичні розвинення розв'язків. Наведено результати числових експериментів, що характеризують вплив просторово розподілених дифузійних факторів вірусного захворювання на розвиток імунної відповіді. Проілюстровано модельне зниження максимального рівня кількості антигенів в епіцентрі зараження унаслідок їх дифузійного «розмивання» в процесі розвитку вірусного захворювання. Підкреслено, що навіть у випадку, коли початкова кількість антигенів на деякій ділянці території зараження організму перевищуватиме певне критичне значення (імунологічний бар'єр), дифузійний «перерозподіл» за певний проміжок часу знизить понад критичні значення концентрацій антигенів до рівня вже нижчого за критичний, і подальше їх знешкодження може бути забезпеченим наявним в організмі до зараження рівнем імунного захисту. Тобто, у рамках даної моделі з деякого моменту часу є впевненість, що розвиток гострої форми протікання вірусного захворювання не лише не відбуватиметься, але й матиме місце прямування до асимптотично стійкого стаціонарного режиму, що характеризує стан здорового організму.

Ключові слова: *модель противірусної імунної відповіді, динамічні системи, асимптотичні методи, сингулярно збурені задачі, задачі із запізненням.*

Вступ. В [1] наведено математичну модель противірусної імунної відповіді Марчука-Петрова, що описує динаміку вірусної інфекції і імунної відповіді системою з десяти диференціальних рівнянь із запізненнями, яку запишемо так:

$$\begin{aligned} \frac{dV_f}{dt} &= h_1 C_V + h_2 C_V E - h_3 F V_f - h_4 V_f - h_5 (C^* - C_V - m) V_f, \\ \frac{dM_V}{dt} &= h_6 V_f - h_7 M_V, \\ \frac{dH_E}{dt} &= h_8 \xi(m) M_V (t - \tau_1) H_E (t - \tau_1) - h_9 M_V H_E - h_{10} M_V H_E E + h_{11} (H_E^* - H_E), \\ \frac{dH_B}{dt} &= h_{12} \xi(m) M_V (t - \tau_2) H_B (t - \tau_2) - h_{13} M_V H_B - h_{14} M_V H_B B + h_{15} (H_B^* - H_B), \\ \frac{dE}{dt} &= h_{16} \xi(m) M_V (t - \tau_3) H_E (t - \tau_3) E (t - \tau_3) - \\ &\quad - h_{17} M_V H_E E - h_{18} C_V E + h_{19} (E^* - E), \\ \frac{dB}{dt} &= h_{20} \xi(m) M_V (t - \tau_4) H_B (t - \tau_4) B (t - \tau_4) - h_{21} M_V H_B B + h_{22} (B^* - B), \\ \frac{dP}{dt} &= h_{23} \xi(m) M_V (t - \tau_5) H_B (t - \tau_5) B (t - \tau_5) + h_{24} (P^* - P), \\ \frac{dF}{dt} &= h_{25} P - h_{26} V_f F - h_{27} F, \\ \frac{dC_V}{dt} &= h_{28} V_f (C^* - C_V - m) - h_{29} C_V E - h_{30} C_V, \\ \frac{dm}{dt} &= h_{31} C_V E + h_{32} C_V - h_{33} m, \end{aligned} \quad (1)$$

з початковими умовами

$$\begin{aligned} V_f(0) &= V_f^0, \quad M_V(0) = 0, \quad H_E(0) = H_E^0, \quad H_B(0) = H_B^0, \quad E(0) = E^0, \\ B(0) &= B^0, \quad P(0) = P^0, \quad F(0) = F^0, \quad C_V(0) = 0, \quad m(0) = 0, \\ M_V(\tilde{t}) H_E(\tilde{t}) &= 0, \quad -\tau_1 \leq \tilde{t} < 0, \quad M_V(\tilde{t}) H_B(\tilde{t}) = 0, \quad -\tau_2 \leq \tilde{t} < 0, \\ M_V(\tilde{t}) H_E(\tilde{t}) E(\tilde{t}) &= 0, \quad -\tau_3 \leq \tilde{t} < 0, \\ M_V(\tilde{t}) H_B(\tilde{t}) B(\tilde{t}) &= 0, \quad -\tilde{\tau} \leq \tilde{t} < 0, \quad \tilde{\tau} = \max\{\tau_4, \tau_5\}, \end{aligned} \quad (2)$$

де V_f — кількість вільно циркулюючих в організмі вірусів, M_V — кількість стимульованих (антигенпрезентуючих) макрофагів, H_E — кількість T -лімфоцитів-помічників клітинного імунітету, H_B — кількість T -лімфоцитів-помічників гуморального імунітету, E — кількість

T -клітин-ефекторів (кілерів), B — кількість B -лімфоцитів, P — кількість плазматичних клітин, F — кількість антитіл, C_V — кількість заражених вірусами клітин органу-мішені, m — нефункціонуюча частина ураженого вірусами органу-мішені, функція $\xi(m)$ пов'язана зі зниженням ефективності імунної системи при ураженні органу вірусами, $0 \leq \xi(m) \leq 1$, h_1, h_2, \dots, h_{33} — параметри моделі.

На відміну від найпростішої моделі інфекційного захворювання [1], в моделі (1)-(2), окрім імунітету гуморального типу з виробництвом антитіл B -системою лімфоцитів, враховано клітинний тип імунітету з накопиченням цитотоксичних T -лімфоцитів-ефекторів. Як зазначено в [1], антитіла нейтралізують віруси, які вільно циркулюють в крові, але не можуть проникати всередину заражених клітин і нейтралізувати віруси, які в них розмножуються. Виявлення і знищення заражених клітин здійснюють накопичені в результаті імунної відповіді цитотоксичні T -лімфоцити-ефектори. Причому, за відсутності в організмі вірусів в [1] наведений єдиний стаціонарний розв'язок, що відповідає стану здорового організму:

$$\begin{aligned} V_f &= 0, \quad M_V = 0, \quad H_E = H_E^*, \quad H_B = H_B^*, \quad E = E^*, \\ B &= B^*, \quad P = P^*, \quad F = \frac{h_{25}F^*}{h_{27}}, \quad C_V = 0, \quad m = 0. \end{aligned} \quad (3)$$

Його асимптотична стійкість гарантується виконанням достатньої умови [1]

$$(h_4 + h_3F^* + h_5C^*)(h_{31}E^* + h_{32}) > h_{28}C^*(h_1 + h_2E^*) \quad (4)$$

і $h_7 > 0$, $h_{11} > 0$, $h_{15} > 0$, $h_{19} > 0$, $h_{22} > 0$, $h_{24} > 0$, $h_{27} > 0$ і $h_{33} > 0$.

Умова (4) забезпечує існування деякого імунологічного бар'єру V^* такого, що при зараженні здорового організму (3) деякою малою дозою вірусів V_f^0 , яка не перевищує V^* ($V_f^0 < V^*$), хвороба в організмі не розвиватиметься. При цьому зазначимо, що в рамках найпростішої математичної моделі інфекційного захворювання Марчука даний параметр знаходиться за явною формулою [1], а саме $V^* = \mu_f (\gamma F^* - \beta) / (\beta \eta \gamma)$, де стала F^* характеризує нормальний рівень концентрації антитіл в здоровому організмі, μ_f — швидкість природного руйнування антитіл, η — витрата антитіл на нейтралізацію одного антигену, β — швидкість розмноження антигенів, γ — коефіцієнт, що враховує ефект взаємодії антигенів з антитілами.

Як і найпростіша модель інфекційного захворювання, модель протівірусної імунної відповіді (1)-(2) побудована у припущенні, що середовище «організму» є однорідним і усі компоненти процесу в ньому

миттєво перемішуються. З іншого боку, при побудові моделі припускалося, що запуск протівірусної імунної відповіді клітинного і гуморального типів здійснюється через так званий механізм «подвійного розпізнавання», який передбачає розпізнавання антигенів T -лімфоцитами та B -лімфоцитами у комплексі з антигенами макрофагів та, природно, не відбувається миттєво після потрапляння антигенів в організм. Отже, частини антигенів, яка не буде відразу знешкоджена імунною системою, проникатиме в клітини, розмножуватимуться в них та поширюватимуться далі в організмі. В результаті навколо уражених клітин утворюватимуться осередки зараження з більш високою концентрацією антигенів. Природно вважати, що з часом породжені в організмі антигени перерозподілятимуться від початкових осередків зараження до навколишніх незаражених ділянок, збільшуючи зону ураження та зменшуючи значення концентрацій антигенів в епіцентрах зараження.

Метою роботи є урахування малих просторово розподілених дифузійних збурень при математичному моделюванні такого роду розвитку протівірусної імунної відповіді.

Постановка задачі. Модифікуємо модель протівірусної імунної відповіді (1)-(2) шляхом введення додаткових членів, які описують малі дифузійні просторово розподілені впливи («перерозподіли»). Відповідну просторово-часову динаміку модельних компонент процесу в області $G_Z = \{(x, t) : -\infty < x < +\infty; 0 < t < +\infty\}$ опишемо такою сингулярно-збуреною системою нелінійних диференціальних рівнянь:

$$\begin{aligned} \frac{dV_f(x, t)}{dt} &= h_1 C_V(x, t) + h_2 C_V(x, t) E(x, t) - h_3 F(x, t) V_f(x, t) - h_4 V_f(x, t) - \\ &\quad - h_5 (C^* - C_V(x, t) - m(x, t)) V_f(x, t) + \varepsilon D_{V_f} \frac{\partial^2 V_f(x, t)}{\partial x^2}, \\ \frac{dM_V(x, t)}{dt} &= h_6 V_f(x, t) - h_7 M_V(x, t) + \varepsilon^2 D_{M_V} \frac{\partial^2 M_V(x, t)}{\partial x^2}, \\ \frac{dH_E(x, t)}{dt} &= h_8 \xi(m) M_V(x, t - \tau_1) H_E(x, t - \tau_1) - h_9 M_V(x, t) H_E(x, t) - \\ &\quad - h_{10} M_V(x, t) H_E(x, t) E(x, t) + h_{11} (H_E^* - H_E(x, t)) + \varepsilon^2 D_{H_E} \frac{\partial^2 H_E(x, t)}{\partial x^2}, \\ \frac{dH_B(x, t)}{dt} &= h_{12} \xi(m) M_V(x, t - \tau_2) H_B(x, t - \tau_2) - h_{13} M_V(x, t) H_B(x, t) - \\ &\quad - h_{14} M_V(x, t) H_B(x, t) B(x, t) + h_{15} (H_B^* - H_B(x, t)) + \varepsilon^2 D_{H_B} \frac{\partial^2 H_B(x, t)}{\partial x^2}, \end{aligned}$$

$$\frac{dE(x,t)}{dt} = h_{16}\xi(m)M_V(x,t-\tau_3)H_E(x,t-\tau_3)E(x,t-\tau_3) - h_{17}M_V(x,t) \times \quad (5)$$

$$\times H_E(x,t)E(x,t) - h_{18}C_V(x,t)E(x,t) + h_{19}(E^* - E(x,t)) + \varepsilon^2 D_E \frac{\partial^2 E(x,t)}{\partial x^2},$$

$$\frac{dB(x,t)}{dt} = h_{20}\xi(m)M_V(x,t-\tau_4)H_B(x,t-\tau_4)B(x,t-\tau_4) - h_{21}M_V(x,t) \times$$

$$\times H_B(x,t)B(x,t) + h_{22}(B^* - B(x,t)) + \varepsilon^2 D_B \frac{\partial^2 B(x,t)}{\partial x^2},$$

$$\frac{dP(x,t)}{dt} = h_{23}\xi(m)M_V(x,t-\tau_5)H_B(x,t-\tau_5)B(x,t-\tau_5) +$$

$$+ h_{24}(P^* - P(x,t)) + \varepsilon^2 D_P \frac{\partial^2 P(x,t)}{\partial x^2},$$

$$\frac{dF(x,t)}{dt} = h_{25}P(x,t) - h_{26}V_f(x,t)F(x,t) - h_{27}F(x,t) + \varepsilon D_F \frac{\partial^2 F(x,t)}{\partial x^2},$$

$$\frac{dC_V(x,t)}{dt} = h_{28}V_f(x,t)(C^* - C_V(x,t) - m(x,t)) -$$

$$- h_{29}C_V(x,t)E(x,t) - h_{30}C_V(x,t) + \varepsilon D_{C_V} \frac{\partial^2 C_V(x,t)}{\partial x^2},$$

$$\frac{dm(x,t)}{dt} = h_{31}C_V(x,t)E(x,t) + h_{32}C_V(x,t) - h_{33}m(x,t) + \varepsilon^2 D_m \frac{\partial^2 m(x,t)}{\partial x^2},$$

з початковими умовами

$$V_f(x,0) = V_f^0(x), \quad M_V(x,0) = 0, \quad H_E(x,0) = H_E^0(x), \quad H_B(x,0) = H_B^0(x),$$

$$E(x,0) = E^0(x), \quad B(x,0) = B^0(x), \quad P(x,0) = P^0(x), \quad F(x,0) = F^0(x),$$

$$C_V(x,0) = 0, \quad m(x,0) = 0, \quad M_V(x,\tilde{t})H_E(x,\tilde{t}) = 0, \quad -\tau_1 \leq \tilde{t} < 0, \quad (6)$$

$$M_V(x,\tilde{t})H_B(x,\tilde{t}) = 0, \quad -\tau_2 \leq \tilde{t} < 0, \quad M_V(x,\tilde{t})H_E(x,\tilde{t})E(x,\tilde{t}) = 0,$$

$$-\tau_3 \leq \tilde{t} < 0, \quad M_V(x,\tilde{t})H_B(x,\tilde{t})B(x,\tilde{t}) = 0, \quad -\tilde{\tau} \leq \tilde{t} < 0, \quad \tilde{\tau} = \max\{\tau_4, \tau_5\},$$

де $V_f = V_f(x,t)$, $M_V = M_V(x,t)$, $H_E = H_E(x,t)$, $H_B = H_B(x,t)$,
 $E = E(x,t)$, $B = B(x,t)$, $P = P(x,t)$, $F = F(x,t)$, $C_V = C_V(x,t)$,
 $m = m(x,t)$ — відповідно кількості вільно циркулюючих вірусів,

стимульованих макрофагів, T -лімфоцитів-помічників клітинного імунітету, T -лімфоцитів-помічників гуморального імунітету, T -клітин-ефекторів, B -лімфоцитів, плазматичних клітин, антитіл, заражених вірусами клітин та нефункціонуюча частина ураженого вірусами ор-

гану-мішені в точці x в момент часу t , $V_f^0(x)$ — функція типу «шапочки», $V_f^0(x)$, $H_E^0(x)$, $H_B^0(x)$, $E^0(x)$, $B^0(x)$, $P^0(x)$, $F^0(x)$ — достатнє число раз неперервно диференційовані обмежені функції, εD_{V_f} , $\varepsilon^2 D_{M_v}$, $\varepsilon^2 D_{H_E}$, $\varepsilon^2 D_{H_B}$, $\varepsilon^2 D_E$, $\varepsilon^2 D_B$, $\varepsilon^2 D_P$, εD_F , εD_{C_v} , $\varepsilon^2 D_m$ — коефіцієнти дифузійного перерозподілу відповідних компонент процесу, ε — малий параметр, що характеризує малий вплив відповідних компонент у порівнянні з іншими (домінуючими) складовими процесу. При цьому відхилення у часі в загальному випадку представимо, як $\tau_i = \tilde{s}_i \cdot \Delta \tau$, де $\tilde{s}_i = s_0 \cdot s_i$, $s_0 = \text{НСД}(\tilde{s}_1, \dots, \tilde{s}_5)$, $s_i \in N$, $i = \overline{1, 5}$, $\Delta \tau > 0$.

Значимо, що в [1, 2] поряд з іншим розглянуті задачі ідентифікації параметрів моделі противірусної імунної відповіді на основі клініко-лабораторних даних конкретних вірусних захворювань. Наведені тут результати свідчать, що процеси ділення H_E і H_B клітин, а також T -ефекторів і B -лімфоцитів є досить близькими. А тому тривалості циклів ділення відповідних клітин можемо далі прийняти однаковими, а саме $\tau_1 = \tau_2$ і $\tau_3 = \tau_4$, при цьому $\tau_1 < \tau_3 < \tau_5$. Отже, відхилення у часі в цьому випадку можемо представити так: $\tau_1 = \tau_2 = \bar{s} \cdot \tau$, $\tau_3 = \tau_4 = \bar{s} \cdot \tau$, $\tau_5 = s \cdot \tau$, де $\tau = s_0 \cdot \Delta \tau$, $\bar{s} = s_1 = s_2$, $\bar{s} = s_3 = s_4$, $s = s_5$, $\bar{s} < \bar{s} < s$.

Обмежимося випадком, коли функціонування імунологічних органів є повноцінним і не залежить від важкості хвороби ($\xi(m) = 1$). Зведемо розв'язок задачі (5)-(6) із запізненнями до послідовності розв'язків задач без запізнень [3]. При цьому, на проміжку $0 \leq t \leq \tau_1$ матимемо:

$$\begin{aligned} \frac{dV_{fk}}{dt} &= h_1 C_{V_k} + h_2 C_{V_k} E_k - h_3 F_k V_{fk} - h_4 V_{fk} - \\ &\quad - h_5 (C^* - C_{V_k} - m_k) V_{fk} + \varepsilon D_{V_f} \frac{\partial^2 V_{fk}}{\partial x^2}, \\ \frac{dM_{V_k}}{dt} &= h_6 V_{fk} - h_7 M_{V_k} + \varepsilon^2 D_{M_v} \frac{\partial^2 M_{V_k}}{\partial x^2}, \\ \frac{dH_{Ek}}{dt} &= -h_9 M_{V_k} H_{Ek} - h_{10} M_{V_k} H_{Ek} E_k + h_{11} (H_E^* - H_{Ek}) + \varepsilon^2 D_{H_E} \frac{\partial^2 H_{Ek}}{\partial x^2}, \\ \frac{dH_{Bk}}{dt} &= -h_{13} M_{V_k} H_{Bk} - h_{14} M_{V_k} H_{Bk} B_k + h_{15} (H_B^* - H_{Bk}) + \varepsilon^2 D_{H_B} \frac{\partial^2 H_{Bk}}{\partial x^2}, \end{aligned}$$

$$\frac{dE_k}{dt} = -h_{17}M_{V_k}H_{E_k}E_k - h_{18}C_{V_k}E_k + h_{19}(E^* - E_k) + \varepsilon^2 D_E \frac{\partial^2 E_k}{\partial x^2}, \quad (7)$$

$$\frac{dB_k}{dt} = -h_{21}M_{V_k}H_{B_k}B_k + h_{22}(B^* - B_k) + \varepsilon^2 D_B \frac{\partial^2 B_k}{\partial x^2},$$

$$\frac{dP_k}{dt} = h_{24}(P^* - P_k) + \varepsilon^2 D_P \frac{\partial^2 P_k}{\partial x^2},$$

$$\frac{dF_k}{dt} = h_{25}P_k - h_{26}V_{f_k}F_k - h_{27}F_k + \varepsilon D_F \frac{\partial^2 F_k}{\partial x^2},$$

$$\frac{dC_{V_k}}{dt} = h_{28}V_{f_k}(C^* - C_{V_k} - m_k) - h_{29}C_{V_k}E_k - h_{30}C_{V_k} + \varepsilon D_{C_v} \frac{\partial^2 C_{V_k}}{\partial x^2},$$

$$\frac{dm_k}{dt} = h_{31}C_{V_k}E_k + h_{32}C_{V_k} - h_{33}m_k + \varepsilon^2 D_m \frac{\partial^2 m_k}{\partial x^2},$$

$$V_{f_k}(x, (k-1)\tau) = V_{f_{k-1}}(x, (k-1)\tau), \quad M_{V_k}(x, (k-1)\tau) = M_{V_{k-1}}(x, (k-1)\tau),$$

$$H_{E_k}(x, (k-1)\tau) = H_{E_{k-1}}(x, (k-1)\tau), \quad H_{B_k}(x, (k-1)\tau) = H_{B_{k-1}}(x, (k-1)\tau),$$

$$E_k(x, (k-1)\tau) = E_{k-1}(x, (k-1)\tau), \quad B_k(x, (k-1)\tau) = B_{k-1}(x, (k-1)\tau),$$

$$P_k(x, (k-1)\tau) = P_{k-1}(x, (k-1)\tau), \quad F_k(x, (k-1)\tau) = F_{k-1}(x, (k-1)\tau),$$

$$C_{V_k}(x, (k-1)\tau) = C_{V_{k-1}}(x, (k-1)\tau), \quad m_k(x, (k-1)\tau) = m_{k-1}(x, (k-1)\tau),$$

$$(k-1)\tau \leq t \leq k\tau, \quad k = 1, 2, \dots, \bar{s},$$

де

$$V_{f_0}(x, 0) = V_f^0(x), \quad M_{V_0}(x, 0) = 0, \quad H_{E_0}(x, 0) = H_E^0(x),$$

$$H_{B_0}(x, 0) = H_B^0(x), \quad E_0(x, 0) = E^0(x), \quad B_0(x, 0) = B^0(x),$$

$$P_0(x, 0) = P^0(x), \quad F_0(x, 0) = F^0(x), \quad C_{V_0}(x, 0) = 0, \quad m_0(x, 0) = 0.$$

На проміжках $\tau_1 = \tau_2 \leq t \leq \tau_3$ та $\tau_3 = \tau_4 \leq t \leq \tau_5$ маємо відповідно такі задачі:

$$\frac{dV_{f_k}}{dt} = h_1 C_{V_k} + h_2 C_{V_k} E_k - h_3 F_k V_{f_k} - h_4 V_{f_k} -$$

$$-h_5 (C^* - C_{V_k} - m_k) V_{f_k} + \varepsilon D_{V_f} \frac{\partial^2 V_{f_k}}{\partial x^2},$$

$$\frac{dM_{V_k}}{dt} = h_6 V_{f_k} - h_7 M_{V_k} + \varepsilon^2 D_{M_v} \frac{\partial^2 M_{V_k}}{\partial x^2},$$

$$\begin{aligned}
 \frac{dH_{Ek}}{dt} &= h_8 M_{V_{k-\bar{s}}} (x, t - \tau_1) H_{Ek-\bar{s}} (x, t - \tau_1) - h_9 M_{V_k} H_{Ek} - \\
 &\quad - h_{10} M_{V_k} H_{Ek} E_k + h_{11} (H_E^* - H_{Ek}) + \varepsilon^2 D_{H_E} \frac{\partial^2 H_{Ek}}{\partial x^2}, \\
 \frac{dH_{Bk}}{dt} &= h_{12} M_{V_{k-\bar{s}}} (x, t - \tau_2) H_{Bk-\bar{s}} (x, t - \tau_2) - h_{13} M_{V_k} H_{Bk} - \\
 &\quad - h_{14} M_{V_k} H_{Bk} B_k + h_{15} (H_B^* - H_{Bk}) + \varepsilon^2 D_{H_B} \frac{\partial^2 H_{Bk}}{\partial x^2}, \\
 \frac{dE_k}{dt} &= -h_{17} M_{V_k} H_{Ek} E_k - h_{18} C_{V_k} E_k + h_{19} (E^* - E_k) + \varepsilon^2 D_E \frac{\partial^2 E_k}{\partial x^2}, \quad (8) \\
 \frac{dB_k}{dt} &= -h_{21} M_{V_k} H_{Bk} B_k + h_{22} (B^* - B_k) + \varepsilon^2 D_B \frac{\partial^2 B_k}{\partial x^2}, \\
 \frac{dP_k}{dt} &= h_{24} (P^* - P_k) + \varepsilon^2 D_P \frac{\partial^2 P_k}{\partial x^2}, \\
 \frac{dF_k}{dt} &= h_{25} P_k - h_{26} V_{fk} F_k - h_{27} F_k + \varepsilon D_F \frac{\partial^2 F_k}{\partial x^2}, \\
 \frac{dC_{V_k}}{dt} &= h_{28} V_{fk} (C^* - C_{V_k} - m_k) - h_{29} C_{V_k} E_k - h_{30} C_{V_k} + \varepsilon D_{C_V} \frac{\partial^2 C_{V_k}}{\partial x^2}, \\
 \frac{dm_k}{dt} &= h_{31} C_{V_k} E_k + h_{32} C_{V_k} - h_{33} m_k + \varepsilon^2 D_m \frac{\partial^2 m_k}{\partial x^2}, \\
 V_{fk} (x, (k-1)\tau) &= V_{fk-1} (x, (k-1)\tau), \quad M_{V_k} (x, (k-1)\tau) = M_{V_{k-1}} (x, (k-1)\tau), \\
 H_{Ek} (x, (k-1)\tau) &= H_{Ek-1} (x, (k-1)\tau), \quad H_{Bk} (x, (k-1)\tau) = H_{B_{k-1}} (x, (k-1)\tau), \\
 E_k (x, (k-1)\tau) &= E_{k-1} (x, (k-1)\tau), \quad B_k (x, (k-1)\tau) = B_{k-1} (x, (k-1)\tau), \\
 P_k (x, (k-1)\tau) &= P_{k-1} (x, (k-1)\tau), \quad F_k (x, (k-1)\tau) = F_{k-1} (x, (k-1)\tau), \\
 C_{V_k} (x, (k-1)\tau) &= C_{V_{k-1}} (x, (k-1)\tau), \quad m_k (x, (k-1)\tau) = m_{k-1} (x, (k-1)\tau), \\
 &\quad (k-1)\tau \leq t \leq k\tau, \quad k = \bar{s} + 1, \bar{s} + 2, \dots, \bar{\bar{s}}; \\
 \frac{dV_{fk}}{dt} &= h_1 C_{V_k} + h_2 C_{V_k} E_k - h_3 F_k V_{fk} - h_4 V_{fk} - \\
 &\quad - h_5 (C^* - C_{V_k} - m_k) V_{fk} + \varepsilon D_{V_f} \frac{\partial^2 V_{fk}}{\partial x^2}, \\
 \frac{dM_{V_k}}{dt} &= h_6 V_{fk} - h_7 M_{V_k} + \varepsilon^2 D_{M_V} \frac{\partial^2 M_{V_k}}{\partial x^2},
 \end{aligned}$$

$$\begin{aligned}
 \frac{dH_{Ek}}{dt} &= h_8 M_{V_{k-\bar{s}}} (x, t - \tau_1) H_{Ek-\bar{s}} (x, t - \tau_1) - h_9 M_{V_k} H_{Ek} - \\
 &\quad - h_{10} M_{V_k} H_{Ek} E_k + h_{11} (H_E^* - H_{Ek}) + \varepsilon^2 D_{H_E} \frac{\partial^2 H_{Ek}}{\partial x^2}, \\
 \frac{dH_{Bk}}{dt} &= h_{12} M_{V_{k-\bar{s}}} (x, t - \tau_2) H_{Bk-\bar{s}} (x, t - \tau_2) - h_{13} M_{V_k} H_{Bk} - \\
 &\quad - h_{14} M_{V_k} H_{Bk} B_k + h_{15} (H_B^* - H_{Bk}) + \varepsilon^2 D_{H_B} \frac{\partial^2 H_{Bk}}{\partial x^2}, \\
 \frac{dE_k}{dt} &= h_{16} M_{V_{k-\bar{s}}} (x, t - \tau_3) H_{Ek-\bar{s}} (x, t - \tau_3) E_{k-\bar{s}} (x, t - \tau_3) - \\
 &\quad - h_{17} M_{V_k} H_{Ek} E_k - h_{18} C_{V_k} E_k + h_{19} (E^* - E_k) + \varepsilon^2 D_E \frac{\partial^2 E_k}{\partial x^2}, \\
 \frac{dB_k}{dt} &= h_{20} M_{V_{k-\bar{s}}} (x, t - \tau_4) H_{Bk-\bar{s}} (x, t - \tau_4) B_{k-\bar{s}} (x, t - \tau_4) - \\
 &\quad - h_{21} M_{V_k} H_{Bk} B_k + h_{22} (B^* - B_k) + \varepsilon^2 D_B \frac{\partial^2 B_k}{\partial x^2}, \\
 \frac{dP_k}{dt} &= h_{24} (P^* - P_k) + \varepsilon^2 D_P \frac{\partial^2 P_k}{\partial x^2}, \\
 \frac{dF_k}{dt} &= h_{25} P_k - h_{26} V_{fk} F_k - h_{27} F_k + \varepsilon D_F \frac{\partial^2 F_k}{\partial x^2}, \\
 \frac{dC_{V_k}}{dt} &= h_{28} V_{fk} (C^* - C_{V_k} - m_k) - \\
 &\quad - h_{29} C_{V_k} E_k - h_{30} C_{V_k} + \varepsilon D_{C_v} \frac{\partial^2 C_{V_k}}{\partial x^2}, \\
 \frac{dm_k}{dt} &= h_{31} C_{V_k} E_k + h_{32} C_{V_k} - h_{33} m_k + \varepsilon^2 D_m \frac{\partial^2 m_k}{\partial x^2}, \\
 V_{fk} (x, (k-1)\tau) &= V_{fk-1} (x, (k-1)\tau), \quad M_{V_k} (x, (k-1)\tau) = M_{V_{k-1}} (x, (k-1)\tau), \\
 H_{Ek} (x, (k-1)\tau) &= H_{Ek-1} (x, (k-1)\tau), \quad H_{Bk} (x, (k-1)\tau) = H_{B_{k-1}} (x, (k-1)\tau), \\
 E_k (x, (k-1)\tau) &= E_{k-1} (x, (k-1)\tau), \quad B_k (x, (k-1)\tau) = B_{k-1} (x, (k-1)\tau), \\
 P_k (x, (k-1)\tau) &= P_{k-1} (x, (k-1)\tau), \quad F_k (x, (k-1)\tau) = F_{k-1} (x, (k-1)\tau), \\
 C_{V_k} (x, (k-1)\tau) &= C_{V_{k-1}} (x, (k-1)\tau), \quad m_k (x, (k-1)\tau) = m_{k-1} (x, (k-1)\tau), \\
 &\quad (k-1)\tau \leq t \leq k\tau, \quad k = \bar{s} + 1, \bar{s} + 2, \dots, s.
 \end{aligned} \tag{9}$$

А при $t \geq \tau_5$ маємо:

$$\begin{aligned}
 \frac{dV_{fk}}{dt} &= h_1 C_{V_k} + h_2 C_{V_k} E_k - h_3 F_k V_{fk} - h_4 V_{fk} - \\
 &\quad - h_5 (C^* - C_{V_k} - m_k) V_{fk} + \varepsilon D_{V_f} \frac{\partial^2 V_{fk}}{\partial x^2}, \\
 \frac{dM_{V_k}}{dt} &= h_6 V_{fk} - h_7 M_{V_k} + \varepsilon^2 D_{M_v} \frac{\partial^2 M_{V_k}}{\partial x^2}, \\
 \frac{dH_{E_k}}{dt} &= h_8 M_{V_{k-\bar{s}}} (x, t - \tau_1) H_{E_{k-\bar{s}}} (x, t - \tau_1) - h_9 M_{V_k} H_{E_k} - \\
 &\quad - h_{10} M_{V_k} H_{E_k} E_k + h_{11} (H_E^* - H_{E_k}) + \varepsilon^2 D_{H_e} \frac{\partial^2 H_{E_k}}{\partial x^2}, \\
 \frac{dH_{B_k}}{dt} &= h_{12} M_{V_{k-\bar{s}}} (x, t - \tau_2) H_{B_{k-\bar{s}}} (x, t - \tau_2) - h_{13} M_{V_k} H_{B_k} - \\
 &\quad - h_{14} M_{V_k} H_{B_k} B_k + h_{15} (H_B^* - H_{B_k}) + \varepsilon^2 D_{H_b} \frac{\partial^2 H_{B_k}}{\partial x^2}, \\
 \frac{dE_k}{dt} &= h_{16} M_{V_{k-\bar{s}}} (x, t - \tau_3) H_{E_{k-\bar{s}}} (x, t - \tau_3) E_{k-\bar{s}} (x, t - \tau_3) - \\
 &\quad - h_{17} M_{V_k} H_{E_k} E_k - h_{18} C_{V_k} E_k + h_{19} (E^* - E_k) + \varepsilon^2 D_E \frac{\partial^2 E_k}{\partial x^2}, \\
 \frac{dB_k}{dt} &= h_{20} M_{V_{k-\bar{s}}} (x, t - \tau_4) H_{B_{k-\bar{s}}} (x, t - \tau_4) B_{k-\bar{s}} (x, t - \tau_4) - \\
 &\quad - h_{21} M_{V_k} H_{B_k} B_k + h_{22} (B^* - B_k) + \varepsilon^2 D_B \frac{\partial^2 B_k}{\partial x^2}, \\
 \frac{dP_k}{dt} &= h_{23} M_{V_{k-s}} (x, t - \tau_5) H_{B_{k-s}} (x, t - \tau_5) B_{k-s} (x, t - \tau_5) + \\
 &\quad + h_{24} (P^* - P_k) + \varepsilon^2 D_P \frac{\partial^2 P_k}{\partial x^2}, \\
 \frac{dF_k}{dt} &= h_{25} P_k - h_{26} V_{fk} F_k - h_{27} F_k + \varepsilon D_F \frac{\partial^2 F_k}{\partial x^2}, \\
 \frac{dC_{V_k}}{dt} &= h_{28} V_{fk} (C^* - C_{V_k} - m_k) - h_{29} C_{V_k} E_k - h_{30} C_{V_k} + \varepsilon D_{C_v} \frac{\partial^2 C_{V_k}}{\partial x^2}, \\
 \frac{dm_k}{dt} &= h_{31} C_{V_k} E_k + h_{32} C_{V_k} - h_{33} m_k + \varepsilon^2 D_m \frac{\partial^2 m_k}{\partial x^2},
 \end{aligned} \tag{10}$$

$$V_{fk}(x, (k-1)\tau) = V_{f_{k-1}}(x, (k-1)\tau), \quad M_{V_k}(x, (k-1)\tau) = M_{V_{k-1}}(x, (k-1)\tau),$$

$$\begin{aligned}
 H_{Ek}(x, (k-1)\tau) &= H_{Ek-1}(x, (k-1)\tau), \quad H_{Bk}(x, (k-1)\tau) = H_{Bk-1}(x, (k-1)\tau), \\
 E_k(x, (k-1)\tau) &= E_{k-1}(x, (k-1)\tau), \quad B_k(x, (k-1)\tau) = B_{k-1}(x, (k-1)\tau), \\
 P_k(x, (k-1)\tau) &= P_{k-1}(x, (k-1)\tau), \quad F_k(x, (k-1)\tau) = F_{k-1}(x, (k-1)\tau), \\
 C_{V_k}(x, (k-1)\tau) &= C_{V_{k-1}}(x, (k-1)\tau), \quad m_k(x, (k-1)\tau) = m_{k-1}(x, (k-1)\tau), \\
 &(k-1)\tau \leq t \leq k\tau, \quad k = s+1, s+2, \dots
 \end{aligned}$$

Для забезпечення достатньої гладкості відповідних розв'язків при $t = \tau_1 = \tau_2, \dots, t = \tau_3 = \tau_4, \dots, t = \tau_5, \dots$ крім традиційних умов гладкості щодо функцій початкових умов в моделі імунної відповіді потрібно накласти ще й умови їх узгодженості при $t = -\tau_5, \dots, t = -\tau_3 = -\tau_4, \dots$ [4]. Зокрема, має виконуватись умова

$$\begin{aligned}
 \frac{dP_1(x, 0)}{dt} &= h_{23}M_{V_0}(x, -\tau_5)H_{B_0}(x, -\tau_5)B_0(x, -\tau_5) + \\
 &+ h_{24}(P^* - P_1(x, 0)) + \varepsilon^2 D_P \frac{\partial^2 P_1(x, 0)}{\partial x^2}.
 \end{aligned}$$

Враховуючи те, що просторово розподілений дифузійний перерозподіл компонент процесу імунної відповіді є малими у порівнянні з іншими складовими процесу, застосуємо асимптотичний метод для розв'язання відповідних сингулярно збурених модельних задач (7)-(10) [4-6]. Зокрема, розв'язки задач (7)-(10) формально представимо у вигляді асимптотичних рядів

$$\begin{aligned}
 V_{fk}(x, t) &= \sum_{i=0}^n \varepsilon^i V_{fik}(x, t) + R_{nk}^{V_f}(x, t, \varepsilon), \\
 M_{V_k}(x, t) &= \sum_{i=0}^n \varepsilon^i M_{V_{ik}}(x, t) + R_{nk}^{M_V}(x, t, \varepsilon), \\
 H_{Ek}(x, t) &= \sum_{i=0}^n (\varepsilon^i \times H_{E_{ik}}(x, t)) + R_{nk}^{H_E}(x, t, \varepsilon), \\
 H_{Bk}(x, t) &= \sum_{i=0}^n \varepsilon^i H_{B_{ik}}(x, t) + R_{nk}^{H_B}(x, t, \varepsilon), \\
 E_k(x, t) &= \sum_{i=0}^n \varepsilon^i E_{ik}(x, t) + R_{nk}^E(x, t, \varepsilon), \\
 B_k(x, t) &= \sum_{i=0}^n \varepsilon^i B_{ik}(x, t) + R_{nk}^B(x, t, \varepsilon), \\
 P_k(x, t) &= \sum_{i=0}^n \varepsilon^i P_{ik}(x, t) + R_{nk}^P(x, t, \varepsilon),
 \end{aligned}$$

$$\begin{aligned}
 F_k(x, t) &= \sum_{i=0}^n \varepsilon^i F_{ik}(x, t) + R_{nk}^F(x, t, \varepsilon), \\
 C_{V_k}(x, t) &= \sum_{i=0}^n \varepsilon^i C_{Vik}(x, t) + R_{nk}^{C_V}(x, t, \varepsilon), \\
 m_k(x, t) &= \sum_{i=0}^n \varepsilon^i m_{ik}(x, t) + R_{nk}^m(x, t, \varepsilon)
 \end{aligned}$$

як збурення розв'язків відповідних вироджених задач [6, 7], де $k = 1, 2, \dots$, $V_{fk}(x, t)$, $M_{V_k}(x, t)$, $H_{E_k}(x, t)$, $H_{B_k}(x, t)$, $E_k(x, t)$, $B_k(x, t)$, $P_k(x, t)$, $F_k(x, t)$, $C_{V_k}(x, t)$, $m_k(x, t)$ — шукані функції (члени асимптотики), $R_{nk}^{V_f}(x, t, \varepsilon)$, $R_{nk}^{M_V}(x, t, \varepsilon)$, $R_{nk}^{H_E}(x, t, \varepsilon)$, $R_{nk}^{H_B}(x, t, \varepsilon)$, $R_{nk}^E(x, t, \varepsilon)$, $R_{nk}^B(x, t, \varepsilon)$, $R_{nk}^P(x, t, \varepsilon)$, $R_{nk}^F(x, t, \varepsilon)$, $R_{nk}^{C_V}(x, t, \varepsilon)$, $R_{nk}^m(x, t, \varepsilon)$ — відповідні залишкові члени. Після підстановки асимптотичних рядів та виконання стандартної процедури прирівнювання коефіцієнтів при однакових степенях ε отримаємо, наприклад, такі задачі для визначення функцій $V_{fi,k}(x, t)$, $M_{Vi,k}(x, t)$, $H_{Ei,k}(x, t)$, $H_{Bi,k}(x, t)$, $E_{i,k}(x, t)$, $B_{i,k}(x, t)$, $P_{i,k}(x, t)$, $F_{i,k}(x, t)$, $C_{Vi,k}(x, t)$, $m_{i,k}(x, t)$ ($i = 0, 1, \dots, n$) на проміжку $0 \leq t \leq \tau_1$:

$$\begin{aligned}
 \frac{dV_{f0,k}}{dt} &= h_1 C_{V0,k} + h_2 C_{V0,k} E_{0,k} - h_3 F_{0,k} V_{f0,k} - \\
 &\quad - h_4 V_{f0,k} - h_5 (C^* - C_{V0,k} - m_{0,k}) V_{f0,k}, \\
 \frac{dM_{V0,k}}{dt} &= h_6 V_{f0,k} - h_7 M_{V0,k}, \\
 \frac{dH_{E0,k}}{dt} &= -h_9 M_{V0,k} H_{E0,k} - h_{10} M_{V0,k} H_{E0,k} E_{0,k} + h_{11} (H_E^* - H_{E0,k}), \\
 \frac{dH_{B0,k}}{dt} &= -h_{13} M_{V0,k} H_{B0,k} - h_{14} M_{V0,k} H_{B0,k} B_{0,k} + h_{15} (H_B^* - H_{B0,k}), \\
 \frac{dE_{0,k}}{dt} &= -h_{17} M_{V0,k} H_{E0,k} E_{0,k} - h_{18} C_{V0,k} E_{0,k} + h_{19} (E^* - E_{0,k}), \\
 \frac{dB_{0,k}}{dt} &= -h_{21} M_{V0,k} H_{B0,k} B_{0,k} + h_{22} (B^* - B_{0,k}), \\
 \frac{dP_{0,k}}{dt} &= h_{24} (P^* - P_{0,k}),
 \end{aligned} \tag{11}$$

$$\begin{aligned} \frac{dF_{0,k}}{dt} &= h_{25}P_{0,k} - h_{26}V_{f0,k}F_{0,k} - h_{27}F_{0,k}, \\ \frac{dC_{V0,k}}{dt} &= h_{28}V_{f0,k}(C^* - C_{V0,k} - m_{0,k}) - h_{29}C_{V0,k}E_{0,k} - h_{30}C_{V0,k}, \\ \frac{dm_{0,k}}{dt} &= h_{31}C_{V0,k}E_{0,k} + h_{32}C_{V0,k} - h_{33}m_{0,k}, \\ V_{f0,k}(x, (k-1)\tau) &= V_{f,k-1}(x, (k-1)\tau), \quad M_{V0,k}(x, (k-1)\tau) = M_{V,k-1}(x, (k-1)\tau), \\ H_{E0,k}(x, (k-1)\tau) &= H_{E,k-1}(x, (k-1)\tau), \quad H_{B0,k}(x, (k-1)\tau) = H_{B,k-1}(x, (k-1)\tau), \\ E_{0,k}(x, (k-1)\tau) &= E_{k-1}(x, (k-1)\tau), \quad B_{0,k}(x, (k-1)\tau) = B_{k-1}(x, (k-1)\tau), \\ P_{0,k}(x, (k-1)\tau) &= P_{k-1}(x, (k-1)\tau), \quad F_{0,k}(x, (k-1)\tau) = F_{k-1}(x, (k-1)\tau), \\ C_{V0,k}(x, (k-1)\tau) &= C_{V,k-1}(x, (k-1)\tau), \quad m_{0,k}(x, (k-1)\tau) = m_{k-1}(x, (k-1)\tau), \\ &(k-1)\tau \leq t \leq k\tau, \quad k = 1, 2, \dots, \bar{s}; \\ \frac{dV_{f1,k}}{dt} &= h_1C_{V1,k} + h_2(a_k^{C_v}E_{1,k} + a_k^E C_{V1,k}) - h_3(a_k^F V_{f1,k} + a_k^{V_f} F_{1,k}) - \\ &- (h_4 + h_5 C^*)V_{f1,k} + h_5((a_k^{C_v} + a_k^m)V_{f1,k} + a_k^{V_f}(C_{V1,k} + m_{1,k})) + \Phi_{1,k}^{V_f}, \\ \frac{dM_{V1,k}}{dt} &= h_6V_{f1,k} - h_7M_{V1,k}, \\ \frac{dH_{E1,k}}{dt} &= -h_9(a_k^{M_v}H_{E1,k} + a_k^{H_\varepsilon}M_{V1,k}) - \\ &- h_{10}(a_k^{H_\varepsilon}a_k^E M_{V1,k} + a_k^{M_v}a_k^E H_{E1,k} + a_k^{M_v}a_k^{H_\varepsilon}E_{1,k}) - h_{11}H_{E1,k}, \\ \frac{dH_{B1,k}}{dt} &= -h_{13}(a_k^{M_v}H_{B1,k} + a_k^{H_B}M_{V1,k}) - \\ &- h_{14}(a_k^{H_B}a_k^B M_{V1,k} + a_k^{M_v}a_k^B H_{B1,k} + a_k^{M_v}a_k^{H_B}B_{1,k}) - h_{15}H_{B1,k}, \\ \frac{dE_{1,k}}{dt} &= -h_{17}(a_k^{H_\varepsilon}a_k^E M_{V1,k} + a_k^{M_v}a_k^E H_{E1,k} + a_k^{M_v}a_k^{H_\varepsilon}E_{1,k}) - \\ &- h_{18}(a_k^E C_{V1,k} + a_k^{C_v}E_{1,k}) - h_{19}E_{1,k}, \\ \frac{dB_{1,k}}{dt} &= -h_{21}(a_k^{H_B}a_k^B M_{V1,k} + a_k^{M_v}a_k^B H_{B1,k} + a_k^{M_v}a_k^{H_B}B_{1,k}) - h_{22}B_{1,k}, \quad (12) \\ \frac{dP_{1,k}}{dt} &= -h_{24}P_{1,k}, \end{aligned}$$

$$\begin{aligned}
 \frac{dF_{1,k}}{dt} &= h_{25}P_{1,k} - h_{26} \left(a_k^F V_{f1,k} + a_k^{V_f} F_{1,k} \right) - h_{27}F_{1,k} + \Phi_{1,k}^F, \\
 \frac{dC_{V1,k}}{dt} &= h_{28}C^*V_{f1,k} - h_{28} \left(\left(a_k^{C_v} + a_k^m \right) V_{f1,k} + a_k^{V_f} \left(C_{V1,k} + m_{1,k} \right) \right) - \\
 &\quad - h_{29} \left(a_k^E C_{V1,k} + a_k^{C_v} E_{1,k} \right) - h_{30}C_{V1} + \Phi_{1,k}^{C_v}, \\
 \frac{dm_{1,k}}{dt} &= h_{31} \left(a_k^E C_{V1,k} + a_k^{C_v} E_{1,k} \right) + h_{32}C_{V1,k} - h_{33}m_{1,k}, \\
 V_{f1,k} \left(x, (k-1)\tau \right) &= 0, \quad M_{V1,k} \left(x, (k-1)\tau \right) = 0, \quad H_{E1,k} \left(x, (k-1)\tau \right) = 0, \\
 H_{B1,k} \left(x, (k-1)\tau \right) &= 0, \quad E_{1,k} \left(x, (k-1)\tau \right) = 0, \quad B_{1,k} \left(x, (k-1)\tau \right) = 0, \\
 P_{1,k} \left(x, (k-1)\tau \right) &= 0, \quad F_{1,k} \left(x, (k-1)\tau \right) = 0, \quad C_{V1,k} \left(x, (k-1)\tau \right) = 0, \\
 m_{1,k} \left(x, (k-1)\tau \right) &= 0, \quad (k-1)\tau \leq t \leq k\tau, \quad k = 1, 2, \dots, \bar{s}; \\
 \frac{dV_{fi,k}}{dt} &= h_1 C_{Vi,k} + h_2 \left(a_k^{C_v} E_{i,k} + a_k^E C_{Vi,k} \right) - h_3 \left(a_k^F V_{fi,k} + a_k^{V_f} F_{2,k} \right) - \\
 &\quad - \left(h_4 + h_5 C^* \right) V_{fi,k} + h_5 \left(\left(a_k^{C_v} + a_k^m \right) V_{fi,k} + a_k^{V_f} \left(C_{Vi,k} + m_{i,k} \right) \right) + \Phi_{i,k}^{V_f}, \\
 \frac{dM_{Vi,k}}{dt} &= h_6 V_{fi,k} - h_7 M_{Vi,k} + \Phi_{i,k}^{M_v}, \\
 \frac{dH_{Ei,k}}{dt} &= -h_9 \left(a_k^{H_e} M_{Vi,k} + a_k^{M_v} H_{Ei,k} \right) - \\
 &\quad - h_{10} \left(a_k^{H_e} a_k^E M_{Vi,k} + a_k^{M_v} a_k^E H_{Ei,k} + a_k^{M_v} a_k^{H_e} E_{i,k} \right) - h_{11} H_{Ei,k} + \Phi_{i,k}^{H_e}, \\
 \frac{dH_{Bi,k}}{dt} &= -h_{13} \left(a_k^{H_b} M_{Vi,k} + a_k^{M_v} H_{Bi,k} \right) - \\
 &\quad - h_{14} \left(a_k^{H_b} a_k^B M_{Vi,k} + a_k^{M_v} a_k^B H_{Bi,k} + a_k^{M_v} a_k^{H_b} B_{i,k} \right) - h_{15} H_{Bi,k} + \Phi_{i,k}^{H_b}, \\
 \frac{dE_{i,k}}{dt} &= -h_{17} \left(a_k^{H_e} a_k^E M_{Vi,k} + a_k^{M_v} a_k^E H_{Ei,k} + a_k^{M_v} a_k^{H_e} E_{i,k} \right) - \\
 &\quad - h_{18} \left(a_k^E C_{Vi,k} + a_k^{C_v} E_{i,k} \right) - h_{19} E_{i,k} + \Phi_{i,k}^E, \\
 \frac{dB_{i,k}}{dt} &= -h_{21} \left(a_k^{H_b} a_k^B M_{Vi,k} + a_k^{M_v} a_k^B H_{Bi,k} + a_k^{M_v} a_k^{H_b} B_{i,k} \right) - h_{22} B_{i,k} + \Phi_{i,k}^B, \quad (13) \\
 \frac{dP_{i,k}}{dt} &= -h_{24} P_{i,k} + \Phi_{i,k}^P, \\
 \frac{dF_{i,k}}{dt} &= h_{25} P_{i,k} - h_{26} \left(a_k^F V_{fi,k} + a_k^{V_f} F_{i,k} \right) - h_{27} F_{i,k} + \Phi_{i,k}^F,
 \end{aligned}$$

$$\frac{dC_{Vi,k}}{dt} = h_{28} C_{Vi,k}^* - h_{28} \left((a_k^{C_v} + a_k^m) V_{fi,k} + a_k^{V_f} (C_{Vi,k} + m_{i,k}) \right) - h_{29} \left(a_k^E C_{Vi,k} + a_k^{C_v} E_{i,k} \right) - h_{30} C_{Vi,k} + \Phi_{i,k}^{C_v},$$

$$\frac{dm_{i,k}}{dt} = h_{31} \left(a_k^E C_{Vi,k} + a_k^{C_v} E_{i,k} \right) + h_{32} C_{Vi,k} - h_{33} m_{i,k} + \Phi_{i,k}^m,$$

$$V_{fi,k}(x, (k-1)\tau) = 0, \quad M_{Vi,k}(x, (k-1)\tau) = 0, \quad H_{Ei,k}(x, (k-1)\tau) = 0,$$

$$H_{Bi,k}(x, (k-1)\tau) = 0, \quad E_{i,k}(x, (k-1)\tau) = 0, \quad B_{i,k}(x, (k-1)\tau) = 0,$$

$$P_{i,k}(x, (k-1)\tau) = 0, \quad F_{i,k}(x, (k-1)\tau) = 0, \quad C_{Vi,k}(x, (k-1)\tau) = 0,$$

$$m_{i,k}(x, (k-1)\tau) = 0, \quad (k-1)\tau \leq t \leq k\tau, \quad k = 1, 2, \dots, \bar{s};$$

де

$$a_k^{C_v} = C_{V0,k}, \quad a_k^E = E_{0,k}, \quad a_k^F = F_{0,k}, \quad a_k^{V_f} = V_{f0,k}, \quad a_k^m = m_{0,k},$$

$$a_k^{M_v} = M_{V0,k}, \quad a_k^{H_E} = H_{E0,k}, \quad a_k^{H_B} = H_{B0,k}, \quad a_k^B = B_{0,k};$$

$$\Phi_{1,k}^{V_f} = D_{V_f} \frac{\partial^2 V_{f0,k}}{\partial x^2}, \quad \Phi_{1,k}^F = D_F \frac{\partial^2 F_{0,k}}{\partial x^2}, \quad \Phi_{1,k}^{C_v} = D_{C_v} \frac{\partial^2 C_{V0,k}}{\partial x^2};$$

$$\Phi_{i,k}^{V_f} = h_2 \sum_{j=1}^{i-1} C_{Vj,k} E_{i-j,k} - h_3 \sum_{j=1}^{i-1} F_{j,k} V_{fi-j,k} +$$

$$+ h_5 \sum_{j=1}^{i-1} (C_{Vj,k} + m_{j,k}) V_{fi-j,k} + D_{V_f} \frac{\partial^2 V_{fi-1,k}}{\partial x^2},$$

$$\Phi_{i,k}^{M_v} = D_{M_v} \frac{\partial^2 M_{Vi-2,k}}{\partial x^2}, \quad \Phi_{i,k}^F = -h_{26} \sum_{j=1}^{i-1} V_{fj,k} F_{i-j,k} + D_F \frac{\partial^2 F_{i-1,k}}{\partial x^2},$$

$$\Phi_{i,k}^{H_E} = -h_9 \sum_{j=1}^{i-1} M_{Vj,k} H_{Ei-j,k} - h_{10} \left(E_{0,k} \sum_{j=1}^{i-1} M_{Vj,k} H_{Ei-j,k} + \sum_{l=1}^{i-1} E_{l,k} \sum_{j=0}^{i-l} (M_{Vj,k} H_{Ei-l-j,k}) \right) + D_{H_E} \frac{\partial^2 H_{Ei-2,k}}{\partial x^2},$$

$$\Phi_{i,k}^{H_B} = -h_{13} \sum_{j=1}^{i-1} M_{Vj,k} H_{Bi-j,k} - h_{14} \left(B_{0,k} \sum_{j=1}^{i-1} M_{Vj,k} H_{Bi-j,k} + \sum_{l=1}^{i-1} B_{l,k} \sum_{j=0}^{i-l} (M_{Vj,k} H_{Bi-l-j,k}) \right) + D_{H_B} \frac{\partial^2 H_{Bi-2,k}}{\partial x^2},$$

$$\begin{aligned} \Phi_{i,k}^E &= -h_{17} \left(E_{0,k} \sum_{j=1}^{i-1} M_{V j,k} H_{Ei-j,k} + \sum_{l=1}^{i-1} E_{l,k} \sum_{j=0}^{i-1} M_{V j,k} H_{Ei-l-j,k} \right) - \\ &\quad - h_{18} \sum_{j=1}^{i-1} M_{V j,k} E_{i-j,k} + D_E \frac{\partial^2 E_{i-2,k}}{\partial x^2}, \\ \Phi_{i,k}^B &= -h_{21} \left(B_{0,k} \sum_{j=1}^{i-1} M_{V j,k} H_{Bi-j,k} + \sum_{l=1}^{i-1} B_{l,k} \sum_{j=0}^{i-1} M_{V j,k} H_{Bi-l-j,k} \right) + D_B \frac{\partial^2 B_{i-2,k}}{\partial x^2}, \\ \Phi_{i,k}^{C_v} &= -h_{28} \sum_{j=1}^{i-1} (C_{V j,k} + m_{j,k}) V_{fi-j,k} - h_{29} \sum_{j=1}^{i-1} C_{V j,k} E_{i-j,k} + D_{C_v} \frac{\partial^2 C_{Vi-1,k}}{\partial x^2}, \\ \Phi_{i,k}^P &= D_P \frac{\partial^2 P_{i-2,k}}{\partial x^2}, \quad \Phi_{i,k}^m = h_{31} \sum_{j=1}^{i-1} C_{V j,k} E_{i-j,k} + D_m \frac{\partial^2 m_{i-2,k}}{\partial x^2}. \end{aligned}$$

Зауважимо, що запропонований підхід нескладно переноситься і на інші, зокрема, скінченні області G_Z . При цьому, звичайно, замість описаних вище мають бути застосовані більш складні розклади (див., наприклад, [4, 5, 7]), що, зокрема, враховують ще й граничні умови.

Оцінки залишкових членів $R_{nk}^{V_f}(x, t, \varepsilon)$, $R_{nk}^{M_v}(x, t, \varepsilon)$, $R_{nk}^{H_E}(x, t, \varepsilon)$, $R_{nk}^{H_B}(x, t, \varepsilon)$, $R_{nk}^E(x, t, \varepsilon)$, $R_{nk}^B(x, t, \varepsilon)$, $R_{nk}^P(x, t, \varepsilon)$, $R_{nk}^F(x, t, \varepsilon)$, $R_{nk}^{C_v}(x, t, \varepsilon)$, $R_{nk}^m(x, t, \varepsilon)$ встановлення просторово-часових проміжків збіжності для прогнозування конкретних процесів здійснюються на основі принципу типу максимуму аналогічно до [4, 5, 7].

Результати числових експериментів. На рис. 1 схематично продемонстровано процес «розмивання» з часом початкового зараження

$$V_f^0(x) = \begin{cases} -\gamma(x-x_1)(x-x_2), & x_1 \leq x \leq x_2, \\ 0, & x \leq x_1 \vee x_2 \leq x \end{cases}$$

«території» $[x_1, x_2]$, що містить «критичну» ділянку $[a, b]$ — перевищення імунологічного бар'єру, коли за рахунок обміну з сусідніми ділянками в деякий момент часу $t = t_*$ на усій території маємо зараження, що не перевищує імунологічний бар'єр $V_f \leq V_f^*$. Тобто з даного моменту часу є впевненість, що в рамках даної моделі організм самостійно не те, що знешкодить це зараження, але й матиме місце прямування до асимптотично стійкого стаціонарного режиму, що характеризує стан здорового організму. А на рис. 2 зображено випадок початкового зараження $V_f^0(x) = -\gamma / \left[1 + (x - (x_1 + x_2)/2)^2 \right]$ усього

організму $(-\infty, +\infty)$, що містить ділянку $[a, b]$, де значення функції $V_f^0(x)$ перевищують імунологічний бар'єр. За відносно невеликий проміжок часу t_* внаслідок дифузійного перерозподілу (обміну з сусідніми ділянками) матиме місце встановлення докритичного стану, при якому імунна система може самостійно знешкودити зараження та вивести організм на асимптотично стійкий стаціонарний режим. В обидвох наведених випадках не має необхідності приймати додаткові «лікувальні заходи» «території» (організму).

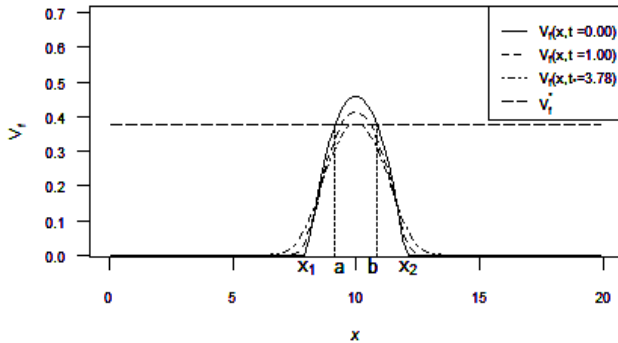


Рис. 1. Схематичне зображення переходу початкового зараження «території» $[x_1, x_2]$ організму $(-\infty, +\infty)$ до некритичного стану

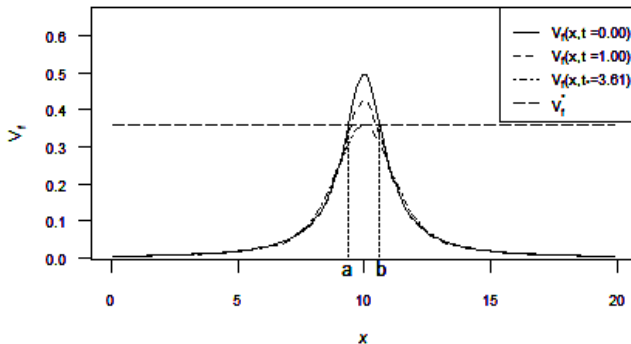


Рис. 2. Схематичне зображення переходу початкового зараження усієї «території» організму $(-\infty, +\infty)$ до некритичного стану

На рис. 3 проілюстровано модельну динаміку кількості вільно циркулюючих антигенів V_f при вірусному захворюванні в епіцентрі зараження за умов нерівномірного у просторі їх розподілу у початковий момент часу та різній модельній інтенсивності дифузійного перерозподілу (значення параметра ϵ). Наведені результати демонструють зменшення швидкості зростання кількості вірусних антигенів в епі-

центрі зараження зі зростанням інтенсивності дифузійного перерозподілу. Зазначимо, що відповідним чином змінюється і динаміка інших діючих факторів захворювання. Отже, модельне урахування дифузійного перерозподілу діючих факторів спричиняє менш «гостре» прогнозне протікання вірусного захворювання.

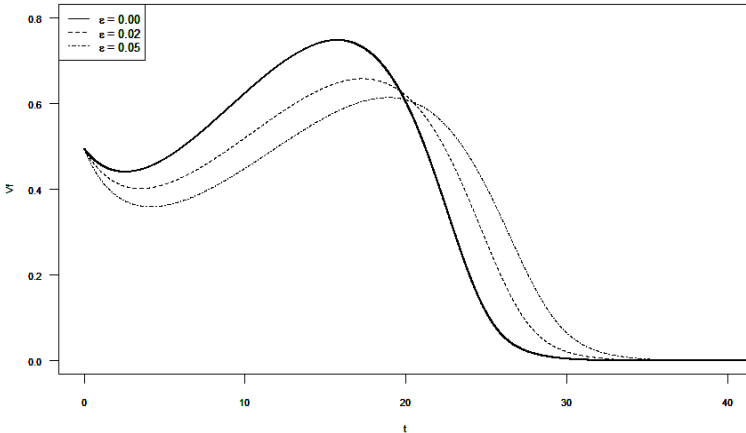


Рис. 3. Динаміка концентрації вірусних антигенів при різних рівнях інтенсивності дифузійного впливу

При проведенні числових експериментів, згідно [1, 2], змінні моделі були відповідним чином масштабовані, а для параметрів моделі приймалися такі значення: $h_1 = 0.1$, $h_2 = 10^{-4}$, $h_3 = 0.1$, $h_4 = 10^{-4}$, $h_5 = 10^{-4}$, $h_6 = 0.05$, $h_7 = 0.02$, $h_8 = 10^{-2}$, $h_9 = 10^{-3}$, $h_{10} = 10^{-4}$, $h_{11} = 0.05$, $h_{12} = 10^{-2}$, $h_{13} = 10^{-3}$, $h_{14} = 10^{-4}$, $h_{15} = 0.05$, $h_{16} = 0.8$, $h_{17} = 0.08$, $h_{18} = 1.5 \cdot 10^{-4}$, $h_{19} = 0.1$, $h_{20} = 0.8$, $h_{21} = 0.08$, $h_{22} = 0.1$, $h_{23} = 0.5$, $h_{24} = 0.16$, $h_{25} = 0.17$, $h_{26} = 0.2$, $h_{27} = 0.17$, $h_{28} = 0.4$, $h_{29} = 0.002$, $h_{30} = 0.005$, $h_{31} = 0.002$, $h_{32} = 0.005$, $h_{33} = 0.12$, $\tau_1 = \tau_2 = 0.5$, $\tau_3 = \tau_4 = 2$, $\tau_5 = 3$.

Висновки. На основі модифікації математичної моделі імунної відповіді Марчука-Петрова представлено підхід для урахування просторово розподілених дифузійних збурень на розвиток вірусного захворювання. Відповідна модельна задача із запізненнями зведена до послідовності задач без запізнень, для яких побудовані представлення шуканих функцій у вигляді асимптотичних рядів як збурення розв'язків відповідних вироджених задач.

За результатами числових експериментів проілюстровано модельне зниження максимального значення кількості антигенів в осередку зараження унаслідок їх дифузійного «перерозподілу» в процесі розвитку вірусного захворювання. Продемонстровано, що навіть у випадку, коли початкова доза антигенів V_f^0 в деякій області зони зараження перевищу-

ватиме певне критичне значення V^* дифузійний «перерозподіл» за певний проміжок часу понизить понад критичні значення кількості антигенів до рівня вже нижчого за критичний, і подальше їх знешкодження може бути забезпеченим наявним рівнем антитіл. Тобто, у рамках даної моделі гострота протікання вірусного захворювання зменшиться. При цьому послідовність розв'язків відповідних сингулярно збурених задач (які визначають покроковий (з урахуванням запізнень $\tau_1, \tau_2, \tau_3, \tau_4, \tau_5$) прогноз розподілу в просторі і часі концентрацій антигенів, антитіл, плазматичних клітин і міри зараженості) прямуватиме до деякого стійкого, зокрема, асимптотично стійкого стаціонарного значення.

Розроблена обчислювальна процедура може бути в основі більш широкого комплексу прийняття рішень типу: чи можемо в даному випадку згідно значень відповідних вхідних даних, зокрема, щодо інтенсивності дифузійного «перерозподілу», величини території зараження тощо повністю покластися на наявний в організмі імунний захист, або, в іншому випадку, здійснювати зовнішній вплив (лікування).

Список використаних джерел:

1. Марчук Г. И. Математические модели в иммунологии. Вычислительные методы и эксперименты. М.: Наука, 1991. 304 с.
2. Марчук Г. И., Романюха А. А., Бочаров Г. А. Математическое моделирование противовирусного иммунного ответа при вирусном гепатите В. *Математические вопросы кибернетики*. М.: Наука, 1989. Вып. 2. С. 5-70.
3. Эльсгольц Л. Э., Норкин С. Б. Введение в теорию дифференциальных уравнений с отклоняющимся аргументом. М.: Наука, 1971. 296 с.
4. Бомба А. Я., Барановський С. В., Присяжнюк І. М. Нелінійні сингулярно збурені задачі типу «конвекція-дифузія». Рівне: НУВГП, 2008. 254 с.
5. Бомба А. Я. Про асимптотичний метод розв'язання однієї задачі масопереносу при фільтрації в пористому середовищі. *Укр. мат. журн.* 1982. Т. 4. №4. С. 493-496.
6. Бомба А.Я., Барановський С. В. Моделювання малих просторово розподілених впливів на динаміку інфекційного захворювання в умовах типу фармакотерапії. *Журнал обчислювальної та прикладної математики*. 2020. № 1 (133). С. 5-17.
7. Бомба А. Я., Барановський С. В. Сингулярні просторово розподілені дифузійні збурення одного класу динамічних процесів. *Вісник Національного університету водного господарства і природокористування* : зб. наук. пр. Рівне: НУВГП, 2019. Вип. 3 (87). С. 54-65.

GENERALIZATION OF ANTI-VIRUS IMMUNE RESPONSE MATHEMATICAL MODEL BY MARCHUK AND PETROV TAKING INTO ACCOUNT THE INFLUENCE OF SMALL SPATIALLY DISTRIBUTED PARTICLES

To study the patterns of development of the body's immune defenses against viruses and pathogenic bacteria, a very diverse range of mathemat-

ical models has been developed. The well-known antiviral immune response model by Marchuk and Petrov, which describes the mechanisms of immune protection of cellular and humoral types, is based on the assumption that the environment of the «organism» is homogeneous and all process components are instantly mixed.

The article summarizes the mathematical model by Marchuk and Petrov in order to take into account small spatially distributed diffusion effects on the development of viral disease. The corresponding singularly perturbed model problem with delays is reduced to a sequence of problems without delay, for which the corresponding asymptotic developments of solutions are obtained. The numerical experiments results characterizing the influence of spatially distributed diffusion factors of viral disease on the development of the immune response are presented. The model decrease of the antigens number maximum level in the infection epicenter due to their diffusion «erosion» in the viral disease process is illustrated. It is emphasized that even if the initial antigens amount in some part of the infected area exceeds a certain critical value (immunological barrier), diffusion «redistribution» for a certain period of time reduces above critical values of antigen concentrations to a level below critical, and their further neutralization can be provided by the immune protection level available in an organism before infection. That is, within this model for some time there is confidence that the development of an viral disease acute form not only not occurs, but takes place to an asymptotically stable steady state, which characterizes the state of a healthy organism.

Key words: *antiviral immune response model, dynamic systems, asymptotic methods, singularly perturbed problems, delayed problems.*

Отримано: 21.05.2020

УДК 004 81

DOI: 10.32626/2308-5916.2020-21.25-42

А. Ф. Верлань*, д-р техн. наук, професор,

И. А. Чмырь**, д-р техн. наук, професор

*Институт проблем моделирования в энергетике

имени Г. Е. Пухова НАН Украины, г. Киев,

**Одесский государственный экологический университет, г. Одесса

ДВЕ МОДЕЛИ ВОПРОСНО-ОТВЕТНОЙ «МАШИНЫ ДИАЛОГА»

В статье описаны две модели вопросно-ответного диалога. Обе модели описывают поведение активного агента (агента, задающего вопросы) вопросно-ответного диалога в процессе решения задачи при помощи диалогового процесса. В этом смысле модели предназначены для описания сценария диалогового метода решения задачи. Предполагается, что при помощи диалоговых методов решаются задачи, относящиеся к классу плохо-формализуемых. Первая модель, базируется на идее конечного автомата Мили. Она имеет ограниченную сферу применимости, поскольку, при генерации очередного вопроса не учитывается история предыдущих ответов реактивного агента диалога. Вторая модель базируется на идее сети Петри. Модель учитывает историю предыдущих ответов реактивного агента, а также возможность нестандартной обработки вопросно-ответной транзакции при помощи процессов-демонов. Модели являются проблемно-независимыми и моделируют вопросно-ответный диалог, который не зависит от предметной области диалога и характера задач, решаемых при помощи диалогового метода. Проблемная независимость моделей обусловлена тем, что стандартные, циклически повторяющиеся действия по обработке вопросно-ответной транзакции отражают унифицированный когнитивный диалоговый цикл. Унифицированный когнитивный цикл диалога построен на основе циклической модели восприятия Найссера. В когнитивном цикле диалога, в отличие от цикла Найссера, источником потока сенсорных событий является не внешняя среда, а реактивный диалоговый агент. Модели предназначены для специфицирования программной «машины диалога», имитирующей целенаправленное поведение активного агента диалога в процессе решения диалоговой задачи. Обе модели предполагают, что основные компоненты «машины диалога» отображаются не в программный код, а представляются данными, хранящимися в базе данных. Машина диалога, таким образом, является программной системой, которая реализует сценарий

диалогового метода решения задачи путем интерпретации данных, хранящихся в базе данных.

Ключевые слова: *вопросно-ответный диалог, диалоговый агент, сценарий диалога, диалоговый метод решения задачи, когнитивный цикл диалога.*

Введение. Диалог является одним из важнейших феноменов ментальной и интеллектуальной деятельности человека. Анализ практически любой когнитивной активности человека обнаруживает, что в её основе лежит диалоговый процесс. Вопросно-ответный (эротетический) диалоговый процесс является важным средством получения знаний [1], а также эффективным средством решения некоторых плохо-формализуемых задач [2, 3]. Исследования, имеющие отношения к диалоговому взаимодействию, можно обнаружить в различных, и не очень близких научных областях: искусственный интеллект, когнитивная психология, эротетическая логика, эпистемология, и др. В то же время отсутствует признанное научное направление, фокусирующее внимание на теории и практике диалогового взаимодействия. Например, в современных учебниках по Когнитивным наукам отсутствует раздел, посвященный онтологии и формальным моделям диалогового взаимодействия [4]. Отмеченное научное направление необходимо не только для лучшего понимания сущности диалога, но и как источник прагматически ценных моделей искусственных диалоговых агентов, предназначенных для встраивания в системы искусственного интеллекта.

В ряде публикаций предлагаются онтологические и формализованные модели естественного и вопросно-ответного диалога, которые могли бы служить «исходным материалом» для теории диалога [5-7]. Однако, предлагаемые модели, как правило, содержат значительное количество неопределённых описаний и их трудно отобразить в компьютерные программы искусственных диалоговых агентов. Настоящая статья посвящена развитию вопросно-ответного моделирования диалогового взаимодействия и описанию нескольких формальных проблемно-независимых моделей. Модели инвариантны к проблемной области диалога и могут генерировать протоколы вопросно-ответных диалогов при реализации диалогового метода решения плохо-формализуемой задачи.

Целью настоящей статьи является синтез двух формальных моделей вопросно-ответного диалога, которые могут быть полезны при проектировании «машины диалога», имитирующей целенаправленное поведение искусственного диалогового агента, инвариантное к предметной области диалога.

1. Когнитивный цикл вопросно-ответного диалога. При разработке формальной модели диалога важным является её адекватность

психологии диалога, или адекватность модели процессам восприятия и переработки информации человеком в процессе диалогового взаимодействия. В том случае, когда в основу формальной модели диалога положена адекватная психологическая модель, то можно ожидать, что искусственный диалоговый агент наследует антропоморфность у системы восприятия и переработки информации человеком.

Среди множества теорий и моделей, предлагаемых когнитивной психологией и имеющих отношение к диалоговому процессу, остановимся на моделях, описывающих диалог на уровне восприятия, который не зависит от предметной области диалога и от характера задачи, решаемой в процессе диалогового процесса. В этом смысле, психологическая модель диалога, полезная в прикладном аспекте, должна описывать некоторую *унифицированную когнитивную «машину диалога»*.

Диалоговый процесс аналогичен процессу перцептивного взаимодействия человека с окружающей средой, моделируемому циклом Найсера [8, 9]. Отличие заключается в том, что в процессе диалога главные компоненты сенсорной системы человека — зрение и слух — подключены не к «естественной» окружающей среде, а к «искусственной» среде, формируемой потоками зрительных и звуковых сенсорных событий, генерируемых противоположным агентом диалога. Таким образом, в диалоговом процессе реальная окружающая среда подменяется искусственной. Однако очевидно, что перцепция и последующая переработка как сенсорных событий, порождаемых искусственной средой (диалоговым агентом), так и сенсорных событий, генерируемых естественной окружающей средой, осуществляется по одним и тем же «правилам и законам» и одной и той же цепью подсистем переработки информации человеком. Диаграмма, иллюстрирующая когнитивный цикл вопросно-ответного диалогового процесса, приведена на рис. 1.

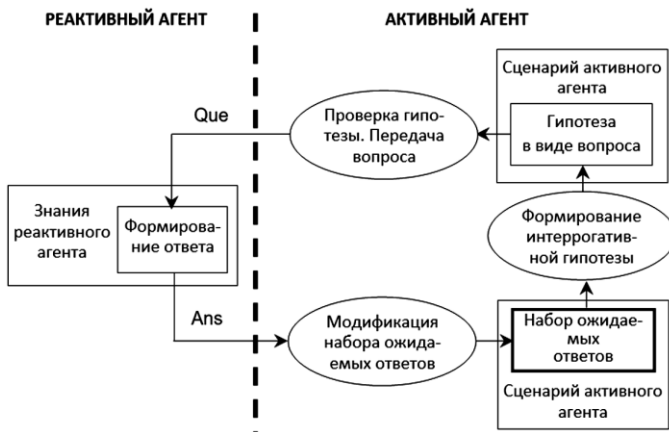


Рис. 1. Когнитивный цикл вопросно-ответного диалогового процесса

Когнитивний цикл діалогового процесу, приведений на рис. 1, моделює поведінку активного агента в діалозі з фіксованими ролями. Фіксування ролей агентів означає заборону на зміну ролей агентів в ході діалогу. Один з агентів завжди задає питання, а інший — завжди відповідає на них. Будемо вважати, для зручності, що діалоговий цикл починається з моменту, коли активний агент оперує відносно невеликим набором відповідей, названим *набором очікуваних відповідей*, а реальна відповідь реактивного агента збігається з одним з відповідей з набору очікуваних відповідей активного агента. Таким чином, однією з основних когнитивних структур, якою оперує активний агент, є схема-відповідь, або ментальне представлення відповіді. Набор очікуваних відповідей вбудований в більш обширну когнитивну структуру, названу *сценарієм активного агента*. Припускається, що сценарій активного агента включає всі необхідні для даного діалогу схеми-відповіді і що набор очікуваних відповідей — це деяке підмножество відповідей з сценарію, релевантне поточному циклу діалогового процесу. Після сприйняття і розпізнавання реальної відповіді, отриманої від реактивного агента, здійснюється модифікація поточного набору очікуваних відповідей, метою якою є підготовка набору очікуваних відповідей для наступного циклу. В результаті модифікації формується новий набор очікуваних відповідей, включаючий відповідь, очікуваний на наступному циклі. В процесі модифікації поточний набор очікуваних відповідей може бути замінений повністю або частково.

Концепція сценарію діалогу означає, що активний агент не «придумує» наступне питання з допомогою деякого «супералгоритму», а шукає його в пам'яті питань, використовуючи як *метод доступу до пам'яті питань* метод досягнення мети діалогу. Таким чином, *метод доступу до пам'яті питань* є *зберігачем методу досягнення мети діалогу*, або *діалоговим методом рішення задачі*. Концепти «пам'ять питань» і «діалоговий метод рішення задачі» входять в обсяг поняття «сценарій» активного агента і деталізують його структуру.

Як інтроспекція, так і аналіз реальних протоколів запитно-відповідних діалогових процесів, виявляють, що існує неоднозначна зв'язь між поточним відповіду реактивного агента і наступним запитом, генерованим активним агентом. Іншими словами, в різних діалогових транзакціях на один і той же відповідь, отриманий від реактивного агента, активний агент може генерувати різні запити. Тому, при розробці діалогових методів рішення задач, для моделювання відміченої неоднозначності необхідно враховувати наступні три принципи.

Принцип «глубины диалога». Принцип «глубины диалога» означает, что активный агент при формировании очередного вопроса должен учитывать как воспринятый ответ, так и индекс диалогового цикла. Здесь термин «индекс» используется как синоним термина имя, или идентификатор. В различных диалоговых циклах на один и тот же ответ, полученный от реактивного агента, могут быть сформированы разные вопросы.

Принцип «истории ответов». Принцип «истории ответов» означает, что активный агент, при формировании очередного вопроса, должен учитывать как воспринятый ответ, так и историю воспринятых ответов. В одном и том же диалоговом цикле на один и тот же ответ, полученный от реактивного агента, могут быть сформированы разные вопросы в зависимости от того, какие ответы были получены в предыдущих циклах.

Принцип «истории вопросов». Принцип «истории вопросов» означает, что активный агент, при формировании очередного вопроса, должен учитывать как воспринятый ответ, так и историю ранее сформированных вопросов. В одном и том же диалоговом цикле на один и тот же ответ могут быть сформированы разные вопросы в зависимости от того, какие вопросы были сформированы в предыдущих циклах.

Диалоговый цикл, приведенный на рис. 1, хорошо описывает «гармоничный диалог», соответствующий циклу Найссера для случая рутинной перцепции. Под гармоничным диалогом будем понимать такой диалог, когда оба агента удовлетворены своими ролями и не хотят их изменять. Однако гармоничный диалог не всегда возможен. Каким образом учесть возможность нарушения гармоничного диалога и изменения текущих ролей диалоговых агентов на противоположные? Инициатором смены ролей обычно выступает реактивный агент, а признаком момента смены ролей является генерация и передача активному агенту информационного сообщения, имеющего статус вопроса. Поэтому одним из способов учёта возможности смены ролей может быть включение в список ожидаемых ответов *детектора вопроса*. Детектор вопроса может быть включён в каждый набор ожидаемых ответов (тогда агенты будут иметь возможность сменить роли в любом диалоговом цикле) или только в некоторые наборы ожидаемых ответов (тогда роли агентов могут быть изменены только в некоторых, заранее определенных, диалоговых циклах).

Рассмотренные, ниже, формальные проблемно-независимые модели диалога являются моделями диалогового метода доступа к памяти вопросов, рассматриваемого также как диалоговый метод решения задачи. Эти модели моделируют поведение активного агента вопросно-ответного диалога, как функционирование некоторой «машины диалога». Целью моделирования является получение структуры «машины диалога», инвариантной к цели и предметной области диалогового процесса.

2. Конечно-автоматная модель. Конечно-автоматная модель вопросно-ответного диалога моделирует гармоничный диалог и базируется на трёх предположениях: (1) задача, решаемая в ходе вопросно-ответного диалога, может быть решена за конечное количество диалоговых циклов (шагов диалога); (2) к началу решения задачи детерминированы все вопросы и ответы, необходимые для формирования диалоговых шагов; (3) модификация текущего набора ожидаемых ответов всегда порождает новый набор, отличающийся от текущего, и имеющий уникальный индекс.

Конечно-автоматная модель вопросно-ответного диалога, в виде диалогового метода доступа к памяти вопросов, базируется на идее автомата Мили [10] и описывается следующей четвёркой:

$$DiAM = (QUE, ANS, S, \varphi), \quad (1)$$

где $DiAM$ обозначает диалоговый метод доступа к памяти вопросов.

$$QUE = \{Que_i; i = 1, \dots, k. \quad (2)$$

QUE — множество индексов вопросов, используемых при решении задачи, (спецификации вопросов, находятся в памяти вопросов).

$$ANS = \{Ans_i; i = 1, \dots, l. \quad (3)$$

ANS — множество ответов, используемых при решении задачи.

$$S = \{S_i; i = 1, \dots, m. \quad (4)$$

S — множество состояний ожидания ответа. Количество элементов множества S равно общему количеству шагов.

$$\varphi: S \times ANS \rightarrow S \times QUE. \quad (5)$$

φ — функция шага, детерминирующая индекс нового вопроса и индекс следующего шага в зависимости от текущего ответа и индекса текущего шага.

При выполнении шага автомат, имитирующий вопросно-ответный диалоговый процесс, выполняет итерации, состоящие из следующей последовательности действий:

- воспринимается текущий ответ реактивного агента;
- распознаётся текущий ответ реактивного агента;
- детерминируется индекс следующего вопроса активного агента.

Если никак не структурировать множество ответов ANS , то это означает, что для распознавания текущего ответа реактивного агента, диалоговый метод доступа $DiAM$ должен, на каждом шаге, оперировать всем множеством ответов ANS . Для обширных сценариев мощность этого множества может быть значительной. Для того, чтобы уменьшить мощность множества ответов, с которым на каждом шаге оперирует $DiAM$, и сделать модель практически реализуемой, введём в модель *множество ответов, распознаваемых на i -м шаге*. Множество распознаваемых ответов содержит только те ответы, которые

ожидаются на i -м шаге, необходимы для реализации диалогового метода и, следовательно, должны быть распознаны. Все остальные ответы отнесём к классу ответов, не распознаваемых на i -м шаге. В дальнейшем будем использовать следующие обозначения: R^i — множество ответов, распознаваемых на i -м шаге; NR^i — множество ответов, не распознаваемых на i -м шаге.

К множеству ответов, не распознаваемых на i -м шаге относятся все ответы, принадлежащие множеству ANS и не принадлежащие множеству R^i , т.е.

$$NR^i = ANS - R^i. \quad (6)$$

Однако, множество ответов, не распознаваемых на i -м шаге, необходимо рассматривать шире и понимать как множество *любых мыслимых ответов* реактивного агента, не принадлежащие множеству R^i , т.е.

$$NR^i = U - R^i, \quad (7)$$

где U — универсум ответов, т.е. множество любых ответов, которые, когда-либо, может породить реактивный агент диалога.

Естественно предположить, что появление на i -м шаге любого ответа, принадлежащего NR^i , приводит к тому, что диалоговый метод доступа $DiAM$ генерирует один и тот же индекс последующего вопроса. Ибо, в противном случае, это означало бы, что ответы из NR^i относятся к классу распознаваемых ответов. Поэтому в функции *шага* φ для каждого шага диалога *весь класс нераспознаваемых ответов будем моделировать одним нераспознаваемым ответом*.

Проведенные рассуждения позволяют вернуться к циклу диалогового процесса, приведенному на рис. 1, и уточнить структуру набора ожидаемых ответов. Таким образом, на каждом шаге диалога набор ответов, который ожидает активный агент, может включать ответы из следующих классов ответов:

- класс распознаваемых ответов;
- класс нераспознаваемых ответов (моделируется одним ответом);
- класс детекторов вопроса (для определения потребности в смене ролей).

Простой пример вопросно-ответного диалога, рассмотренный ниже, преследует две цели: во-первых, оценить, насколько описанная конечно-автоматная модель вопросно-ответного диалога (формулы 1-7) полезна при конструировании диалогового метода доступа к памяти вопросов, и, во-вторых, обеспечить переход к последующей модели в виде сети Петри. Пример включает несколько шагов.

На шаге 1 активный агент генерирует вопрос с индексом Que_1 и ожидает получить следующие ответы:

Ans_1 — запрос на получение твёрдой копии вопроса;

Ans_3 — ответ, подразумевающий продолжение диалога и генерацию вопроса с индексом Que_3 ;

Ans_4 — ответ, подразумевающий продолжение диалога и генерацию вопроса с индексом Que_4 ;

Ans_5 — нераспознаваемый ответ (любой ответ, отличный от Ans_1, Ans_3, Ans_4). Ответ Ans_5 означает, что необходимо сформировать вопрос с индексом Que_2 .

На шаге 2 активный агент генерирует вопрос с индексом Que_2 , являющийся переформулированным вопросом с индексом Que_1 . На этом шаге активный агент ожидает получить те же ответы, что и на первом шаге, а также ответ Ans_2 , являющийся запросом на возвращение на шаг 1.

Диалоговый метод *DiAM*, для приведенного примера вопросно-ответного диалога, может быть описан при помощи следующих формул

$$ANS = \{Ans_1, Ans_2, Ans_3, Ans_4, Ans_5\}; \quad (8)$$

$$QUE = \{Que_1, Que_2, Que_3, Que_4\}; \quad (9)$$

$$S = \{S_1, S_2, S_3, S_4\}; \quad (10)$$

$$\varphi : \left. \begin{array}{l} (S_1, Ans_1) \rightarrow (S_1, Que_1), \\ (S_1, Ans_3) \rightarrow (S_3, Que_3), \\ (S_1, Ans_4) \rightarrow (S_4, Que_4), \\ (S_1, Ans_5) \rightarrow (S_1, Que_2), \\ (S_2, Ans_1) \rightarrow (S_2, Que_2), \\ (S_2, Ans_2) \rightarrow (S_1, Que_1), \\ (S_2, Ans_3) \rightarrow (S_3, Que_3), \\ (S_2, Ans_4) \rightarrow (S_4, Que_4), \\ (S_1, Ans_5) \rightarrow (S_2, Que_2). \end{array} \right\} \quad (11)$$

Функция шага φ , представленная отображениями (11), с одной стороны, описывает «логику» доступа к памяти вопросов, а с другой — «логику» диалогового метода решения некоторой задачи. Функцию шага можно представить в табличной форме. Табличное представление функции шага удобно тем, что позволяет легко перейти к графическому представлению вопросно-ответного диалога в форме диаграммы состояний [10]. На рис. 2 приведен пример графического представления вопросно-ответного диалога в виде диаграммы состояний.

В диаграмме состояний, приведенной на рис. 2, множество вершин соответствует множеству шагов вопросно-ответного диалога (или множеству устойчивых состояний), а каждая ветвь — одному из возможных вариантов развития диалога, определяемого ответом реактивного агента. Ветви помечены парами Ans_i, Que_i , где Ans_i — один из ожидаемых ответов; Que_i — индекс вопроса.

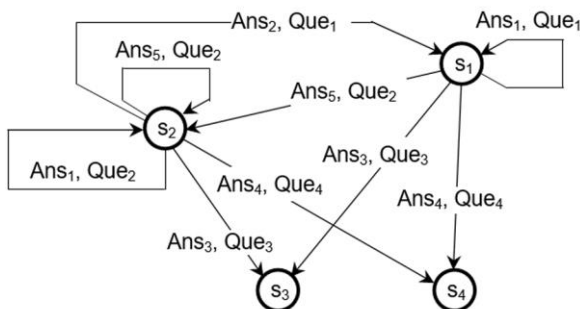


Рис. 2. Диаграмма состояний вопросно-ответного диалога для примера, описанного в тексте

Как было отмечено ранее, функция шага может рассматриваться как способ описания «логики» диалогового метода решения задачи, и в этом смысле диаграмма состояний, пример которой приведен на рис. 2, может рассматриваться как диаграмма, описывающая поведение активного агента в процессе решения задачи при помощи вопросно-ответного диалога.

3. Модель в виде сети Петри. Конечно-автоматная модель вопросно-ответного диалога имеет свою сферу применимости и может использоваться для специфицирования ряда прикладных диалоговых систем. Так, например, при помощи этой модели легко описывается метод внутреннего программирования предложенный Краудером и используемый при проектировании систем программированного обучения [11]. Однако, конечно-автоматная модель обладает рядом недостатков, проявляющихся на этапе её компьютерной реализации.

Эффективным подходом к компьютерной реализации вопросно-ответных диалоговых систем является *дatalogический подход*, который, в случае его применения к проектированию диалогового метода доступа *DiAM*, предполагает реализацию функции шага не в виде программного кода, а в виде отображения в некоторую базу данных. Решающим преимуществом даталогического подхода является возможность лёгкой модификации диалогового метода при помощи полноэкранных средств редактирования. Таким образом, даталогический подход исключает процесс компиляции исходных текстов при модификации системы и *позволяет выполнить модификацию диалогового метода решения задачи силами автора диалогового метода*.

С позиций даталогического подхода, формализм, принятый для описания диалогового метода необходимо отождествлять с концептуальной моделью некоторой базы данных. Как отмечено в [12], проблематика концептуальной модели базы данных связана с таким представле-

нием модели, которое, с одной стороны, наиболее естественным образом отражает предметную область (в нашем случае диалоговый метод доступа к памяти вопросов), а, с другой, — может поддерживаться компьютерными средствами. Иными словами, необходим формализм, легко трансформируемый в схему данных. Опыт проектирования диалоговых приложений позволяет сделать вывод, что отмеченным свойством в ббльшей степени обладает не диаграмма состояний диалогового метода доступа, (пример которой приведен на рис. 2), а граф, моделирующий метод доступа в виде сети Петри [13].

Петри-модель диалогового метода, так же, как и автоматная модель, базируется на трёх предположениях: (1) задача, решаемая в процессе диалога, может быть решена за конечное количество шагов диалога, (2) к началу решения задачи детерминированы все вопросы и ответы, необходимые для её решения; (3) модификация текущего набора ожидаемых ответов реактивного агента, в любом случае, порождает набор ответов, отличающийся от текущего и, следовательно, имеющий уникальный индекс. Петри-модель диалогового метода доступа опишем следующей четвёркой

$$DiAM = (QUE, ANS, NextQue, NextAns); \quad (12)$$

$$QUE = \{Que_i; i = 1, \dots, n. \quad (13)$$

QUE — комплект индексов вопросов, составленный из элементов множества (2), который допускается присутствие нескольких экземпляров одного и того же элемента [13].

$$ANS = \{Ans_i; i = 1, \dots, p. \quad (14)$$

ANS — комплект ответов, составленный из элементов множества (3).

$$NextQue : ANS \rightarrow QUE. \quad (15)$$

NextQue — функция индексов последующих вопросов. Поскольку на каждом шаге *активный агент генерирует только один вопрос*, функция *NextQue* для каждого ответа детерминирует индекс единственно го вопроса.

$$NextAns : QUE \rightarrow ANS. \quad (16)$$

NextAns — функция последующих ответов, отображающая комплект индексов вопросов в комплект ответов.

Для иллюстрации Петри-модели представим, описанный ранее, пример формулами (12)-(16).

$$ANS = \{Ans_1, Ans_2, Ans_3, Ans_4, Ans_5\}, \quad (17)$$

$$QUE = \{Que_1, Que_2, Que_3, Que_4\}. \quad (18)$$

$$\left. \begin{aligned} NextAns(Que_1) &= \{Ans_1, Ans_3, Ans_4, Ans_5\}, \\ NextAns(Que_2) &= \{Ans_1, Ans_2, Ans_3, Ans_4, Ans_5\}, \\ NextAns(Que_3) &= \{\}, \\ NextAns(Que_4) &= \{\}. \end{aligned} \right\} \quad (19)$$

$$\left. \begin{aligned} NextQue(Ans_1) &= \{Que_1\}, \\ NextQue(Ans_2) &= \{Que_1\}, \\ NextQue(Ans_3) &= \{Que_3\}, \\ NextQue(Ans_4) &= \{Que_4\}, \\ NextQue(Ans_5) &= \{Que_2\}. \end{aligned} \right\} \quad (20)$$

Формулы (19) и (20) можно использовать для табличного задания функций $NextAns$ и $NextQue$ соответственно. Правая часть i -ой строки функции $NextAns$ задает множество ответов реактивного агента, ожидаемых на i -ом шаге.

$$NextAns(Que_i) = R^i U NR^i. \quad (21)$$

Правая часть i -ой строки функции $NextQue$ состоит из одного элемента комплекта QUE , что является следствием отмеченного ограничения вопросно-ответного диалога, заключающегося в том, что для каждого ответа из комплекта ожидаемых ответов функция индексов последующих вопросов детерминирует индекс одного единственного вопроса.

При графической интерпретации Петри-модели диалогового метода, комплект индексов вопросов QUE и комплект ответов ANS представлены множествами позиций (окружности) и переходов (прямоугольники) соответственно. Множество ожидаемых на i -ом шаге ответов $R^i U NR^i$ представлено совокупностью переходов инцидентных i -ой позиции и соединённых с ней исходящими ветвями.

Для Петри-модели шаг является естественным «строительным блоком» диалогового метода доступа и для i -го шага может быть определён, как совокупность

$$Que_i, (R^i U NR^i). \quad (22)$$

На рис. 3 приведен граф Петри-модели диалогового метода, соответствующий примеру, описанному в тексте.

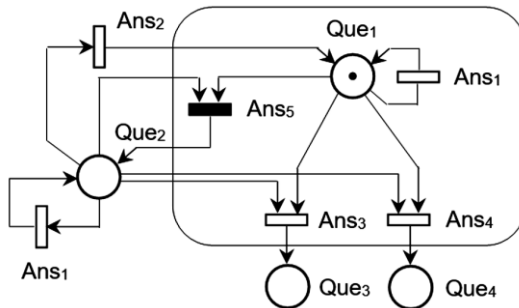


Рис. 3. Граф Петри-модели диалогового метода для примера, описанного в тексте. Зачернённый переход моделирует нераспознаваемые ответы

При графической интерпретации Петри-модели шаг диалога представляет собой позицию и совокупность переходов, соединённых с ней исходящими ветвями. На рис. 3 выделены элементы, входящие в объём понятия «шаг диалога».

Граф Петри-модели маркирован единственной фишкой, отмечающей текущий активный шаг. Начальная разметка, в этом случае, должна пониматься, как указание на первый шаг, с которого начинается реализация диалогового метода. Условием запуска перехода является совпадение ответа, кодируемого данным переходом, и ответа, полученного от реактивного агента. Иными словами, переход запускается и «пропускает» фишку, если от реактивного агента получен ответ, совпадающий с ответом, кодируемым данным переходом.

С точки зрения существующей классификации, сеть Петри, граф которой приведен на рис. 3, является ординарной (все дуги имеют одну и ту же кратность, равную единице) и не автоматной, поскольку Петри-модель допускает несколько входов для одного и того же перехода.

Теория сетей Петри предоставляет удобный аппарат для отслеживания динамики моделируемого процесса, состоящий из маркировки позиций и условий запуска переходов. Поскольку, в Петри-модели вопросно-ответного диалога, имеется единственная фишка, то маркировка может быть представлена вектором, количество компонентов которого равно количеству элементов комплекта индексов вопросов — числу n :

$$\mu = (\mu_1, \mu_2, \dots, \mu_n). \quad (23)$$

Каждый компонент вектора μ принимает значения на двухэлементном множестве $\{0,1\}$. $\mu_i = 0$ в том случае, если фишка отсутствует в i -ой позиции, соответствующей i -му элементу комплекта *QUE*. $\mu_i = 1$ в том случае, если фишка находится в i -ой позиции, соответствующей i -му элементу комплекта *QUE*.

Вопросно-ответный диалоговый процесс сопровождается перемещением фишки по сети без её уничтожения или размножения. Следовательно, в любой момент диалога имеет место $\sum \mu_i = 1$. Таким образом, в процессе вопросно-ответного диалога, диалоговый метод доступа характеризуется непрерывно изменяющимся вектором μ , который указывает на текущий, активный шаг диалога. Для сети, приведенной на рис. 3, вектор μ указывает на шаг номер один и имеет значение: $\mu = (1, 0, 0, 0)$. Для того, чтобы фишка переместилась в последующую позицию, необходимо, чтобы запустился один из переходов шага.

Представление диалогового метода в виде сети Петри (формулы 12-16) моделирует его пространственную структуру и не отражает динамики диалогового процесса, задаваемой, в общем случае, когнитивным циклом вопросно-ответного диалогового процесса, изобраа-

женным на рис. 1. Дополним структурное описание диалогового метода компонентами, моделирующими динамику диалогового процесса. С этой целью рассмотрим, каким образом в рамках предложенной Петри-модели реализуется цикл вопросно-ответного диалогового процесса. Моделирование цикла диалогового процесса будем рассматривать, исходя из предположения, что функции *NextAns* и *NextQue* представлены таблично.

Можно предположить, что отдельная строка таблицы функции *NextAns* соответствует одному шагу диалога. Каждая строка начинается с индекса вопроса, генерируемого на данном шаге, за которым следует множество ответов, ассоциированных с данным шагом. В этом случае таблица функции *NextAns* состоит из n строк, где n — количество элементов комплекта индексов вопросов. Количество столбцов таблицы функции *NextAns* определяется наибольшим возможным количеством ответов, ожидаемых от реактивного агента в процессе диалога. Поскольку множество ответов, ассоциированных с шагом, индивидуально для каждого шага и определяется количеством ожидаемых на данном шаге ответов, то часть ячеек в таблице *NextAns* остаются незаполненными.

Таблица функции *NextQue* имеет более простую структуру. Она состоит из p строк, где p — количество элементов комплекта ответов. Каждая строка таблицы функции *NextQue* содержит два элемента: ожидаемый ответ и связанный с ним последующий вопрос.

Введем в модель *унифицированный процесс DiMC*, моделирующий циклически повторяющуюся динамику диалога:

$$DiMC(\mu, NextAns, NextQue). \quad (24)$$

Унифицированный процесс *DiMC* управляет переходом от предыдущего шага диалога к последующему, что эквивалентно открытию одного из переходов шага и перемещению фишки. Унифицированность процесса *DiMC* означает, что для выполнения любого шага диалога необходимо выполнить одну и ту же последовательность действий, и, что эта последовательность действий не зависит от предметной области диалогового процесса. На содержательном уровне отдельную итерацию цикла *DiMC* опишем следующим образом.

1. При помощи вектора маркировки μ определяется текущий активный шаг диалогового процесса, который соответствует номеру строки в таблице функции *NextAns*.
2. Определяется и передаётся в память вопросов индекс текущего вопроса. При помощи спецификаций вопроса, находящихся в памяти вопросов, активный агент генерирует и передаёт реактивному агенту текущий вопрос. Активный агент переходит в состояние ожидания ответа реактивного агента.
3. Активный агент воспринимает текущий ответ реактивного агента.

4. Обработывается текущий ответ реактивного агента. Последовательно сравнивается воспринятый ответ со всеми, ожидаемыми на данном шаге ответами, хранящимися в ячейках выбранной строки таблицы функции *NextAns*. Для совпавшего ответа определяется строка таблицы функции *NextQue*.
5. При помощи таблицы функции *NextQue* определяется компонент вектора μ , который детерминирует следующий шаг диалога. Значение этого компонента изменяется с нулевого на единичное, и процесс возвращается к пункту 1.

Среди действий, итерационно выполняемых унифицированным процессом *DiMC*, есть два действия, требующие более детального рассмотрения и уточнения: (1) генерация текущего вопроса; (2) обработка текущего ответа. Поскольку *DiMC* является унифицированным процессом, который «единообразно» обрабатывает любой шаг, то, как генерация вопроса, так и обработка ответа осуществляются *DiMC* по одним и тем же правилам для любого шага диалога. Эти правила определяют некую *стандартную модель функционирования активного агента диалога*. Иными словами, *DiMC* неким стандартным способом генерирует вопрос и обрабатывает ответ. Ясно, что даже при достаточно обширном стандарте могут встретиться случаи, когда стандартных средств недостаточно. Например, при генерации вопроса, в том случае, когда он представлен не вербально, невозможно учесть все многообразие небинарной презентации информации, а при обработке ответа может понадобиться специфическая численная обработка. Поэтому, кроме базового процесса *DiMC*, в модель целесообразно ввести «внешние», по отношению к *DiMC*, процессы. Эти процессы должны вызываться из *DiMC* и возвращать ему управление после завершения. Назовём такие процессы процессами-демонами и введём в модель *два класса процессов-демонов*

QueDemon и *AnsDemon*,

которые осуществляют «дообработку» вопроса или ответа на любом шаге диалогового процесса в том случае, когда стандартных средств *DiMC* недостаточно [14].

Рассмотрим сферу применимости Петри-модели вопросно-ответного диалога. Под сферой применимости будем понимать разделение задач, решаемых в ходе диалогового процесса, на два класса: (1) легко реализуемых в рамках Петри-модели и (2) сложно реализуемых в рамках Петри-модели.

К сложно реализуемым, в рамках Петри-модели, задачам относятся, например, задачи, решаемые при проведении психодиагностических тестов. Как правило, на первом этапе психодиагностического тестирования (в процессе диалогового взаимодействия с субъектом диагностики)

формируется первичный набор данных, который затем используется для формирования заключения об интеллектуальных способностях или структуре интеллекта субъекта. Примером психодиагностического теста может служить методика прогрессивных матриц Равена, используемая для тестирования интеллектуальных способностей [15]. Диалоговый процесс, реализующий методику прогрессивных матриц Равена, обладает двумя специфическими особенностями. Во-первых, на каждом шаге диалога, реактивному агенту (субъекту тестирования) предъявляются *вопросы в невербальной форме*. Во-вторых, последовательность вопросов, передаваемых реактивному агенту, формируется активным агентом *без учёта ответов реактивного агента*.

Простой психодиагностический тест опишем следующим образом. На первом шаге генерируется вопрос Que_1 , на который ожидается один из ответов Ans_1 или Ans_2 . На втором шаге, *вне зависимости от полученного ответа*, генерируется вопрос Que_2 , на который ожидается один из ответов Ans_3 или Ans_4 . Заключение определяется в зависимости от комбинации полученных ответов. Петри-модель диалогового метода, реализующего описанный тест, изображена на рис. 4.

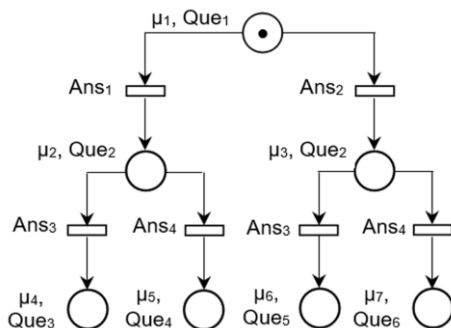


Рис. 4. Петри-модель диалогового метода, трудно реализуемого в рамках Петри-модели

В нижней части рис. 4 находятся *целевые позиции*, соответствующие окончанию диалога и формированию заключения. Трудность реализации Петри-модели проявляется в быстром увеличении количества шагов. Если на каждом шаге ожидается a ответов, и последующие вопросы не зависят от предыдущих ответов, то общее количество шагов определяется формулой

$$n = a^0 + a^1 + a^2 + \dots + a^{a-1}, \quad (25)$$

где a — количество вопросов. Петри-модель методики прогрессивных матриц Равена состоит из 2^{61} шагов.

Древовидная форма Петри-модели и большое количество шагов определяются тем, что модель учитывает комбинацию ответов путем

формирования в сети *уникальной траектории прохождения фишки*. Способом борьбы с этим недостатком является введение в модель памяти «истории ответов». Наличие такой памяти означает учёт сформулированного в разделе 1 принципа истории ответов, который означает, что диалоговый метод при формировании очередного вопроса должен учитывать как только что воспринятый ответ, так и ранее воспринятые ответы. Для упрощения изложения будем считать, что модель включает память *Mem*, хранящую *только один ответ, полученный на предыдущем шаге диалога*. Таким образом, содержимое памяти *Mem*: (1) автоматически обновляется при переходе к следующему шагу; (2) процесс *DiMC* при определении индекса следующего вопроса учитывает содержимое *Mem*. На рис. 5 приведена Петри-модель с памятью *Mem*, хранящей ответ, полученный на предыдущем шаге.

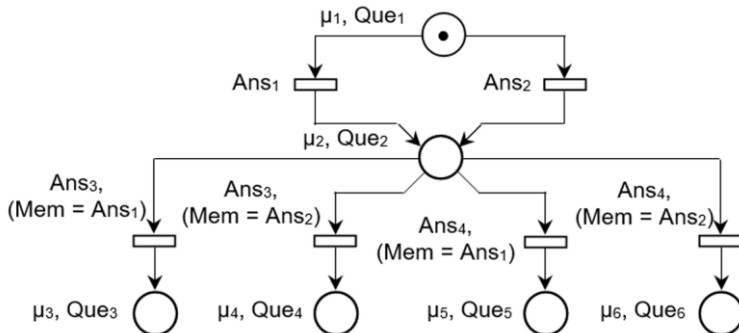


Рис. 5. Петри-модели при наличии памяти предыдущего ответа *Mem*

Включение в модель памяти предыдущего ответа не требует формирования уникальной траектории для каждого заключения. Уникальными остаются целевые позиции, а траектории, приводящие к целевым позициям, имеют общую часть.

Заключение. В статье описаны две формальные модели вопросно-ответного диалога: конечно-автоматная модель и модель в виде сети Петри. Модели могут быть использованы при проектировании «машины диалога», имитирующей целенаправленное поведение искусственного активного агента, не зависящее от предметной области диалога. Функционирование машины диалога основано на реализации унифицированного когнитивного цикла диалога. Модель вопросно-ответного диалога в виде сети Петри имеет более широкую сферу применимости, чем конечно-автоматная модель. Её преимущество в том, что она моделирует стандартный цикл диалогового процесса при помощи унифицированного процесса *DiMC*, а также возможность нестандартной обработки вопросов и ответов при помощи уникальных процессов *QueDemon* и

AnsDemon. Модель в виде сети Петри, также, позволяет учитывать принцип истории ответов реактивного агента.

Предполагается направить дальнейшие исследования на развитие главных идей описанных моделей, но отталкиваться не от математических структур в виде автомата Мили или сети Петри, а от идеи представления диалогового метода в виде сети, состоящей из *разнотипных узлов*. А priori предполагается, что сеть отображается в реляционную базу данных. В такой сетевой модели существенным должно быть следующее: (1) должны учитываться два из трёх, сформулированных в разделе 1, принципов организации вопросно-ответного диалога: принцип глубины диалога и принцип истории ответов реактивного агента; (2) должен развиваться диалогический подход и, следовательно, диалоговый метод решения задачи (диалоговый метод доступа к памяти вопросов), а также вопросы активного агента представляться данными, хранящимися в базе данных..

Список использованных источников:

1. Hintikka, J. Socratic Epistemology: Explorations of Knowledge Seeking by Questions. Cambridge: Cambridge University Press, 2007. 239 p.
2. Чмырь И. А. Принципы организации интеллектуальной САПР на основе диалоговой базы знаний. *Математические машины и системы*. Киев, 2014. № 3. С. 29-38.
3. Чмырь И. А. Электронный текст со встроенным интеллектуальным компонентом. *Математические машины и системы*. Киев, 2015. № 1. С. 72-83.
4. Bermudez J. Cognitive Science. An Introduction to the Science of the Mind. Second Edition. Cambridge: Cambridge University Press, 2014. 520 p.
5. Чмырь И. А. Естественный диалог: моделирование диалоговой транзакции в контексте представления знаний. *Интеллектуальные системы*. М.: Издательство Московского государственного университета. 2012. Т. 16. Вып. 1-4. С. 73-99.
6. Чмырь И. А. Моделирование активного агента эротетического диалога. *Математичне та комп'ютерне моделювання. Серія: Технічні науки : збірник наукових праць*. Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2014. Вип. 10. С. 206-217.
7. Asher N., Lascarides A. Logics of Conversation. Cambridge: Cambridge University Press, 2003. 220 p.
8. Найссер У. Теория восприятия. В сборнике Психология внимания / под ред. Ю. Б. Гиппенрейтера, В. Я. Романова. М.: Московский психолого-социальный институт, 2001. С. 640-648.
9. Чмырь И. А., Верлань А. Ф., Ус М. Ф. Когнитивные основы и концептуальный базис диалогового процесса. *УСiМ*. Киев, 2002. № 6. С. 54-60.
10. Брауэр В. Введение в теорию конечных автоматов. М.: Радио и связь, 1987. 219 с.
11. Ричмонд У. К. Учителя и машины. Введение в теорию и практику программированного обучения. М.: Мир, 1968. С. 57-64.

12. Цикриджис Д., Лоховски Ф. Модели данных. М.: Финансы и статистика, 1985. 254 с.
13. Питерсон Дж. Теория сетей Петри и моделирования систем. М.: Мир, 1984. 264 с.
14. Chmyr I. Dialogue of Partners as a Method of Non-Formal Problem Solving. In Maddy D. Brouwer-Janse and Thomas Harrington (eds). Human- Machine Communication for Educational Systems Design. NATO ASI Series F129. Berlin: Springer-Verlag, 1994. P. 222-228.
15. Raven J. Mental tests used in genetic studies: The performance of related individuals on tests mainly educative and mainly reproductive. *MSc Thesis, University of London*. 1936.

TWO MODELS OF QUESTION-ANSWERING «DIALOGUE MACHINE»

The article describes two models of question-answering dialogue. Both models describe the behavior of an active agent (an agent, who is asking questions) of a question-answering dialogue in the process of solving a problem by means of a dialogue process. In this sense, the models are intended to describe the script of a dialogue method for solving a problem. It is assumed that with the help of dialogue methods it is possible to solve problems, which belong to the class of ill-formalized. The first model is based on the idea of the Mealy state machine. It has a limited applicability, since when generating the next question, the history of previous answers of the reactive agent is not taken into account. The second model is based on the idea of a Petri net. The model takes into account the history of previous answers of the reactive agent, as well as the possibility of non-standard processing of the question-answering transaction using processes-daemons. The models are problem-independent and model a question-answering dialogue, which does not depend on the subject area of the dialogue and the nature of the problems solved by means of the dialogue method. The problem-independence of the models is due to the fact that the standard, cyclically repeated actions for processing a question-answering transaction reflect a unified cognitive dialogue cycle. The unified cognitive dialogue cycle is built on the basis of Neisser's cyclical model of perception. In the cognitive dialogue cycle, in contrast to the Neisser's cycle of perception, the source of the flow of sensory events is not the external environment, but the reactive agent. The models are designed to specify «dialogue machine» software that simulates the purposeful behavior of an active dialogue agent in the process of solving a problem. Both models assume that the main components of the «dialogue machine» are not mapped into a program code, but represented by data stored in a database. A dialogue machine is thus a software system that implements a script of a dialogue method by interpreting the data stored in a database.

Key words: *Question-answering dialogue, Dialogue agent, Dialogue script, Dialogue problem solving method, Cognitive cycle of dialogue.*

Отримано: 28.08.2020

УДК 519.626, 519.6, 621.276.12
DOI: 10.32626/2308-5916.2020-21.43-51

Andriy Verlan, PhD,
Jo Sterten, PhD

Norwegian University of Science and Technology, Gjøvik, Norway

INTELLIGENT OBJECT-ORIENTED APPROACH TO DYNAMIC ENERGY SYSTEMS' MODELLING

Proposed object-oriented approach and general architecture of intellectual software for mathematical modelling of dynamic energy systems. Introduced and considered in detail an architecture of knowledge bases for modelling of systems described by linear integral equations. Considered a knowledge-based system as a composition of a specific functional network and an expert system.

Key words: *modelling, simulation, dynamic systems, intelligent software, integral equations, knowledge base, functional network, expert system.*

Introduction and problem set up. New techniques for modelling of complicated energy systems are being rapidly developed. There is a broad spectrum of available simulation packages, however the main scientific problem which remains actual is a need in advancing of intelligent support for end-users. Here innovative mathematical modelling methods and corresponding intelligent software development concept should be applied. Generally, any scientific problem, including the indicated above, can be described by means of algebraic, differential, integral or other mathematical equations, graphs, logical descriptions etc.

There are many methods for dealing with these model descriptions, many of them have got their software implementation. If a user could specify exactly which method should be applied for solving current task, it would not encounter serious problems. The user could call appropriate procedure or even create a new code for the algorithm. But the typical problem is that it's unclear which proper method to use for solving the task, also the parameters of the method could be unknown. Frequently it requires deep knowledge in mathematics and modelling, and advanced capabilities to formalize the problem and to build a mathematical model. Then user deals with fuzzy descriptions instead of exact models.

Another problem is the following. There are many techniques of choosing and optimizing parameters of specific algorithms. These techniques are commonly implemented in the programs performing the algorithms. But it appears useful to move the parameters selection, optimization etc. from the level of specific programs to the kernel of the modelling environment. This approach makes program engineers' work significantly effective.

Therefore, the problem of creating advanced intellectual environments for solving practical problems of computer modelling of dynamic energy objects appears to be very urgent.

Object-oriented approach and intelligent software development concept used. According to the common approach, when user makes a query to the dialog system, the system should accomplish three stages:

- translation of the query to the inner language of the system;
- processing the query formulated in the inner language for getting answer;
- reverse translation of the answer formulated in the inner system language to the user's language.

For dynamic energy objects considered in this article, the language of mathematical formulae and mathematical equations can be regarded as the inner language of the modelling system. Therefore, we can consider the following proposed more detailed stages:

- translation of fuzzy user's query to a mathematical model;
- solving of the model for getting inner answer;
- reverse translation of the answer.

This paper is focused mainly on the second stage. Conversely to the first and the third stages, the second stage does not depend upon the subject. We consider a mathematical model being known. The problem then is to automate selection of algorithms and their parameters as well as choosing relevant forms of data representation.

Some general principles of creating knowledge bases on the algorithms for solving model equations given in [1]. According to the approach proposed in current paper, a functional network, which is the particular case of more common semantic networks, should be constructed. Here we consider application of this approach to the specific, however very powerful type of equations used for modelling, — to the linear integral equations of the following kind [2, 3]:

$$\int_a^b K(t, s) y(s) ds + y(t) = u(t). \quad (1)$$

As specified above, generally the main tasks in the problem of creating modern intellectual environments for dynamic energy objects' modelling are: translation of user's queries to the inner languages of mathematical models; and automation of selecting the proper methods and getting optimal parameters for these methods.

More formally, the solving of the user's query Q can be described as following composition:

$$R = L_S^- (U_M (L_S^+ (Q))), \quad (2)$$

where R is the system reply, L_S^+ is the translation of original query which depends upon the semantic context S . This semantic context depends on the knowledge about the subject, the user's features, the goals of the dialog, the language, the own opportunities of the system, etc. The result of this translation is the specific mathematical model. L_S^- is the inverse translation. U_M is the solving operator for the obtained model; it takes into account the knowledge base M about mathematical methods. This knowledge base doesn't depend on the subject. It contains knowledge about the representing of mathematical entities (functions, vectors, matrices, operators etc.), about the methods for solving specific equations, etc.

Here we consider the operator U_M only. To specify this operator firstly we should describe the knowledge base. So, the objective is developing the object-oriented approach which considers the processing as an interaction of some interconnected classes. We can introduce different levels of such description. Here we consider four levels: the *black box* level; the *generating process* level; the *functional network* level and the *data structures* level.

The black box level is the highest one. It deals with the connections determined by the operator equation

$$A\xi = \eta. \quad (3)$$

Both ξ and η are objects of the class FUNCTION but their roles in the equation are different. We introduce two roles: KNOWN and UNKNOWN. Then any function g can be represented as

$$g = IS_A(FUNCTION, Role, ID), \quad (4)$$

where ID is a set of parameters which distinguish this function from the others.

Then the connection is described in the following way:

$$Operator(IS_A(FUNCTION, Role1, ID1), IS_A(FUNCTION, Role2, ID2), IS_A(FUNCTION, Role3, ID3)). \quad (5)$$

The *Operator* can be regarded as a method of some other class. But it can be described as a specific class which depends upon the *Role*. This class is formed by three subclasses:

- Q is an operator for solving the direct problem which appears when $Role1 = KNOWN, Role2 = KNOWN, Role3 = UNKNOWN$;
- R is an operator for solving the inverse problem which appears when $Role1 = UNKNOWN, Role2 = KNOWN, Role3 = KNOWN$;
- S is an operator for getting A ; in this case $Role1 = KNOWN, Role2 = UNKNOWN, Role3 = KNOWN$.

The methods for solving specific equations and the software which implements these methods can be described as objects of the classes Q, R, S .

The next level is the level of generating processes. It depends upon the kind of equations. Here we consider the indicated above linear integral equations of the (1) kind.

There is a wide choice of methods and specific software modules for solving these equations (for instance given in [4-10]). According to the proposed approach given specific programs should be integrated into the knowledge base, and this base has to provide the user possibility/tools to solve the three types of tasks:

- direct tasks ($y(t)$ and $K(t, s)$ are known; $u(t)$ is to be found);
- reverse tasks ($u(t)$ and $K(t, s)$ are known; $y(t)$ is to be found);
- identification tasks ($u(t)$ and $y(t)$ are known; $K(t, s)$ is to be found);

The particular case of such equations is the Fredholm integral equations of the 1-st kind [11-14], which have many important applications, for example, the signal restoration problem. This problem is commonly formulated in the following way: to obtain a required signal from experimental data by means of solving an operator equation

$$Ay = u, \tag{6}$$

where u is an available experimental signal, y is a required/sought signal, A is a certain distorting operator, here we regard it as an integral operator.

This problem is incorrect since the inverse operator does not exist or isn't limited. We can apply methods of solving such problems based on specific regularization procedures. These procedures should significantly rely upon the available preliminary information about the solution. There are many procedures of this type but there is a lack of general recommendations how to apply them properly. The most common algorithms are Tikhonov method [15, 16], Lavrentiev method [17], etc.

The information model which describes this problem is formalized as a triplet $V_H = \langle Q, M, F \rangle$, where Q is a set of generating random processes; functions $u(t)$ and $y(t)$ may be regarded as realizations of these processes; M is a set of model spaces where the realizations may be projected to; F is a set of various dependencies between the elements of $\sigma(\Omega^* \cup M)$, where Ω^* is a set of all random processes' realizations from Ω , $\sigma(\Omega)$ is a set of all subsets of the set Ω .

Model spaces determined by some sets of coordinate functions are crucially important. These are, for example, spaces determined by Fourier, Karhunen-Loeve and other expansions.

The Karhunen-Loeve expansion is known to be an optimal linear expansion in respect of data compression. Moreover, Fourier spectral functions of non-stationery random processes depend upon two variables. Therefore, use of Fourier expansion for analysis of non-stationery processes encounters difficulties. Then the Karhunen-Loeve expansion might be applied instead of the Fourier one [18].

Another positive feature of the Karhunen-Loeve expansion appears when we are solving inverse tasks described by integral equations. This is just

the particular case considered in this article. If we expand input functions by the eigenfunctions of the kernel, then we can use eigenfunctions method which enables to solve inverse problems in a more convenient way.

The next level is the level of the functional network. Following the principles of conceptional programming [19], we develop a specific functional network for integral equations (1). Here we apply the object-oriented approach and consider our functional network as a composition of some basic objects.

The general architecture of the proposed functional network is represented in the figure 1.

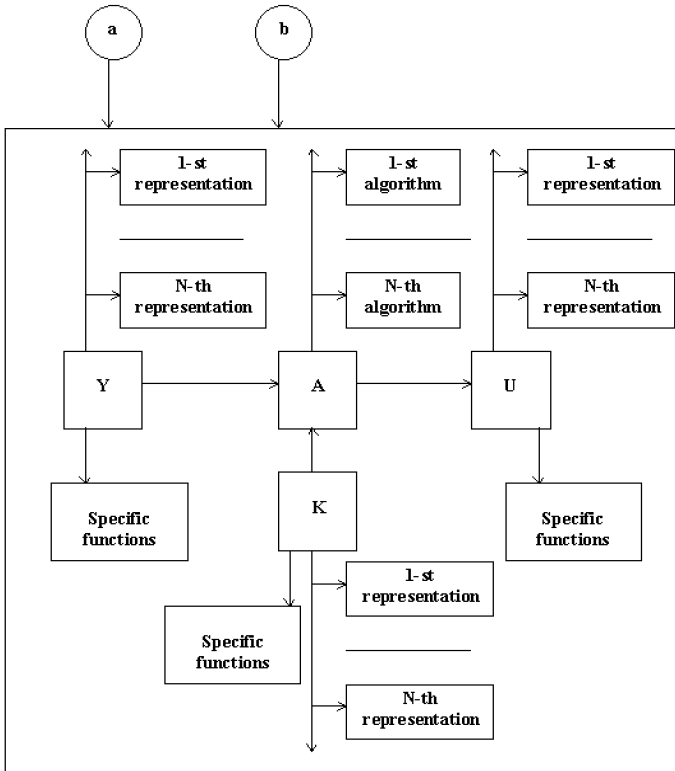


Fig. 1. General architecture of the functional network

The functional network, describing the whole model, can be seen as the top-level object. It consists of three basic objects Y , A , U and K . They describe $y(t)$, transformation operator, $u(t)$ and the kernel $K(t, s)$ respectively.

It is important, that the objects Y , U , K demand two lines of descendants. The first line is the representation line. It describes the different representa-

tions of the functions such as the values in the discrete points, spectral coefficients of Fourier, Karhunen-Loeve etc. The second line is the line of specific functions. Eventually, the descendants of the object A are the programs which perform the specific algorithms for solving direct and reverse tasks.

Functions $y(t)$, $u(t)$, $K(t, s)$ can be represented in the following way:

$$\text{Object} = IS_A(\text{FUNCTION}, \text{Role}, M, ID), \quad (7)$$

where M is a certain model space. Moreover, here we can introduce the new class $VECTOR$ which describes functions in some model spaces.

The nodes of the functional network are interconnected connected. The links are the following:

- links between different representations of the same function;
- links between input and output functions; these links can be realized by specific methods.

Then the accomplishment of the user's task can be considered as the searching of a path through the graph from an initial object to the target object.

This can be specified for each individual case. Thus, if the user wants to solve a direct task, he/she should specify the descendants of the Y by pointing them in interactive way. These descendants are the specific function and its representations. Then the user should specify the kernel. The linked objects are activated automatically until this reaches descendants of U .

If user wants to solve the reverse task, he should specify $u(t)$ and $K(t, s)$ in the same way. The activation is transmitted through the network in the reverse order.

Thus, this provides the selection of the proper method and its parameters. The environment must enable the user to specify the algorithm for solving the problem. The situation when the selected algorithm demands another representation of the input function is very typical, then the links between different representations of the same function should be involved.

Another general situation arises when there are different algorithms for solving the problem, or the user doesn't know which parameters are to be applied for the method. So, this demands another part of the knowledge base — an expert system. This expert system should accumulate an experience of solving integral equations and contain recommendations about applying specific algorithms. This may be, for example, a production system with the sort of rules as the following: «If the function has representation G with parameters H , the algorithm X with parameters Y should be applied».

For reflecting possible uncertainties, these rules may be fuzzy. Let us consider the case with one parameter only, other cases are similar. Then, the rules of the production system may be formulated, for example, in the following way:

$$G [a_1, a_2] \rightarrow (\mu_1 (H)), (\mu_2 (c)) H(c). \quad (8)$$

This means that if a function is represented in a model space G and its coordinate in this space varies from a_1 to a_2 , then the fuzzy function μ_1 is a degree of certainty that the algorithm H should be applied; the fuzzy function μ_2 is a degree of certainty that this algorithm should be applied with the parameter c . Then the method and its parameters may be determined by minimizing respective fuzzy functions. Here also should be considered the function which determines the rules of changing fuzzy functions while accumulating the experience.

On the data structures level, all structures of the classes should be specified in detail.

According to the general object-oriented approach proposed, we consider at least two categories of users dealing with the environment:

- end-users who apply the environment for solving their practical tasks;
- experts who determine rules of applying specific algorithms and links between the nodes of the functional network. Experts can also create programs for specific algorithms; the special language should be suggested for this purpose.

Now we can summarize the general structure of the intellectual modelling environment, which includes the following main modules:

- end-user's interface;
- expert's interface;
- interpreting system, which should provide translation of fuzzy formulated queries to specific mathematical models as well as return translation;
- knowledge-based system for mathematical modelling which is the combination of functional network (such network for the particular case was described above) and expert production system determining the use of the specific algorithms;
- simulation system for experimental investigations;
- archive database for storing results of experiments.

Conclusions. Thus, the concept and principles formulated above, particularly, proposed object-oriented approach and general architecture of intellectual software, introduced knowledge-based architecture for modelling the linear integral equations described systems as a composition of specific functional network and expert system, creates fundamentals for development of intellectual simulation program environment for solving practical problems of dynamic energy objects' computer modelling (using suggested program languages C++, Java and Matlab/Simulink).

References:

1. Do & Nguyen Hien & Mai. A Method of Ontology Integration for Designing Intelligent Problem Solvers. *Applied Sciences*. 2019. DOI: 9.3793.10.3390/app9183793.

2. Wazwaz A. Linear and Nonlinear Integral Equations: Methods and Applications. 2011. DOI: 10.1007/978-3-642-21449-3.
3. Verlan A., Sterten Jo. Implementation of Integral Explicit Macromodels by means of Quick-Acting Algorithms. *Mathematical and computer modelling. Series: Technical sciences*. 2018. Vol. 18. P. 26-33. DOI: 10.32626/2308-5916.
4. Burton T. A. Volterra integral and Differential Equations. 2nd ed. *Mathematics in science and Engineering*, 202. Elsevier, 2005.
5. Jerri A.J. Introduction to Integral Equations with Applications. Seconded. Jhon Wiley and Sons, 1999.
6. Nadir M. Solving linear integral equations with Fibonacci polynomial. *Malaya Journal of Matematik*. 2018. Vol. 6. P. 711-715. DOI 10.26637/MJM0604/0001.
7. Adawi A. Fadi A., Husein J. A Numerical Method for Solving Linear Integral Equations. *International Journal of Contemporar Mathematical Sciences*. 2009. Vol. 4. P. 485-496.
8. Kameda T. A general method for solving linear integral equations. II. *Proceedings of the Physico-Mathematical Society of Japan*. 2020.
9. Mirceski V. Modification of the step-function method for solving linear integral equations and application in modelling of a voltammetric experiment. *Journal of Electroanalytical Chemistry — J ELECTROANAL CHEM*. 2003. Vol. 545. P. 29-37. DOI 10.1016/S0022-0728(03)00086-X.
10. Verlan A., Sizikov V. Integral equations: methods, algorithms, programs. Kiev : Naukova dumka, 1986.
11. Brosy N. Fredholm integral equation. 2020. DOI: 10.1002/9783527809080.catanz07129.
12. Abdul-Majid W. Fredholm Integral Equations. 2011. DOI: 10.1007/978-3-642-21449-3_15.
13. Georgiev S. Generalized Fredholm Integral Equations. 2016. DOI: 10.2991/978-94-6239-228-1_5.
14. Altürk A. On multidimensional Fredholm integral equations of the first kind. *Journal of Inequalities and Special Functions*. 2017. Vol. 8. P. 85-95.
15. Machado M., Margotti F., Leitao A. On Nonstationary Iterated Tikhonov Methods for Ill-Posed Equations in Banach Spaces. 2018. DOI: 10.1007/978-3-319-70824-9_10.
16. Argyros I.K., Santhosh G., Shobha E. Discretized Newton-Tikhonov method for ill-posed hammerstein type equations. *Communications on Applied Nonlinear Analysis*. 2016. Vol. 23. P. 34-55
17. Favini A., Pandolfi L. Multiscale Lavrentiev method for systems of Volterra equations of the first kind. *Journal of Inverse and Ill-posed Problems — J INVERSE ILL-POSED PROBL*. 2008. Vol. 16. P. 221-238.
18. Wang L. Karhunen-Loeve expansions and their applications. 2008.
19. Hartley R. An Overview of Conceptual Programming I. 2020.

ІНТЕЛЕКТУАЛЬНИЙ ОБ'ЄКТНО-ОРІЄНТОВАНИЙ ПІДХІД ДО МОДЕЛЮВАННЯ ДИНАМІЧНИХ ЕНЕРГЕТИЧНИХ СИСТЕМ

Запропонований об'єктно-орієнтований підхід та загальна архітектура інтелектуального програмного забезпечення для математичного моделювання динамічних енергетичних систем. Представлено та де-

тально розглянуто архітектуру баз знань для моделювання систем, що описуються лінійними інтегральними рівняннями. Розглянутий система, заснована на знаннях у вигляді композиції конкретної функціональної мережі та експертної системи.

Ключові слова: моделювання, динамічні системи, інтелектуальне програмне забезпечення, інтегральні рівняння, база знань, функціональна мережа, експертна система.

Отримано: 3.09.2020

УДК 004

DOI: 10.32626/2308-5916.2020-21.51-60

О. О. Гордєєв*, канд. техн. наук,

К. П. Леонтєв**

* Університет банківської справи, м. Київ,

** Науково-виробниче підприємство «Радій», м. Кропивницький

МОДЕЛЬ ЖИТТЄВОГО ЦИКЛУ ДЕФЕКТУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Процес розробки програмного забезпечення включає в себе обов'язковий додатковий процес забезпечення якості програмного забезпечення, який являє собою сукупність заходів, що охоплюють всі технологічні етапи розробки, випуску та експлуатації програмного забезпечення інформаційних систем, що проводяться на різних етапах життєвого циклу програмного забезпечення для забезпечення необхідного рівня якості програмного забезпечення. Одне з основних завдань такого процесу полягає в знаходженні і усуненні дефектів програмного забезпечення. Дана робота присвячена формальному представленню життєвого циклу дефекту програмного забезпечення. Модель життєвого циклу дефекту програмного забезпечення розглядається як ланцюжок, який починається з помилки розробника і закінчується відмовою програмного забезпечення. У статті подається загальна структура моделі дефекту життєвого циклу програмного забезпечення, яка включає в себе помилку розробника, помилку оператора, прихований дефект у програмному забезпеченні, активний дефект у програмному забезпеченні, помилку обчислення, збій або відмову, породжену вразливістю, активовану вразливістю, несанкціоноване управління та несанкціонований доступ до даних. Така модель деталізується в набір патологічних ланцюжків, які структурно представляють модифікації життєвого циклу дефекту програмного забезпечення з урахуванням природи виникнення самого дефекту програмного забезпечення. Серед патологічних ланцюжків виділяють наступні: фізичний, проектування, розробки та взаємодії.

Патологічний ланцюжок взаємодії деталізується ще на патологічний ланцюжок взаємодії внаслідок фізичного впливу і патологічний ланцюжок взаємодії внаслідок інформаційного впливу. Дана модель життєвого циклу дефекту програмного забезпечення проєктується на V-подібну модель розробки програмного забезпечення, що дозволяє представити своєрідну еволюцію дефекту програмного забезпечення на кожному етапі життєвого циклу розробки програмного забезпечення окремо і в рамках всієї V-подібної моделі розробки програмного забезпечення.

Ключові слова: *якість програмного забезпечення, дефект програмного забезпечення, життєвий цикл дефекту програмного забезпечення, життєвий цикл програмного забезпечення, патологічні ланцюжки.*

Вступ. Розвиток існуючих і поява нових інформаційних технологій призводять до ускладнення життєвого циклу розробки програмного забезпечення. Чим складніші інформаційні технології, тим більш складним стає процес розробки програмного забезпечення (ПЗ). Така динаміка передбачає збільшення числа дефектів при розробці ПЗ. Дефект ПЗ — це результат, який має певні причини і наслідки. Причинами, як правило, є помилки розробників, а наслідки можуть виражатися в збоях або відмовах інформаційних систем.

Постановка задачі. Дефекти ПЗ аналізуються і досліджуються з моменту появи програмної інженерії як окремого інженерного напрямку [1]. З того часу викристалізувався і успішно розвивається в рамках програмної інженерії напрям забезпечення якості ПЗ. Було розроблено багато підходів, методів, методик і інструментальних засобів, спрямованих на забезпечення якості ПЗ. Не дивлячись на це, проблеми в частині появи дефектів ПЗ актуальні і сьогодні [2-4]. На думку авторів статті, у рамках дослідження дефектів ПЗ доцільно більш детально дослідити сам дефект ПЗ і описати його життєвий цикл в рамках процесу розробки ПЗ. Існуючі роботи в повній мірі не розглядають детально дефект ПЗ [5-6], у них взагалі не досліджені або не розглянуті не в повній мірі причинно-наслідкові зв'язки появи дефектів ПЗ [7-8], життєвий цикл (ЖЦ) дефекту ПЗ, а також його прив'язка до життєвого циклу розробки ПЗ [9-10]. У зв'язку з цим метою статті є розробка моделі життєвого циклу дефекту ПЗ і його проєкція на ЖЦ розробки ПЗ.

Життєвий цикл дефекту ПЗ. Життєвий цикл дефекту ПЗ буде розглядатися на основі наступної базової послідовності «помилка розробника — відмова ПЗ»: помилка розробника (оператора) (mistake), дефект в програмному забезпеченні (defect або bug), помилка обчислення при роботі програмного забезпечення (error), збій (fault), відмова (failure) (рис. 1).

Спіраючись на послідовність «помилка розробника — відмова ПЗ» сформуємо базову структуру життєвого циклу дефекту ПЗ. Така струк-

тура включає в себе наступні основні компоненти: причини, результат, наслідки та побічні наслідки (рис. 1). Перш за все, для виникнення результату — дефекту (defect, D) і помилки обчислення (error, E) повинна передувати причина або джерело, як правило, це помилки розробників або оператора (mistake, M). Дефект в програмному забезпеченні приводить до помилки обчислення. Наслідками помилки обчислення є збій (fault, F) або відмова (failure, FR). Варто також відзначити, що дефекти ПЗ можуть привести до виникнення уразливості (vulnerability, V), через яку зловмисники можуть отримувати несанкціонований доступ (unauthorized access, UA) до програмного забезпечення і в підсумку приводити до збою або відмови (рис. 2).

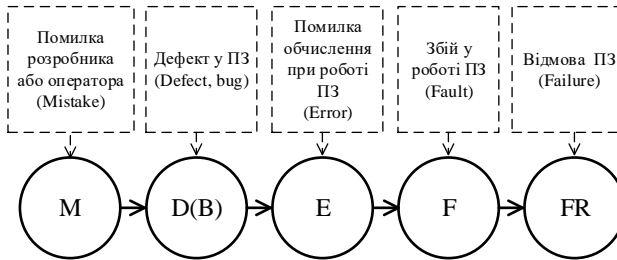


Рис. 1. Послідовність «помилка розробника — відмова ПЗ»

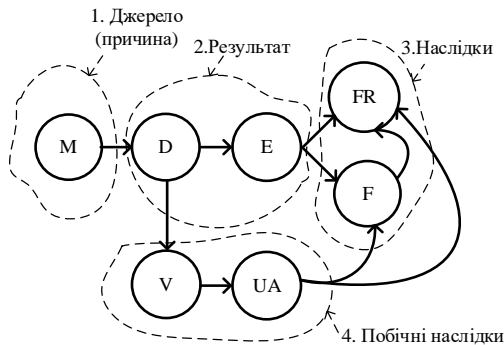


Рис. 2. Загальна структура моделі ЖЦ дефекту ПЗ

Таким чином, множина елементів життєвого циклу дефекту ПЗ (Software Defect Life Cycle, SDLC) складається з 7 елементів і матиме наступний вигляд (1):

$$SDLC = \{M, D, E, FR, F, V, UA\}. \quad (1)$$

Більш детальний аналіз ЖЦ ПЗ дозволив його деталізувати і сформулювати так звані модифікації ЖЦ ПЗ, які отримали назву «патологічні ланцюжки». Патологічні ланцюжки — це послідовність взаємопов'язаних подій, які можуть виникнути при розробці та використанні програм-

ного забезпечення людино-комп'ютерних систем, починаючи від помилки програміста (оператора) і закінчуючи відмовою інформаційної системи в цілому. Як правило, всі такі патологічні ланцюжки мають уніфіковану структуру, а відрізняються, перш за все, природою джерела виникнення ланцюжка (причиною). Розглянемо структуру патологічного ланцюжка. Вона включає в себе наступні елементи: помилку розробника (development mistake, DM), помилку оператора (operator mistake, OM), прихований дефект у програмному забезпеченні (hidden defect, HD), активний дефект в програмному забезпеченні (active defect, AD), помилку обчислення (error, E), збій (fault, F), відмову (failure, FR), породжену вразливість (created vulnerability, CV), активовану вразливість (activated vulnerability, AV), несанкціоноване управління (unauthorized control, UC), несанкціонований доступ до даних (unauthorized access, UA) (рис. 3).

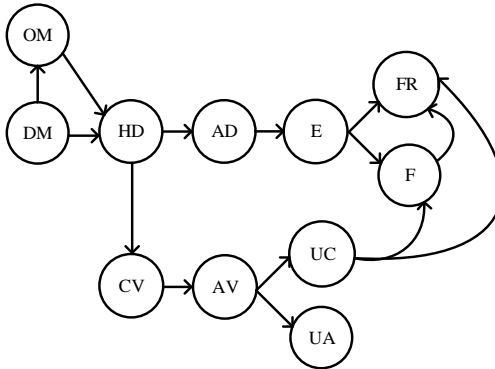


Рис. 3. Структура патологічного ланцюжка

Таким чином, множина елементів патологічного ланцюжка, модифікації життєвого циклу дефекту ПЗ, розширилася до 11 елементів і матиме вже наступний уточнений вид (2):

$$SDLC = \{OM, DM, HD, AD, E, FR, F, CV, AV, UC, UA\}. \quad (2)$$

Серед патологічних ланцюжків можна виділити фізичний, проектування, розробки та взаємодії. Розглянемо їх більш детально:

1. Фізичний патологічний ланцюжок. Природа виникнення джерела цього ланцюжка є фізичною, тобто це фізичні дефекти або несправності апаратного забезпечення. Елементи такого ланцюжка будемо позначати індексом — буквою р (physical) (рис. 4). Множина елементів такого ланцюжка практично ідентична множині елементів уніфікованої структури. У фізичного патологічного ланцюжка додається новий елемент — знос апаратного забезпечення (hardware wear, HR). Вважається, що знос апаратного забезпечення є також джерелом дефектів програмного забезпечення (рис. 4).

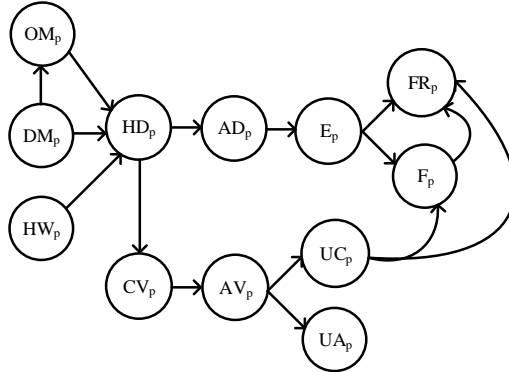


Рис. 4. Фізичний патологічний ланцюжок

Відзначимо, що множина елементів патологічного ланцюжка, модифікації життєвого циклу дефекту ПЗ, збільшилася до 12 елементів і матиме вже наступний уточнений вид (2):

$$SDLC_p = \{OM_p, DM_p, HD_p, AD_p, E_p, FR_p, F_p, HW_p, CV_p, AV_p, UC_p, UA_p\}. \quad (2)$$

2. Патологічний ланцюжок проектування. Природа виникнення джерела цього ланцюжка полягає в процесі проектування, тобто в дефектах проектування. Елементи такого ланцюжка будемо позначати індексом — буквами des (design) (рис. 5). Множина елементів патологічного ланцюжка проектування, модифікації життєвого циклу дефекту ПЗ, матиме наступний уточнений вид (3):

$$SDLC_{des} = \left\{ OM_{des}, DM_{des}, HD_{des}, AD_{des}, E_{des}, FR_{des}, F_{des}, CV_{des}, AV_{des}, UC_{des}, UA_{des} \right\}. \quad (3)$$

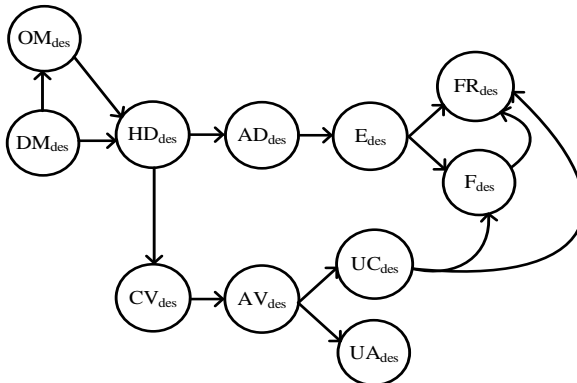


Рис. 5. Патологічний ланцюжок проектування

3. Патологічний ланцюжок розробки. Природа виникнення джерела цього ланцюжка полягає в процесі розробки, тобто дефектах розробки. Елементи такого ланцюжка будемо позначати індексом — буквами dev (development) (рис. 6). Множина елементів патологічного ланцюжка розробки, модифікації життєвого циклу дефекту ПЗ, матиме наступний вигляд (4):

$$SDLC_{dev} = \left\{ OM_{dev}, DM_{dev}, HD_{dev}, AD_{dev}, E_{dev}, FR_{dev}, F_{dev}, CV_{dev}, AV_{dev}, UC_{dev}, UA_{dev} \right\}. \quad (4)$$

4. Патологічний ланцюжок взаємодії. Природа виникнення джерела цього ланцюжка полягає в процесі взаємодії з інформаційною системою, тобто в дефектах взаємодії. Елементи такого ланцюжка будемо позначати індексом — буквою і (interaction) (рис. 7). Множина елементів патологічного ланцюжка взаємодії, модифікації життєвого циклу дефекту ПЗ, матиме наступний вигляд (5):

$$SDLC_i = \{ OM_i, DM_i, HD_i, AD_i, E_i, FR_i, F_i, CV_i, AV_i, UC_i, UA_i \}. \quad (5)$$

У загальному патологічному ланцюжку взаємодії можна виділити кілька його типів. До них відносяться:

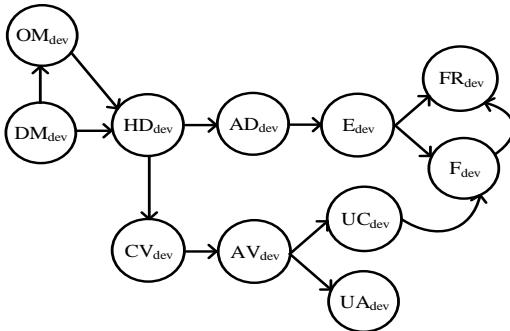


Рис. 6. Патологічний ланцюжок розробки

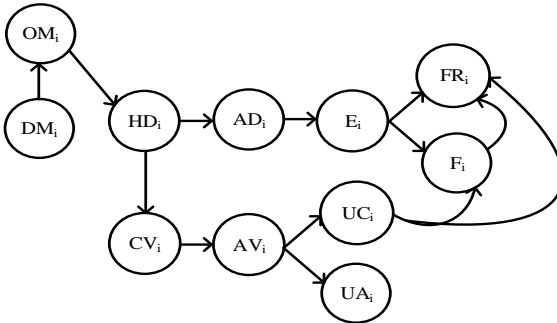


Рис. 7. Патологічний ланцюжок взаємодії

4.1. Патологічний ланцюжок взаємодії внаслідок фізичного впливу. Природа виникнення джерела цього ланцюжка в цілому аналогічна природі виникнення патологічного ланцюжка взаємодії, але з уточненням — внаслідок фізичного впливу. Елементи такого ланцюжка будемо позначати індексом — буквами pi (physical interaction) (рис. 8). Множина елементів патологічного ланцюжка взаємодії внаслідок фізичного впливу, модифікації життєвого циклу дефекту ПЗ, матиме наступний вигляд (6):

$$SDLC_{pi} = \left\{ OM_{pi}, DM_{pi}, HD_{pi}, AD_{pi}, E_{pi}, FR_{pi}, F_{pi}, CV_{pi}, AV_{pi}, UC_{pi}, UA_{pi} \right\}. \quad (6)$$

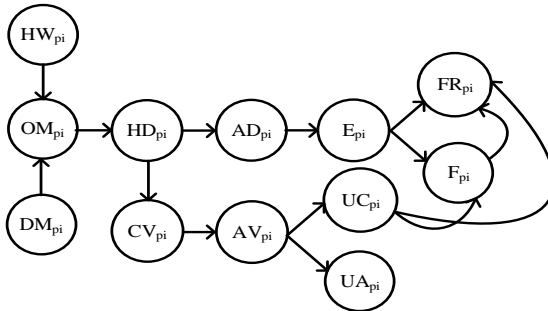


Рис. 8. Патологічний ланцюжок взаємодії внаслідок фізичного впливу

4.2. Патологічний ланцюжок взаємодії внаслідок інформаційного впливу. Природа виникнення джерела цього ланцюжка в цілому аналогічна природі виникнення патологічного ланцюжка взаємодії, але з уточненням — внаслідок інформаційного впливу. Елементи такого ланцюжка будемо позначати індексом — буквами ii (information interaction, ii) (рис. 9). Множина елементів патологічного ланцюжка взаємодії внаслідок інформаційного впливу, модифікації життєвого циклу дефекту ПЗ, матиме наступний вигляд (7):

$$SDLC_{ii} = \{ OM_{ii}, DM_{ii}, HD_{ii}, AD_{ii}, E_{ii}, FR_{ii}, F_{ii}, CV_{ii}, AV_{ii}, UC_{ii}, UA_{ii} \}. \quad (7)$$

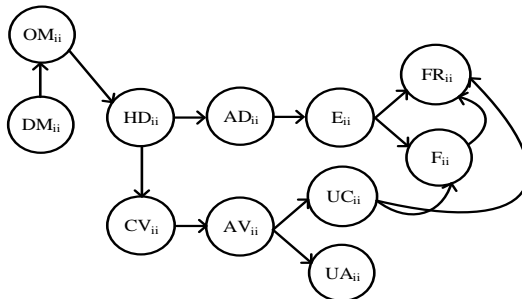


Рис. 9. Патологічний ланцюжок взаємодії внаслідок інформаційного впливу

Модель еволюції дефекту ПЗ з урахуванням V-подібної моделі розробки ПЗ. Варто відзначити, що ЖЦ дефекту ПЗ не завжди є лінійним (однорівневим), у тому вигляді, у якому він графічно представлений на рис. 1-9. Така нелінійність пов'язана з еволюцією дефектів ПЗ в рамках життєвого циклу розробки ПЗ. Наприклад, відповідно до V-подібної моделі дефект може з'явитися на першому етапі «1. Планування проекту та вимог» і бути дефектом вимоги ПЗ. При верифікації вимог на етапі «2. Аналіз вимог до продукту» такий дефект може бути невиявлений, тобто він стане прихованим дефектом і далі, еволюціонуючи, перейти на наступний етап розробки «3. Розробка архітектури» або ще далі — до етапу «5. Кодування». Така еволюція дефекту ПЗ може здійснюватися від етапу до етапу ЖЦ ПЗ, доти, доки він не проявиться при виконанні або тестуванні розроблюваного ПЗ (рис. 10). Таким чином дефект може еволюціонувати від HD_1 до HD_5 . На етапі «5. Кодування» еволюція дефекту ПЗ припиняється і на наступних етапах «6. Модульне тестування», «7. Інтеграція і тестування», «8. Системне приймальне тестування» і «9. Виробництво, експлуатація, супровід» він переходить від етапу до етапу тестування, як програмне забезпечення до моменту виявлення дефекту ПЗ.

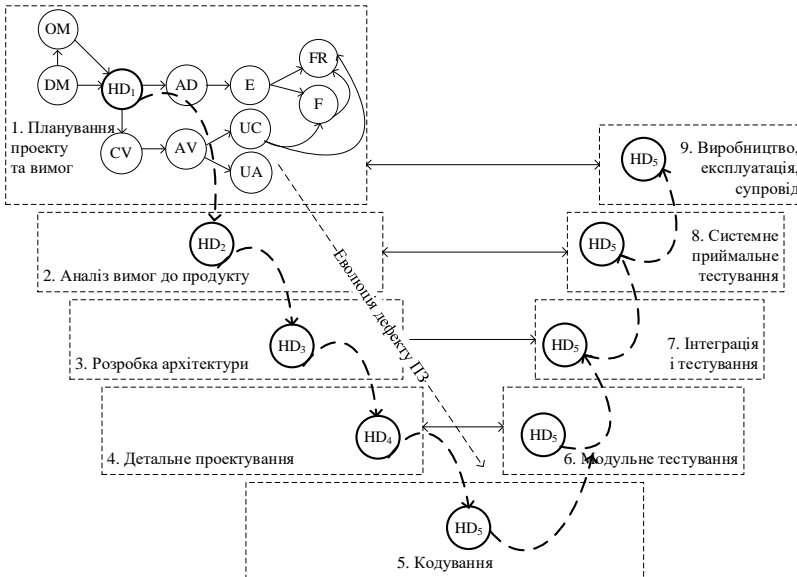


Рис. 10. Графічне представлення еволюції дефекту ПЗ з урахуванням V-подібної моделі розробки ПЗ

Висновки. Представлена в статті модель життєвого циклу програмного забезпечення дозволяє формально представити еволюцію

дефекту програмного забезпечення починаючи з причин, які привели до його виникнення і закінчуючи можливими наслідками. Більш детально представлення моделі життєвого циклу дефекту ПЗ у вигляді множини патологічних ланцюжків дозволяє врахувати особливості різноманіття станів дефекту ПЗ.

Подальші дослідження доцільно спрямувати на розвиток моделі ЖЦ дефекту ПЗ в напрямку оцінювання якості програмного забезпечення на основі засіву дефектів ПЗ. У цьому напрямку окремим завданням є дослідження життєвого циклу дефектів при розробці та реалізації процедур засіву дефектів (Fault Injection Testing, FIT), які використовуються у Науково-виробничому підприємстві «Радій» для оцінки функціональної безпеки FPGA-проектів для локальних інформаційно-керуючих систем АЕС. При цьому сам ЖЦ дефекту ПЗ може уточнитися, а кількість його модифікацій збільшитися.

Список використаних джерел:

1. Software engineering. Report on a conference sponsored by the NATO science committee. Garmisch, Germany, 7-11 October, 1968. URL: <http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF>.
2. Wagner S. Software Product Quality Control: book. Heidelberg: Springer. 2013. 210 p.
3. Gao J., Zhang L., Zhao F., Zhai Ye. Research on Software Defect Classification. In proceedings of the IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC-2019). URL: <https://ieeexplore.ieee.org/document/8729440/authors#authors>.
4. TOP 5 Software Failures of 2018–2019. Website Checkio. URL: <https://blog.checkio.org/%EF%B8%8F-top-5-software-failures-of-2018-2019-5-is-pretty-alarmed-2a5400b01658>.
5. Han W., Jiang H., Li W., Li Ye. A Summary of Software Defect Model. In proceedings of the IEEE 7th International Conference on Control and Automation (ICCA-2014). URL: <https://ieeexplore.ieee.org/document/7026263>.
6. Han W., Jiang H. Y., Lu T. B., Zhang X. Y., Li W. Software defect model based on similarity and association rule. International Journal of Multimedia and Ubiquitous Engineering. 2015. Vol.10(7). P. 1-10. URL: http://gvpress.com/journals/IJMUE/vol10_no7/1.pdf.
7. Fratini F., Pietrantuono R., Russo S. Reproducibility of Software Bugs. In: Fiondella L., Puliafito A. (eds) Principles of Performance and Reliability Modeling and Evaluation. Springer Series in Reliability Engineering. Springer, Cham. 2016. P 551-565. URL: https://link.springer.com/chapter/10.1007%2F978-3-319-30599-8_21.
8. Singh P. Learning from Software defect datasets. In proceedings of the 5th International Conference on Signal Processing, Computing and Control (ISPC-2019). URL: <https://ieeexplore.ieee.org/document/8988366>.
9. Rahman A., Nurdatillah H. Defect Management Life Cycle Process for Software Quality Improvement. In proceedings of the 3rd International

- Conference on Artificial Intelligence, Modelling & Simulation (AIMS-2015). URL: <https://ieeexplore.ieee.org/document/7604582>.
10. Koponen T. Life cycle of Defects in Open Source Software Projects. In: Damiani E., Fitzgerald B., Scacchi W., Scotto M., Succi G. (eds) Open Source Systems. OSS 2006. IFIP International Federation for Information Processing, vol 203. Springer, Boston, MA. URL: https://link.springer.com/chapter/10.1007%2F0-387-34226-5_19.

LIFE CYCLE MODEL OF SOFTWARE DEFECT

The software development process includes a mandatory additional software quality assurance process, which is a set of measures covering all technological stages of software development, release and operation of information systems, carried out at different stages of the software life cycle, to ensure the required level of software quality. One of the main tasks of this process is to find and eliminate software defects. This work is devoted to a formal presentation of the life cycle of a software defect. The life cycle model of the software defect is considered as a chain that begins with a developer error and ends with a software failure. The article presents the general structure of the life cycle model of software defect, which includes developer error, operator error, hidden software defect, active software defect, calculation error, fault or failure, created vulnerability, activated vulnerability, unauthorized control and unauthorized access to data. This model is detailed in a set of pathological chains, which structurally represent modifications of the life cycle of software defect, taking into account the nature of the software defect itself. Among the pathological chains are the following: physical, design, development and interaction. The pathological chain of interaction is detailed in the pathological chain of interaction due to physical influence and the pathological chain of interaction due to informational influence. The already mentioned model of life cycle of software defect is projected on a V-shaped model of software development, which allows to present a kind of evolution of software defect at each stage of life cycle of software development separately and within the whole V-shaped model of software development.

Key words: *software quality, software defect, software defect life cycle, software life cycle, pathological chains.*

Отримано: 19.09.2020

УДК 004.042:004.5

DOI: 10.32626/2308-5916.2020-21.61-75

О. А. Дячук*, канд. техн. наук,

Ю. О. Фуртат**, канд. техн. наук

* ДУ «Інститут економіки та прогнозування НАН України», м. Київ,

** Інститут проблем моделювання в енергетиці

імені Г. Є. Пухова НАН України, м. Київ

ПРОБЛЕМА НАДІЙНОСТІ ПРИ УЧАСТІ ЛЮДИНИ-ОПЕРАТОРА В ПРОЦЕСІ ПРИЙНЯТТЯ РІШЕНЬ ПО КЕРУВАННЮ ОБ'ЄКТАМИ ЕНЕРГЕТИКИ

В наш час розробка проблеми надійності складних систем здійснюється, зазвичай, з позицій чисто технічного підходу. У цьому підході в поле зору потрапляє лише «машинний» компонент складної системи, і підвищення надійності забезпечується за рахунок заходів, спрямованих на вдосконалення технічних ланок. Однак складні системи — це людино-машинні системи, надійність функціонування яких більшою мірою залежить від «людського фактора». Наприклад, на тепловій електростанції через помилки операторів відбувається близько чверті всіх порушень.

Відомо, що у зв'язку із широким впровадженням автоматизації у виробництво і успіхами кібернетики в 60-ті роки ХХ століття ставилося питання про повне виключення людини з контуру керування. Однак ідея повної автоматизації зазнала поразки, коли виникла необхідність у створенні пристроїв, які мали б здатність знаходити вихід з непередбачених ситуацій.

Висока гнучкість і здатність пристосовуватись до зміни умов, інтуїція, здатність до прогнозування, передбачення розвитку подій, прийняття рішень у складних ситуаціях при наявності декількох суперечливих або невизначених критеріїв є чисто людськими якостями. Це визначає місце людини в людино-машинних системах і характер покладених на неї функцій: прийняття рішень, а також включення резерву техніки на випадок її відмови. Однак відведення людині головної, інтелектуальної ролі в людино-машинних системах само по собі не знімає проблем, пов'язаних з їх надійністю.

Розрив між можливостями людини, як керуючої ланки, і зростаючими потребами системи в керуючих впливах аж ніяк не скорочується й змушує людину працювати на грані його «червоної межі». Однак інженерно-психологічний підхід дозволяє методологічно обґрунтувати шлях вирішення цієї проблеми.

Ключові слова: *людино-машинна система, автоматизована система, прийняття рішень, інженерно-психологічний підхід, когнітивний підхід.*

Вступ. Методологічною базою, що забезпечує можливість врахування ролі людини при вирішенні задач надійності систем енергетики, може служити інженерно-психологічний підхід до проектування людино-машинних систем (ЛМС). Згідно із цим підходом домінуючим компонентом будь-якої технічної системи вважається людина. «Машинна» частина, незалежно від ступеня її досконалості, розглядається як підлеглий елемент — знаряддя праці суб'єкта [1].

Ця концепція пронизує будь-які інженерно-психологічні заходи щодо аналізу і синтезу властивостей ЛМС. Так, зокрема, при проектуванні нових людино-машинних комплексів першим етапом стає складання проекту діяльності оператора. Цей проект виступає як фундамент для вирішення інших задач: від визначення принципової схеми до конкретного оформлення панелей, шкал приладів і т.д. [1, 2].

Головним критерієм оптимальності ЛМС повинно бути найкраще використання властивостей, властивих біологічним ланкам, у комбінації з перевагами, що одержуються системою завдяки наявності в ній технічних елементів і підсистем. Цей критерій можна інтерпретувати як принцип синтезу систем, що забезпечує підвищення надійності «людського» компонента. Суть його зводиться до наступного: компенсація недоліків людини за рахунок переваг машин, а також створення таких технічних засобів, які дозволили б повністю використовувати переваги людини [3]. Питання про компенсацію недоліків людини формулюється як задача про раціональний розподіл функцій.

Існують детальні переліки переваг у порівнянні з недоліками людей і машин, які дозволяють ефективно розв'язувати такі ситуації. Наприклад, комп'ютеру відводяться дії, що вимагають швидкодію, точність, стійкість до значних навантажень тощо. При розподілі функцій враховуються особливості виробництва, ступінь автоматизації системи, що розглядається, ієрархічний рівень, який займає в системі оператор.

Норми щодо питання розподілу функцій не є строго фіксованими. Вони залежать від рівня розвитку науки й техніки. Зі зростанням автоматизації виробництва оператор буде переходити на більш високі рівні керування. Тому вже зараз ставиться питання про необхідність створення техніки, яка забезпечувала б можливість максимальної реалізації творчих потенціалів людини. Мова повинна йти не стільки про обмежені здатності людини, скільки про обмежені можливості обчислювальних систем [3].

Основна частина. Одним з напрямків, що забезпечують вирішення зазначеної проблеми, є створення адаптивних людино-машинних комплексів. Синтез цих систем здійснюється на основі принципу адекватності, що передбачає узгодження основних конструктивних параметрів і «управлінських» характеристик біологічних і технічних елементів систем, а також принципу ідентифікації інформаційного середовища, що вимагає узгодження властивостей інформаційних потоків, які циркулюють між технічними і біологічними елементами.

Оскільки властивості людської ланки системи не інваріантні до впливу зовнішніх і внутрішніх факторів, такі системи мають подвійну адаптацію: до зміни зовнішнього середовища і ситуації (зовнішню адаптацію), а також «внутрішню адаптацію», коли технічні елементи системи або цілі підсистеми змінюють свої характеристики як і функції зміни стану біологічних елементів.

Залежно від конкретного ходу вирішення задачі, стану оператора і труднощів, з якими він зустрічається, технічна система виконує наступні функції: виробляє оптимальні форми представлення сигнальної інформації, «дає поради» у вигляді довідкових даних, інструкцій, підказок, аж до пред'явлення алгоритму деякого резервного розв'язку; нормалізує стан оператора шляхом інформаційного керування; відсторонює оператора у випадку втрати працездатності, переводячи функції керування на дублера або автоматизованої системи керування. Особлива увага в цих системах приділяється питанню подання інформації оператору [1-4].

Великий інтерес представляють адаптивні системи з динамічною моделлю ситуації й динамічною моделлю об'єкта, що дозволяють оптимізувати алгоритм керування. У цьому випадку інформація про ситуацію після надходження до системи класифікується (порівнюється з відомою системою); у випадку ж виникнення принципово нової ситуації модель ситуацій коригується блоком коригування і надходить у блок навчання, що вдосконалює програму розвитку ситуацій. Програма розвитку ситуацій пропонує через модель ситуації варіанти розвитку взаємин між керованим об'єктом і навколишнім середовищем. Оператор при цьому виробляє відповідні керуючі впливи, які реалізуються на динамічній моделі об'єкта й надходять у блок оцінки якості керування. У випадку досягнення оптимальної якості керуючих команд вони надходять у блок формування керуючих команд для реального об'єкта. Таким чином, в адаптивних системах є цілий ряд технічних засобів, які полегшують операторові вибір вірного розв'язку, залишаючи за ним право прийняття рішення.

Підвищення надійності ЛМС може бути забезпечене за рахунок удосконалення структури трудової діяльності оператора. Існуюча нині структура діяльності формувалася переважно стихійно. Відомо, що при традиційному підході до проектування систем це питання, як правило, не входить до числа задач, що розглядалися. Про людину згадують на етапі експлуатації системи, коли виявляється неузгодженість «людської» і «машинної» ланок. Це приводить у найкращому разі до проведення заходів щодо пристосування техніки до людини, а в найгіршому разі до «адаптації» людини під техніку, що може створити неприйнятні умови роботи для оператора й стати причиною появи систематичних помилок і «зривів» на рівні людської ланки.

Шляхом проб і помилок, на базі багаторічного досвіду експлуатації систем поступово сформувалася сучасна структура діяльності оператора,

яка не позбавлена недоліків. І це не випадково, оскільки під контролем оператора при керуванні складними системами перебувають тисячі параметрів. Наприклад, під контролем оператора енергоблоку сучасної теплової електростанції (ТЕС) перебувають близько 1500 різних параметрів, що характеризують роботу парогенератора, турбіни й електричного генератора, 69 автоматичних регуляторів, 361 об'єкт дистанційного керування, 90 технологічних захистів і близько 300 сигналізацій на табло.

Однак, як показали дослідження, ця структура далеко не досконала. Зокрема, алгоритм діяльності людини-оператора, що здійснює формування дискретних команд керування, містить наступні операції: перегляд інформаційної моделі (системи відображення інформації); перенесення погляду на моторне поле, зоровий пошук необхідного органу керування; переміщення руки до органу керування; перемикання органу керування; перенесення погляду на інформаційну модель; контроль результату. Аналіз даного алгоритму дозволив виявити ряд операцій [5-7], що характеризуються високою ймовірністю помилкових дій і більшим часом виконання. Крім того, ці операції є причиною виникнення фактора психологічної складності в роботі оператора — кількаразового відволікання уваги людини від системи відображення інформації.

У зв'язку із цим був розроблений новий алгоритм діяльності, у якому дані операції відсутні. У цьому випадку структура діяльності включає всього три етапи:

- 1) перегляд інформаційної моделі;
- 2) перемикання органу керування;
- 3) контроль результату дії по сигналу зворотного зв'язку.

Також відсутній фактор психологічної складності, тому що увага оператора постійно сконцентрована на інформаційній моделі об'єкта. Як показали експериментальна перевірка й оцінка функціонування підсистеми «людина — пристрій введення інформації», робота оператора за даним алгоритмом забезпечує підвищення надійності системи в цілому.

Як відзначалося вище, проект діяльності оператора є основою всього інженерно-психологічного проектування. Відповідно до цього синтезований алгоритм діяльності «підказав» загальну схему перспективного класу пристроїв вводу. Таким чином, ще на етапі проектування системи було вирішено питання про узгодження «машинної» і «людської» ланок.

Забезпечення надійної роботи системи керування за участю людини включає кілька взаємозалежних шляхів. Поряд з раціональним розподілом функцій, пристосуванням машини до людини, удосконаленням структури діяльності велике значення має шлях підвищення надійності «людської» ланки за рахунок проведення заходів щодо професійного відбору й навчання операторів.

Ці методи спрямовані на формування контингенту осіб, що володіють психологічними якостями, необхідними для успішного ви-

конання роботи. Так, наприклад, до числа професійно важливих якостей оперативного персоналу відносяться: розвинене оперативне мислення, стійка концентрація й розподіл уваги, низька стомлюваність, гарна пам'ять, стійкість до впливу стрес-факторів [8].

Існує велика кількість тестів і методик діагностування психофізіологічного стану людини [9-13]. Для відбору операторів застосовуються як професійні методи, так і оперативне тестування. При оперативній діагностиці психофізіологічного стану оператора перевіряється час реакції, увага, логічне мислення, координація роботи з периферійними пристроями робочого місця, уміння ухвалювати розв'язки в умовах швидкої зміни обстановки. Одним з найбільш популярних підходів при цьому є проведення тестування для осіб, що випробовуються і одержанні діагностичних параметрів, що порівнюються потім з еталонним зразком необхідного фахівця.

Найбільш комплексний аналіз професійної придатності може бути проведений із залученням до тестування досвідчених психологів, однак фахівців високого класу поки ще недостатньо, тоді як кількість фахівців-операторів безупинно зростає. У зв'язку із цим великого значення набувають автоматизовані системи тестування. При цьому такі системи повинні вирішувати задачі, пов'язані як з розробкою тестів, так і з їхньою комп'ютерною реалізацією [14].

Слід особливо виділити оперативне тестування на робочому місці, оскільки воно, з одного боку, повинно забезпечувати задану глибину тестування, а з іншого — враховувати обмеження, які є в комп'ютеризованому робочому місці оператора.

По-перше, тестування не повинно відволікати оператора від виконання основної функції на робочому місці, а тому повинно проводитися у фоновому режимі (за винятком початкового тестування при допуску до робочого місця).

По-друге, у силу комп'ютеризованості робочих місць операторів для більшості спеціальностей самі тести також повинні являти собою комп'ютерну програму, що вимагає максимальної формалізації процесу побудови тестів і власне тестування. Формалізація також дозволяє уникнути надмірності тестів і інформації, що отримується в результаті тестування.

Методика тестування й адаптація операторів при цьому виглядає так:

- 1) перед початком роботи здійснюється перевірка стану оператора, висновок про допуск до роботи робиться за результатами тесту;
- 2) якщо результати тесту виявляються незадовільними, то проводиться адаптація оператора шляхом повторення тестів кілька раз із урахуванням результатів кожного тесту;
- 3) періодично у фоновому режимі перевіряється стан оператора.

Методика й програмні засоби оперативного тестування є основою також і для професійного відбору. Крім того, вони служать для одержання еталонних значень діагностичних параметрів і дослідницької роботи з оцінки якості тестування. Програмна реалізація оперативного тестування організована за модульним принципом, що дозволяє змінювати й нарощувати кількість тестів залежно від конкретних вимог і умов тестування.

При цьому й самі тести, які є основою оперативного тестування, повинні бути реалізовані програмно.

Основними етапами створення програмного тестового забезпечення є:

- 1) вибір і формування складу тестів;
- 2) розробка програмних засобів реалізації тестів;
- 3) формування мінімального простору інформаційних діагностичних параметрів;
- 4) одержання еталонних значень діагностичних параметрів і правил прийняття рішень;
- 5) формування й видача інформації про результати тестування.

Пункти 1 і 4 тісно пов'язані із психологічними аспектами (наприклад, можна використовувати існуючі методики професійного відбору); пункти 2 і 5 відносяться до програмування. Багато в чому вони залежать від специфіки комп'ютеризованого робочого місця, на якому буде проводитись тестування; пункт 3 розкриває можливість оптимізації процесу тестування за рахунок вибору мінімально необхідного набору параметрів і використання ефективних алгоритмів роботи з ними.

Діагностичні параметри. Перспективним є вибір таких параметрів оцінки психофізіологічного стану: відношення оператора до ризику, оперативна пам'ять, увага. При цьому час реакції на тестовий вплив і помилки у відповідях повинні дозволити оцінювати зібраність і координацію дій оператора, оскільки вони залежать від цих особистісних властивостей.

В основі методики оцінки відношення оператора до ризику лежать такі поняття як виправданий і невиправданий ризик, надмірна обережність. Результат залежить від імовірності настання несприятливих наслідків у випадку ухвалення певного рішення і їх величини.

Методика оцінки оперативної пам'яті оператора полягає у відстеженні здатності оператора запам'ятовувати послідовності образів (символів, цифр) і реагувати на доповнення послідовності (наприклад, вказуючи скільки разів зустрічається певний образ).

Методика дослідження обсягу уваги значною мірою подібна оцінці оперативної пам'яті, але від оператора потрібно прийняти певний розв'язок на підставі інформації, що короткочасно відображається на дисплеї робочого місця оператора.

Самі методики вже досить добре вивчені й математизовані [10, 11], що дозволяє без особливих зусиль реалізувати їх як частину програмного забезпечення.

Розглянемо окремо методику оцінки кожного з названих параметрів.

Тест для діагностики відношення оператора до ризику. Велику роль при прийнятті рішення відіграє ступінь виправданості ризику [10] — імовірності того, що здійснення деякої поведінки спричинить певні несприятливі наслідки для суб'єкта. Поведінку, що пов'язана з деяким ступенем ризику, прийнято називати ризикованою.

Ризик — явище об'єктивне, і суб'єкт може здійснювати ризиковану поведінку навіть не підозрюючи про це. Суб'єктивним же може бути не ризик як такий, а лише оцінка його ступеня. Причому суб'єктивною є не будь-яка оцінка, а та, яка приналежна саме тому, хто здійснює ризиковану поведінку.

Суб'єктивна оцінка ступеня ризику може, як збігатися з об'єктивною, так і відхилятися від неї в більшій або меншій мірі. При цьому під об'єктивною оцінкою мається на увазі оцінка, яка може бути обчислена на основі аналізу фізичних (а іноді й психологічних) факторів ситуації, що можуть стати або вже стали джерелом несприятливих наслідків. Істотну роль в об'єктивному оцінюванні відіграє врахування відносної частоти виникнення несприятливих наслідків в умовах багаторазового стереотипного повторення даної або аналогічних ситуацій у минулому.

Класифікація ризику за параметром виправданості-невиправданості не залежить також від ступеня усвідомленості суб'єктом ризикованості дії. Можливі випадки виправданого прийняття ризику, заснованого на інтуїції, що виключає можливість усвідомлення всіх істотних аспектів ситуації. Тривале ж обмірковування ситуації може призвести до прийняття об'єктивно невиправданого ризику.

Поряд з виправданим і невиправданим ризиком, існує також невиправдана обережність. Таким чином, при оцінці відношення оператора до ризику, крім усього іншого, необхідно одержати досить точні відповіді на два питання:

- 1) чи є в оператора на даний момент схильність до невиправданого ризику;
- 2) чи володіє він здатністю піти на виправданий ризик.

У нашому випадку головною функцією математичної моделі ситуації ухвалення рішення є чітке розмежування ситуацій виправданого й невиправданого прийняття ризику, а також умов, при яких дії оператора слід класифікувати як прояв надмірної обережності.

Тест для діагностики оперативної пам'яті. Оперативна пам'ять визначається як пам'ять, де інформація повинна зберігатися до закінчен-

ня виконання якоїсь (поточної) операції, після чого пам'ять очищається для прийняття і зберігання нової порції інформації. У чомусь вона подібна короткочасній пам'яті, але існують і відмінності.

Основною відмінною рисою оперативної пам'яті є її своєрідна перманентність. Оперативна пам'ять, зазвичай, є досить тривалим і неперервним процесом, у якому операції сприйняття, збереження, переробки, відтворення й забування певних порцій інформації переплітаються й чергуються один з одним. При цьому об'єктом запам'ятовування стають не тільки сигнали, що надходять ззовні, але й власні попередні дії оператора, у тому числі продукти уявної й інженерної переробки сприйнятої раніше інформації. Спряженість оперативної пам'яті з іншими психічними процесами, її включення до діяльності щодо керування складними виробничими і транспортними системами обумовлює особливо високі вимоги до точності і надійності її функціонування.

При цьому методика оцінки якості (діагностики) оперативної пам'яті в цілому доволі проста. Від особи, що тестується потрібно лише швидко й точно вказати з появою на екрані деякого символу з набору — скільки разів даний символ уже з'являвся на екрані. Після правильної відповіді послідовність продовжиться, і на екрані з'явиться новий символ. У випадку ж помилки, послідовність почнеться спочатку (або ж буде видане попередження для декількох перших помилок). При цьому фіксуються наступні величини:

- кількість коректно виконаних мнемонічних операцій — C ;
- тривалість безпомилкового функціонування — T сек.;
- темп виконання завдання $A = \frac{C}{T}$;
- ефективність роботи $E = \frac{C^2}{T}$.

Тест дослідження обсягу уваги оператора. У ході цього тесту визначається можливість оператора короткочасно утримувати в пам'яті ситуацію на екрані й видавати керуючий вплив згідно із цією ситуацією. Одночасно контролюється координація й швидкість реакції оператора.

При цьому сама методика тестування, як і в попередньому випадку, проста. Оператору на короткий час демонструється квадратне поле із сірих клітинок, частина яких позначена яким-небудь додатковим кольором, утворюючи «змійку». Потім усі клітки стають сірими, і оператору пропонується послідовно відновити «змійку» за допомогою керуючих клавіш. З кожним успіхом «змійка» стає довшою. У випадку помилки випробування починається спочатку (або ж буде видане попередження для декількох перших помилок). Довжина останньої успішно відновленої «змійки» і швидкість, з якою оператор впорався із завданням, дозволяють оцінити його обсяг уваги.

Можна видозмінювати завдання, експериментуючи з різними кольорами і їх комбінаціями, що дозволяє додатково оцінити сприйнятливість оператора до певного кольору, або ж замінити «змійку» довільною комбінацією зафарбованих полів (упорядкованих, наприклад, за допомогою номерів або порядку появи на екрані).

Розв'язуючі правила. Крім задачі створення тестів, існує задача обробки їх результатів. Для цього необхідно одержати еталонні значення діагностичних параметрів, з якими будуть порівнюватися результати тестування, і побудувати розв'язуючі правила, що дозволяють обґрунтовано ухвалити рішення щодо стану людини, що тестується.

Один із способів полягає в експериментальному одержанні еталонних значень діагностичних параметрів (профілю спеціальності). Для цього необхідні «еталонні» фахівці, професійна придатність яких визначена іншим способом.

У загальному вигляді одна із процедур одержання еталонних значень складається з наступних етапів:

- 1) формується група фахівців, що добре вирішують професійні завдання;
- 2) в результаті перевірки цієї групи на множині тестів визначаються математичне очікування і коваріаційна матриця діагностичних параметрів;
- 3) на основі апроксимації області розсіювання діагностичних параметрів будуються розв'язуючі правила.

Більш складна процедура одержання розв'язуючих правил полягає у виділенні групи, що містить два класи фахівців — «гарні» і «погані» фахівці. На основі обробки результатів тестування цієї групи можна побудувати розв'язуючі правила, що дозволяють із заданою ймовірністю віднести особу, що тестується до класу «поганих» або «гарних» фахівців. У цьому випадку для обробки результатів тестування залучаються методи теорії розпізнавання образів [12, 15]. Однак сама методика побудови правил не відрізняється від вище описаної.

Автоматизована система тестування, таким чином, може бути представлена у вигляді програмної надбудови над робочим місцем оператора СППР, що складається із власне модуля тестування (початкового і фонового) і бази знань, що містить профілі операторів і розв'язуючі правила. При цьому повинна бути можливість наповнення бази знань як ззовні (результати професійного психологічного тестування), так і в результаті тестування операторів.

Для наочності така надбудова представлена у вигляді діаграми потоків даних (рис. 1).

Програмно така система може бути реалізована за допомогою резидентного модуля (для фонового тестування необхідна постійна активність паралельно із процесами робочого місця оператора) або бібліотеки,

що динамічно підключається (тоді виклик може відбуватися безпосереднього з робочих програм автоматизованого робочого місця оператора).

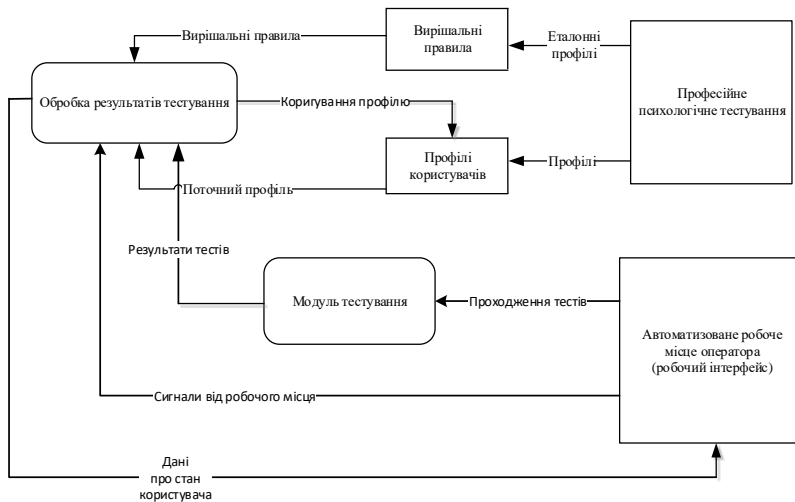


Рис. 1. Діаграма потоків даних автоматизованої системи тестування

При цьому, у зв'язку зі зростаючою потребою автоматизованого виробництва в операторах, основне завдання профвідбору полягає не у звуженні контингенту осіб, що відбираються, а в його розширенні, що може бути забезпечене за рахунок ретельного контролю психофізіологічних якостей людей і складання прогнозу щодо можливості їх подальшого розвитку. Такий підхід використовується, зокрема, у навчально-тренувальному центрі підготовки операторів теплових енергоблоків, створеному при Трипільській ГРЕС.

У цьому центрі розроблена й радикальна програма підготовки операторів, реалізована за допомогою сучасних засобів навчання з використанням обчислювальної техніки. Традиційний спосіб підготовки операторів, що займає 2-3 роки і полягає в проходженні «новачком» усіх службових сходинок — від чергового по насосах до оператора енергоблоку, замінюється тут трьома етапами навчання. На першому етапі за допомогою програмованих посібників і технічних засобів (проекторів, магнітофонів тощо) під керівництвом інструкторів засвоюються знання про керувану техніку, режими роботи окремих агрегатів і всього енергоблоку в цілому. На другому етапі на спеціальних тренажерах виробляються оперативні вміння по керуванню окремими вузлами енергоблоку. На третьому етапі — на тренажері-імітаторі енергоблоку здійснюється об'єднання знань і навичок, освоєння стратегії керування [16].

В операторів, що пройшли навчання по цій програмі, спостерігається зниження числа аварій в 4-4,5 рази [17].

Однак зниження числа аварій в ЕЕС призводить до зниження майстерності операторів. Без практики в них притупляється гострота реакції, погіршується оперативне мислення і як наслідок, губиться кваліфікація й професіоналізм. Тому необхідні відповідні тренування оперативного персоналу в умовах, близьких до реальних, на тренажерах, при розборі конкретних рольових ігор і т.п. Підтримці робочого тону оператора сприяють і поточні протиаварійні тренування, які проводяться в різних ланках ЕЕС. Наприклад, щоквартальні диспетчерські (системні й мережні), станційні, цехові, підстанційні тренування, а також тренування на окремих ділянках.

Керує тренуванням досвідчений оператор з більшим стажем роботи. Їм задаються початкові умови аварії і визначається її подальший розвиток. У випадку здійснення невірних дій або їх значної затримки, що також прирівнюється до помилки, ситуацію ускладнюють і роблять більш важкою. Гра протікає в реальному масштабі часу й будується відповідно до закономірностей розвитку аварій у системах.

Основною емоційно-вольовою якістю, яку повинен мати оператор, що керує енергетичним об'єктом, є готовність до екстрених дій. Оператор повинен уміти мобілізувати всі свої сили в момент настання аварії [8].

Широке впровадження автоматики у виробництво перетворює оператора в пасивний елемент. Цьому сприяє й підміна реального об'єкта інформаційною моделлю (мнемосхемою, табло) керованої машини, що відчужує оператора від реального об'єкта. Реальна мета дій — керування об'єктом — зберегти або створити задане розташування індикаторів на приладовій панелі, що знижує значимість основної мети й міру відповідальності за вчинені дії.

Пасивний оператор, що втратив пильність, може не тільки пропустити момент настання аварії й зробити грубі помилки, але й, як свідчать факти, відмовитися продовжити роботу і залишити пункт управління в момент найбільшої небезпеки. У зв'язку із цим в інженерній психології виникла концепція активного оператора. Згідно із цією концепцією, керування об'єктом повинно бути комбінованим, тобто здійснюватися й автоматично, і вручну. У випадку явної відсутності необхідності в ручному керуванні оператору потрібно час від часу задавати ігрові ситуації, які він повинен відпрацьовувати як реальні. Інші рекомендації зводяться до активізації діяльності по прогнозуванню ситуації: візуально сконцентрувати увагу оператора на тих зонах інформаційної моделі, які дозволяють припустити погіршення поточної ситуації, і зобов'язати оператора програти на моделі об'єкта можливі варіанти виходу з неї.

Сучасна техніка, як правило, обслуговується не окремими операторами, а групами операторів. Процес керування технікою — це не проста сума паралельних дій, які виконуються окремими операторами незалежно один від одного. Його істотним моментом є взаємозв'язок і взаємодія операторів. При цьому ступінь узгодженості дій

операторів суттєво впливає на надійність складних систем [18]. У правильно скомплектованій групі, що включає оптимальну кількість працівників і керівника (неформальний лідер групи) і має продуману організаційну структуру, забезпечується нормальний психологічний клімат, трудова активність особистості й погодженість дій операторів усієї групи. Таким чином, надійність складної системи може бути підвищена за рахунок соціально-психологічних заходів, спрямованих на вдосконалення трудових колективів.

Практика експлуатації систем показує, що проведення перерахованих заходів щодо врахування «людського» фактора не може гарантувати абсолютну надійність «людської» ланки ЛМС. Людина завжди залишається людиною: у будь-якого здорового оператора, що володіє необхідними професійними знаннями, навичками і уміннями, можуть наступати несприятливі психічні стани, що ведуть до більшого або меншого порушення якості його роботи.

«Шкала» цих станів має широкий діапазон: від стану підвищеного збудження (ейфорії) до крайнього ступеня стомлення [19].

Ефективним способом підвищення надійності і ефективності діяльності оператора, що базуються на «машинній», а не на «людській» частині ЛМС, є адаптивні комплекси. Вони також дозволяють додатково персоналізувати роботу оператора в готовій системі керування — в той час як інші «технічні» інженерно-психологічні методи застосовуються в основному на етапі проектування систем.

Самі технічні елементи систем керування (ЕОМ, елементи візуального представлення, системи датчиків тощо) досить важко адаптувати до конкретних операторів, але це можна зробити з інформаційними потоками, що надходять від них.

Експериментально встановлено, що мозок людини здатен сприймати і точно обробляти інформацію зі швидкістю не більше 25 біт за секунду. Перевищення цього значення може призвести до феномена «перманентної часткової уваги», зниження зосередженості.

Проблема ефективного сприйняття інформації людиною активно вирішується для технічних систем, особливо для керування системами військового призначення. Основним підходом до розв'язання проблеми було застосування ергономічних методів, розрахованих на деякі середні показники можливостей і умов для людини сприймати інформацію від програмно-технічних засобів: гострота зору, час фіксації сигналу, розташування інформації, яка надається, частота миготінь, яскравість, розмір, кодування, тип шрифту, структурований і неструктурований текст (у т.ч. довжина повідомлень — 7-11 слів), переваги до представлення у вигляді діаграм, графіків [20].

Застосування ергономічних методів на етапі розробки систем дійсно дозволяє частково розв'язати проблему інформаційного перевантаження операторів, але при сучасних значних обсягах інформації

й необхідності прийняття термінових і важливих рішень в режимі реального часу цього недостатньо. Тому й виникла необхідність в адаптивному керуванні інформаційним комплексом.

Збільшення зручності взаємодії користувача і комп'ютера фокусується на представленні інформації. Одна зі стратегій представлення даних — адаптивна, полягає в співставленні інтерактивної поведінки системи з індивідуальними потребами користувачів, тобто в адаптації інтерфейсу.

Оператор взаємодіє із зовнішнім середовищем, сприймаючи символічну інформацію, обробляючи й кодує її для зберігання в довгостроковій пам'яті, на основі знань у якій він і будує свою поведінку в конкретній ситуації.

При цьому слід враховувати, що інформаційний потік, що надходить до оператора, має певні характеристики (інтенсивність, рівень складності, форма представлення тощо), як і знання в довгостроковій пам'яті оператора (визначаються особистими якостями оператора, рівнем його підготовки).

В дослідженнях зі сприйняття нових знань неодноразово підкреслювалося, що респонденти засвоюють той самий матеріал з різною швидкістю і з різною якістю [21-24]. Причиною є віднесення людини до різних типів особистості. Відмінності у швидкості і якості визначаються: а) каналами сприйняття (візуальний, аудіальний, кінестетичний) б) стилями мислення (конкретний-абстрактний, індуктивний-дедуктивний тощо). Із цього погляду, не може бути універсально зрозумілої інформації й універсального ефективного способу асиміляції знань.

Індивідуальні особливості переробки інформації, стратегій засвоєння, структуризації, організації й актуалізації засвоєного матеріалу можуть бути зведені до переваги операторами тих або інших перцептивних і когнітивних стратегій. У цьому випадку вони є проявом когнітивних стилів — конструктів для позначення образів сприйняття, мислення і дії, які переважно використовуються людиною [25].

Висновки. Когнітивно-орієнтований підхід підкреслює стабільний вплив когнітивного стилю на організацію і представлення інформації в процесі мислення і навчання. Коли певний стиль збігається зі змістом і представленням інформації, яку необхідно сприйняти, людина сприймає завдання як більш легке, ніж тоді, коли стиль і представлення не збігаються. У тому випадку, коли існує така розбіжність, людина може полегшити завдання шляхом розвитку стратегій навчання для роботи з інформацією, яка спочатку була несумісна з її стилем.

Система, у свою чергу, може «піти назустріч» оператору, змінюючи властивості інформаційних потоків для більшої відповідності когнітивному стилю конкретного оператора. Для реалізації такого механізму може бути створена автоматизована керуюча система, що працює з базою когнітивних стилів, індивідуальних особливостей

операторів, що аналізує і модифікує інформаційні потоки системи керування. Очевидно, що оптимально така система буде працювати в якості надбудови над інтерфейсною частиною систем керування.

Полегшення сприйняття оператором інформації від системи знизить рівень інформаційного стресу і дозволить приймати ефективніше правильні рішення по керуванню системою.

Реалізації на основі існуючих систем керування автоматизованих систем оперативного тестування оператора й адаптації інформаційних потоків до його когнітивних особливостей здатні значно підвищити надійність «людського» компонента ЛМС (на додаток до інших інженерно-психологічних методів).

Список використаних джерел:

1. Інженерна психологія: теорія, методологія, практичне застосування / під ред. Б. Ф. Ломова та ін. М.: Наука, 1977. 182 с.
2. Крилов А. А. Людина в автоматизованих системах керування. Л.: Вид-во ЛДУ, 1972. 192 с.
3. Смолян Г. Л. Людина і комп'ютер. М.: Политиздат, 1981. 192 с.
4. Ахутін В. М. Адаптивні біотехнічні системи. *Психологічні проблеми взаємної адаптації людини і машини в системах керування*. М.: Наука, 1980. С. 77-123.
5. Меньшов І. А. Основи інженерної психології. Л., 1969. 171 с. (ЛВіА ім. А. У. Можайського).
6. Галактіонов А. І. Основи інженерно-психологічного проектування АСУ ТП. М.: Енергія, 1978. 208 с.
7. Нілерян Е. А. Емоційно-вольові компоненти надійності оператора. *Нариси психології праці оператора*. М.: Наука, 1974. С. 5-83.
8. Питання професійної придатності оперативного персоналу енергосистем / під ред. Б. М. Теплова, К. М. Гуревича. М.: Освіта, 1966. 219 с.
9. Методика і техніка досліджень операторської діяльності / відп. ред. В. М. Волков. М.: Наука, 1985.
10. Завдання і методи професійної психодіагностики / під ред. В. І. Войтко, Ю. З. Гільбуха. Київ, 1981.
11. Воробйов Г. Автоматизована система профорієнтаційного тестування на ПЕОМ. *АН СРСР. Наукова рада з комплексної проблеми «Кібернетика»*. М.: Препринт, 1988.
12. Анастасі А., Урбіна С. Психологічне тестування. СПб., 2001. 688 с.
13. Бодалев А. А., Столін В. В. Загальна психодіагностика. СПб., 2000. 440 с.
14. Гадецька З. М., Ус М. Ф. До питання про автоматизацію побудови тестів успішності навчання. *Матеріали XXVI науково-технічної конференції «Моделювання»*. Київ: ІПМЭ НАНУ, 2007. С. 20-21.
15. Русін В. П. Структурно-лінгвістичні методи розпізнавання зображень у реальному часі. Київ: Наукова думка, 1986. 128 с.
16. Аварії по сценарію. *Наука й життя*. 1982. № 9. С. 90-96.
17. Тренажерні системи. М.: Машинобудування, 1981. 256 с.
18. Основи інженерної психології. М.: Высш. шк., 1977. 335 с.
19. Гримак Л. Д., Пономаренко В. А. Психічні стани й надійність діяльності оператора. *Питання кібернетики*. М., 1982. С. 145-155.

20. Кляуззе В. Ергономічне забезпечення Web-Дизайну. URL: <http://www.osp.ru/rcworld/2002/05/163334/>.
21. Брусиловський П. Л. Моделі учня, в інтелектуальних навчальних системах. *УСiМ*. 1992. № 7/8. С. 109-119.
22. Петрушин В. А. Експертно-навчальні системи. Київ: Наук. думка, 1992. 196 с.
23. Сороко В. М., Журавльов О. В. Автоматизовані навчаючі системи з елементами штучного інтелекту. Київ: НМК В, 1992. 244 с.
24. Петрушин В. А. Інтелектуальні навчальні системи: архітектура й методи реалізації (огляд). *Технічна кібернетика*. 1993. № 2. С. 164-189.
25. Лібин А. В., Париліс С. Є. Сильові характеристики індивідуальності. *Методологічні й теоретичні проблеми сучасної психології*. М., 1988. Вып. 2. С. 119-129.

THE PROBLEM OF RELIABILITY WITH THE PARTICIPATION OF A HUMAN OPERATOR IN THE DECISION-MAKING PROCESS FOR THE MANAGEMENT OF ENERGY FACILITIES

Nowadays, the development of the problem of reliability of complex systems is carried out, as a rule, from the standpoint of a purely technical approach. In this approach, only the «machine» component of a complex system is in the field of view, and the increase in reliability is provided by measures aimed at improving the technical units. However, complex systems are human-machine systems, the reliability of which depends to a greater extent on the «human factor». For example, at a thermal power plant due to operator errors, about a quarter of all violations occur.

It is known that in connection with the widespread introduction of automation in production and the success of cybernetics in the 60s of XX century, the question of the complete exclusion of man from the control circuit. However, the idea of full automation was defeated when there was a need to create devices that would have the ability to find a way out of unforeseen situations.

High flexibility and the ability to adapt to changing conditions, intuition, the ability to predict, predict developments, decision-making in difficult situations in the presence of several conflicting or uncertain criteria are purely human qualities. This determines a person's place in the human-machine systems and the nature of the functions assigned to him: decision-making, as well as the inclusion of a reserve of equipment in case of failure. However, assigning a person the main, intellectual role in the human-machine systems does not remove the problems associated with its reliability.

The gap between the capabilities of man as a control link and the growing needs of the system in control influences is not reduced and forces a person to work on the verge of his «red line». However, the engineering-psychological approach allows to methodologically substantiate the way to solve this problem.

Key words: *human-machine system, automated system, decision making, engineering-psychological approach, cognitive approach.*

Отримано: 10.09.2020

УДК 519.61/.64:627.05

DOI: 10.32626/2308-5916.2020-21.76-82

Н. В. Іванчук, канд. техн. наук,**П. М. Мартинюк**, д-р техн. наук, професор,**В. Л. Филипчук**, д-р техн. наук, професор

Національний університет водного господарства та природокористування, м. Рівне

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ОЧИЩЕННЯ ВОДИ В БІОПЛАТО-ФІЛЬТРИ

Біоплато — споруда для очищення господарсько-побутових, виробничих стічних вод, забрудненого поверхневого стоку. У відомих конструкціях біоплато відбувається поступова кольматація порового простору фільтрувальної засипки та нижнього дренажу біоплівкою і мінералізованим осадом, накопичення мулу в донній частині споруди. Для відновлення роботи біоплато необхідна їх періодична зупинка на складні і тривалі ремонтно-відновлювальні роботи, пов'язані з промивкою і регенерацією фільтрувальної засипки та дренажу. Тому прогнозування роботи біоплато актуально проводити саме на етапі проектування з метою здійснення прогнозних розрахунків. Доцільніше це робити саме засобами математичного та комп'ютерного моделювання без проведення натурних експериментів. В даній роботі авторами побудовано математичну модель фільтрації з урахуванням кольматації та суфозії в системі біоплато-фільтра у двовимірному випадку. В побудованій математичній моделі враховано фізичні ефекти динамічної зміни пористості та залежності коефіцієнта фільтрації від концентрації забруднень, чого немає у відомих аналогах. Нелінійність побудованої моделі не дозволяє говорити про аналітичні розв'язки відповідної крайової задачі. Більше того, якщо в подальшому розглядати тривимірну задачу. Математичне моделювання дозволяє оцінити ефект впливу розглянутих факторів, але не дозволяє знизити такий негативний вплив. Тому для збереження продуктивності біоплато авторами запропоновані інженерні рішення, які знижують вплив кольматаційно-суфозійних процесів на ефективність роботи біоплато-фільтра. А саме: в математичній моделі авторами враховано наявність додаткової системи перфорованих труб, яка розміщена в товщі фільтрувальної засипки. Урахування вказаних факторів дозволяє підвищити адекватність математичної моделі для досліджуваних фізичних процесів. Числові розв'язки задачі знайдено методом скінченних елементів. Проведення чисельних експериментів планується у безкоштовному програмному середовищі FreeFem++.

Ключові слова: *біоплато-фільтр, проблема фільтрації, кольматація, суфозія, метод скінченних елементів, система перфорованих труб.*

Вступ. В останні роки біоплато набувають значного поширення і використовуються для очищення та доочищення господарсько-побутових, виробничих стічних вод, поверхневого стоку у різних країнах світу [8, 9]. Це пояснюється тим, що вони практично не потребують витрат електроенергії та хімічних реагентів, значного експлуатаційного обслуговування і забезпечують необхідну якість очищення води від широкого спектру забруднюючих речовин органічного та мінерального характеру [1-3].

Однак у відомих конструкціях біоплато відбувається поступова кольматація порового простору фільтрувальної засипки та нижнього дренажу біоплівкою і мінералізованим осадом, накопичення мулу в донній частині споруд, зниження надходження кисню до кореневої системи рослин, що може призводити до зниження ефективності роботи споруд, пептизації осаду і вторинного забруднення води. Особливо негативним є подача на такі біоплато води з підвищеним вмістом завислих частинок, що значно підсилює негативні процеси, які протікають у товщі фільтрувальної засипки.

Аналіз літературних джерел та постановка дослідження. По суті, потрібно досліджувати процес міграції нерозчинених частинок в пористих середовищах. На важливості процесу міграції твердих частинок в пористих середовищах наголошується в роботі [15], оскільки разом з ними можуть мігрувати і віруси, асоційовані з цими частинками. Математичну модель переносу суспензії в пористих середовищах побудовано та застосовано її до процесів суфозії в роботі [16]. В роботі [4] використано модель перенесення (система диференціальних рівнянь першого порядку) та дифузійну модель, коли отримуємо вироджене рівняння параболічного типу. В роботі [16] досліджено перенесення наночастинок заліза в пористих середовищах і їх використання для очищення забрудненого ґрунту та ґрунтових вод. Результати лабораторних досліджень по кольматації та перенесенню частинок в пористому середовищі наведені в роботі [7].

В деяких із вищезгаданих наукових роботах побудовано математичну модель фільтраційно-кольматаційних процесів. Основна увага там приділена процесам перенесення наночастинок та кінетики кольматаційних процесів. Однак, в процесі кольматації відбувається динамічна зміна пористості пористого середовища, що, в свою чергу, впливає на коефіцієнт фільтрації і, опосередковано, на весь процес фільтрації. Урахування вказаних ефектів та модифікація відповідного рівняння фільтрації здійснено авторами у роботі [12].

У роботі [13] авторами розроблено концептуальні підходи та математичну модель для потенційної енергії органічних речовин при оцінці природних систем, яка дозволяє обчислити загальну кількість

первинної біомаси в об'ємі води або поверхневих квадратних одиницях. В роботі [5] вдосконалено математичну модель процесу аеробної очистки стічних вод. Він враховує взаємодію бактерій, а також органічних та біологічно неокислюючих речовин в умовах дифузійних та масообмінних збурень. А в роботі [14] автори використали електрокоагуляція для очищення різних промислових стічних вод.

У роботах [10, 11] авторами досліджено метод біо-повільної піщаної фільтрації для усунення помутніння, органічних забруднень, бактерій та аміачного азоту у фільтрувальній воді. Ця технологія очищення води є актуальною, оскільки характеризується низьким споживанням енергії, простою експлуатацією та високою швидкістю видалення забруднень.

Виклад основних результатів. В якості модельної задачі розглядався біоплато-фільтр довжиною 50 метрів та висотою 2 метри із щебенистою засипкою крупністю фракцій 20 мм. Забруднена вода подається через верхню дренажну систему із перфорованих труб і забирається в нижній частині засипки за допомогою перфорованого дренажу, розміщеного на дні біоплато.

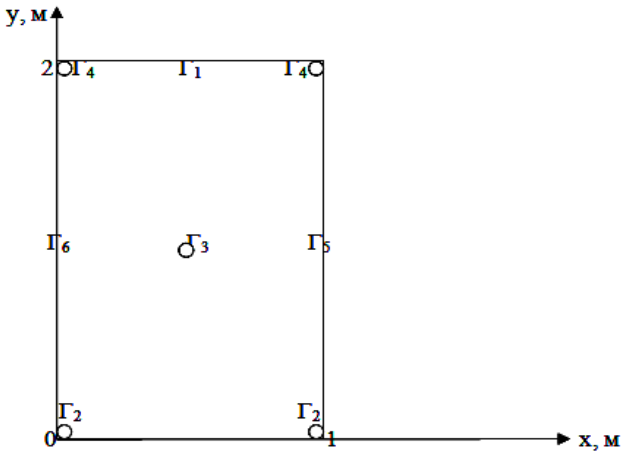


Рис. 1. Поперечний переріз області розв'язку задачі

З роботи [12] математична модель фільтрації з урахуванням колюматації містить наступні рівняння

$$\nabla \cdot (\rho_p(c)k_h(c, \sigma)\nabla h) = \sigma \frac{\partial \rho_p}{\partial c} \cdot \frac{\partial c}{\partial t} - \frac{\rho_p}{\rho_s} \frac{\partial s}{\partial t}, X \in \Omega, \quad (1)$$

$$\sigma \left(1 - \frac{c}{\rho_p} \frac{\partial \rho_p}{\partial c}\right) \frac{\partial c}{\partial t} = \nabla \cdot (D_c \nabla c) - u \left(1 - \frac{c}{\rho_p} \frac{\partial \rho_p}{\partial c}\right) \nabla c - \frac{\partial s}{\partial t}, X \in \Omega, \quad (2)$$

$$\frac{\partial s}{\partial t} = \alpha \cdot c - \beta \cdot s, X \in \Omega, \quad (3)$$

$$u = -k_h(c, s, \sigma) \nabla h, X \in \Omega. \quad (4)$$

Математична модель (1)–(4) дозволила підвищити адекватність результатів для досліджуваних фізичних процесів, однак не дозволяє знизити негативний вплив кольматації і показала необхідність розробки інженерних рішень для зниження впливу кольматаційно-суфозійних процесів на процеси фільтрації. Це вимагатиме, в свою чергу, урахування цих інженерних рішень в математичній моделі. Саме тому в даній роботі авторами запропоновано доповнити систему біоплато-фільтра додатковою системою перфорованих труб, яка розміщена в товщі фільтрувальної засипки. На додатковій системі перфорованих труб задаються умови викачування води, але з меншою інтенсивністю. Ця вода повторно подається на верхню систему перфорованих труб. Таким чином, фільтрувальна вода проходить очистку в два етапи.

Початкові умови:

$$c(x, y, 0) = C_0, (X) \in \overline{\Omega},$$

$$s(x, y, 0) = s_0, (X) \in \overline{\Omega}.$$

Граничні умови

$$h(X, t)|_{X \in \Gamma_1, \Gamma_4} = y.$$

На межах Γ_2 та Γ_3 потік залежить від відстані до правого краю:

$$q|_{X \in \Gamma_2} = \frac{q_{\min}(x - x_2)}{x_1 - x_2} + \frac{q_{\max}(x - x_1)}{x_2 - x_1},$$

$$q|_{X \in \Gamma_3} = w \times q|_{X \in \Gamma_2},$$

де w — коефіцієнт, q — потік рідини на одиницю довжини, $x_1 = 0$, $x_2 = 1$, $c(X, t)|_{X \in \Gamma_1, \Gamma_4} = C_{\max}$, Γ_5, Γ_6 — межі непроникувості.

Тут $\sigma(X, t)$ — пористість ґрунту, яка є змінною в часі за рахунок зміни концентрації кольматуючих частинок; $s(X, t)$ — масова концентрація $\left([s] = \frac{\text{кг}}{\text{м}^3} \right)$ кольматуючих частинок (маса частинок, які асоційовані зі скелетом ґрунту і віднесені до одиниці об'єму); c — концентрація суспензії, яка фільтрується $\left([c] = \frac{\text{кг}}{\text{м}^3} \right)$, маса завислих частинок в одиниці об'єму порової рідини); ρ_s — густина матеріалу кольматуючих частинок; $\rho_p = \rho_p(c)$ — густина порової рідини (су-

спензії), яка залежить від концентрації завислих частинок c ; $k_h = k_h(c, \sigma)$ — коефіцієнт фільтрації, який залежить від концентрації суспензії та пористості; h — напір в поровій рідині; D_c — коефіцієнт дисперсії частинок в поровій суспензії; $u = -k_h(c, s, \sigma)\nabla h$ — швидкість фільтрації порової суспензії; α — коефіцієнт швидкості прилипання частинок; β — коефіцієнт швидкості відриву частинок.

Для відшукування наближеного розв'язку поставленої крайової задачі використано метод скінченних елементів (МСЕ).

Слабке формулювання крайової задачі полягає в наступному. Домножимо рівняння (1) на пробну функцію

$$v1(X) \in H_0 = \{v1(X) : v1(X) \in W_2^1(\Omega)\}, v1(X)|_{\Gamma_1 \cup \Gamma_2} = 0,$$

інтегруємо по області Ω , застосовуємо формулу Остроградського-Гауса і отримаємо

$$\iint_{\Omega} (k_h \rho_p \nabla h \nabla v1) d\Omega + \iint_{\Omega} (\sigma \frac{\partial \rho_p}{\partial c} \cdot \frac{\partial c}{\partial t} \cdot v1) d\Omega - \iint_{\Omega} (\frac{\rho_p}{\rho_s} \cdot \frac{\partial s}{\partial t} \cdot v1) d\Omega = 0.$$

Домножимо рівняння (2) та початкову умову для концентрації суспензії на пробну функцію

$$v2(X) \in H_0 = \{v2(X) : v2(X) \in W_2^1(\Omega)\}, v2(X)|_{\Gamma_1} = 0,$$

інтегруємо по області Ω , застосовуємо формулу Остроградського-Гауса і отримаємо

$$\begin{aligned} \iint_{\Omega} (D_c \nabla c \nabla v2) d\Omega - \iint_{\Omega} (u(1 - \frac{c}{\rho_p} \frac{\partial \rho_p}{\partial c}) \cdot \nabla c \cdot v2) d\Omega + \iint_{\Omega} (\frac{\partial s}{\partial t} \cdot v2) d\Omega = \\ = \iint_{\Omega} (\sigma(1 - \frac{c}{\rho_p} \frac{\partial \rho_p}{\partial c}) \cdot \frac{c^i - c^{i-1}}{dt} \cdot v2) d\Omega, \\ \iint_{\Omega} c(x, y, 0) \cdot v2(X) d\Omega = \iint_{\Omega} c_0 \cdot v2(X) d\Omega, \end{aligned}$$

$$u = -k_h \nabla h,$$

$$s^i = (\alpha \cdot c^{i-1} - \beta \cdot s^{i-1}) \cdot dt + s^{i-1}.$$

Для відшукування наближеного узагальненого розв'язку отриманої задачі потрібно застосовувати дискретизацію в часі (для більш детальної інформації див. [12]).

Висновки. Побудовано вдосконалену математичну модель процесу фільтрації суспензії в біоплато-фільтрі, яка, на відміну від відомих аналогів, враховує нелінійну залежність коефіцієнта фільтрації від концентрації колюматуючих частинок та динамічну зміну пористості в процесі фільтрування. Також представлена математична мо-

дель враховує наявність додаткової системи перфорованих труб, яка розміщена в товщі фільтрувальної засипки. Урахування вказаних факторів дозволяє підвищити адекватність математичної моделі для досліджуваних фізичних процесів. Числові розв'язки відповідної нелінійної крайової задачі знайдено методом скінченних елементів.

Подальші напрямки досліджень авторів будуть стосуватися побудови досліджуваної області та проведення числових експериментів розв'язку отриманої крайової задачі і порівняння отриманих результатів з натурними експериментами. Для проведення числових експериментів планується використати безкоштовне програмне середовище FreeFem++.

Список використаних джерел:

1. Завацький С. В., Котельчук Л. С., Котельчук А. Л. Біоінженерні споруди для очищення стічних вод малої продуктивності. *Будівництво, інженерні системи та комунікації. Чернігівський науковий часопис*. 2012. Серія 2: Техніка і природа. № 1 (3). С. 57-63.
2. Филипчук В. Л., Бондар О. І., Курилюк М. С., Айяа Анісіфіюк, Кривошей П. П., Курилюк О. М., Почтар О. В. Очищення води у фільтраційно-регенераційних біоплато. *Вісник НУВГП*. 2016. Вип. 2 (74). С. 193-204.
3. Филипчук В. Л., Курилюк М. С., Филипчук Л. В. та ін. Очищення каламутних вод у фільтраційно-регенераційних біоплато. *Вісник інженерної академії України*. 2016. Вип. 3. С. 150-155.
4. Berres S., Bürger R., Wendland W. L. Mathematical Models for the Sedimentation of Suspensions. *Lecture Notes in Applied and Computational Mechanics*. 2006. Vol. 28. P. 7-44.
5. Bomba A., Safonik A. Mathematical Simulation of the Process of Aerobic Treatment of Wastewater under Conditions of Diffusion and Mass Transfer Perturbations. *Journal of Engineering Physics and Thermophysics*. 2018. № 91. P. 318-323.
6. Chetti A., Benamar A., Hazzab A. Modeling of Particle Migration in Porous Media: Application to Soil Suffusion. *Transport in Porous Media*. 2016. Vol. 113. № 3. P. 591-606.
7. Chrysikopoulos C. V., Katzourakis V. E. Colloid particle size-dependent dispersivity. *AGU Water Resources Research*. 2015. Vol. 51, Issue 6. P. 4668-4683.
8. Gleichman-Verheyc E. G., Putten W. H., Vander L. Alvalwaterzuvering met helofytenfilters, een haalbaarheidsstudie. *Tijdschr. watervoorz. en. afvalwater*. 1992. № 3. P. 56-60.
9. Healy A., Cawleyb M. Nutrient Processing Capacity of a Constructed Wetland in Western Ireland. *J. Environ. Quality*. 2002. № 31. P. 1739-1747.
10. Liu L., Fu Y., Wei Q., Liu Q., Wu L., Wu J., Huo W. Applying Bio-Slow Sand Filtration for Water Treatment. *Pol. J. Environ. Stud*. 2019. Vol. 28, No. 4. P. 2243-2251.
11. Liu J., Liu L., Huang Z., Fu Y., Huang Z. Contaminant Removal and Optimal Operation of Bio-Slow Sand Filtration Water Treatment Based on Nature-Based Solutions. *Pol. J. Environ. Stud*. 2020. Vol. 29. № 2. P. 1703-1713.
12. Moshynskyi V., Filipchuk V., Ivanchuk N., Martyniuk P. Computer modeling of water cleaning in wetland taking into account of suffusion and colmatation. *Eastern-European Journal of Enterprise Technologies*. 2018. № 1/10(91). P. 38-43.

13. Moshynsky V., Riabova O. Approaches to Aquatic Ecosystems Organic Energy Assessment and Modelling. *NAPSC*. 2013. P. 125-135.
14. Safonyk A., Bomba A., Tarhonii I. Modeling and Automation of the Electrocoagulation Process in Water Treatment. *Advances in Intelligent Systems and Computing*. 2018. Vol. 871. P. 451-463.
15. Seetha N., Mohan Kumar M. S., Hassanizadeh S. M. Modeling the co-transport of viruses and colloids in unsaturated porous media. *Journal of Contaminant Hydrology*. 2015. Vol. 181. P. 82-101.
16. Zhang M., He F., Zhao D., Hao X. Transport of stabilized iron nanoparticles in porous media: Effects of surface and solution chemistry and role of adsorption. *Journal of Hazardous Materials*. 2017. Vol. 322, Part A. P. 284-291.

MATHEMATICAL MODELING OF WATER PURIFICATION IN BIOPILATO-FILTER

Biopilato — a facility for the treatment of domestic, industrial wastewater, contaminated surface runoff. In the known structures of the biopilato there is a gradual colmatation of the pore space of the filter backfill and the bottom drainage with biofilm and mineralized sediment, the accumulation of silt in the lower part of the structure. To restore the operation of the biopilato, it is necessary to periodically stop them for complex and long-term repair and restoration works related to flushing and regeneration of the filter backfill and drainage. Therefore, it is important to forecast the operation of the biopilato at the design stage in order to make predictive calculations. It is better to do it by means of mathematical and computer modeling without conducting field experiments. In this paper, the authors constructed a mathematical model of filtration taking into account colmatation and suffusion in the biopilato-filter system in the two-dimensional case. The constructed mathematical model takes into account the physical effects of the dynamic change of porosity and the dependence of the filtration coefficient on the concentration of contaminants, which is not in the known analogues. The nonlinearity of the constructed model does not allow us to speak about the analytical solutions of the corresponding boundary value problem. Moreover, if we consider the three-dimensional problem in the future. Mathematical modeling allows to estimate the effect of the considered factors, but does not allow to reduce such negative influence. Therefore, to preserve the productivity of the biopilato, the authors proposed engineering solutions that reduce the impact of colmatation-suffusion processes on the efficiency of the biopilato-filter. Namely: in the mathematical model, the authors take into account the presence of an additional system of perforated pipes, which is located in the thickness of the filter backfill. Taking into account these factors allows to increase the adequacy of the mathematical model for the studied physical processes. Numerical solutions of the problem are found by the finite element method. Numerical experiments are planned to do in the free software environment FreeFem++.

Key words: *biopilato-filter, filtration problem, colmatation, suffusion, finite element method, perforated pipe system.*

Отримано: 13.05.2020

УДК 517.95

DOI: 10.32626/2308-5916.2020-21.83-101

Ю. Є. Климяк, канд. техн. наук,

А. Я. Бомба, д-р техн. наук, професор

Національний університет водного господарства

і природокористування, м. Рівне

КОМП'ЮТЕРНЕ ПРОГНОЗУВАННЯ ПРОЦЕСІВ «ФІЛЬТРУВАННЯ-РЕГЕНЕРАЦІЯ» ШВИДКИХ КОНУСОПОДІБНИХ АДСОРБЦІЙНИХ ФІЛЬТРІВ З КУСКОВО-ОДНОРІДНИМИ ПОРИСТИМИ ЗАВАНТАЖЕННЯМИ

Для прогнозування процесів «фільтрування-регенерація» швидких конусоподібних адсорбційних фільтрів з кусково-однорідними пористими завантаженнями сформовано математичні моделі їх основних технологічних режимів: фільтрування, зворотньої промивки, хімічної регенерації та прямої промивки з урахуванням впливу температурних ефектів на внутрішню кінетику масопереносу при дотриманні сталих швидкостей відповідних режимів. Враховуючи, що у режимі фільтрування конвективні складові тепло- і масоперенесення та фізична і хімічна адсорбція переважають над вкладом дифузії й фізичної десорбції, у режимі хімічної регенерації конвективні складові тепло- і масоперенесення та фізична і хімічна десорбція переважають над вкладом дифузії й фізичної адсорбції, а у режимах зворотньої та прямої промивок конвективні складові тепло- і масоперенесення та фізична десорбція і хімічна адсорбція переважають над вкладом дифузії й фізичної адсорбції, отримано алгоритми числово-асимптотичних наближень розв'язків відповідних нелінійних сингулярно-збурених крайових задач для модельної конусоподібної області, обмеженої заданими, гладкими, ортогональними між собою по ребрах, двома еквіпотенціальними поверхнями і однією поверхнею течії та розділеної на декілька підобластей деякими заданими еквіпотенціальними поверхнями. Запропоновані моделі у комплексі за рахунок урахування впливу зміни температури та швидкості фільтраційного потоку вздовж висоти фільтру на коефіцієнти, які характеризують швидкості масообміну при фізичній і хімічній адсорбції та десорбції, коефіцієнт фільтрації, дають змогу шляхом проведення комп'ютерних експериментів більш якісно дослідити зміну концентрацій домішок у фільтраційному потоці і на поверхні адсорбенту завантаження, утримуваних за рахунок фізичної і хімічної адсорбції, температури фільтраційного потоку, коефіцієнта фільтрації та активної пористості у кожному шарі завантаження вздовж висоти фільтру та на їх основі спрогнозувати більш оптимальні варіанти ви-

користання адсорбентів кожного шару завантаження і часові інтервали роботи фільтрів відповідно у режимах фільтрування, зворотньої промивки, хімічної регенерації та прямої промивки при дотриманні сталих швидкостей відповідних режимів.

Ключові слова: *математична модель, прогнозування, фільтрування, промивка, хімічна регенерація, домішка, адсорбція, десорбція, температура, швидкий конусоподібний фільтр, кусково-однорідне пористе завантаження.*

Вступ. Для водопідготовки (очищення) великих обсягів води дедалі ширше використовуються швидкі адсорбційні фільтри з одно- і багат шаровими пористими регенеративними завантаженнями при дотриманні сталої швидкості фільтрування 5-10 м/год. За допомогою регенерації (з використанням спеціальних хімічних речовин) адсорбційні властивості фільтрувальних матеріалів завантажень вдається майже повністю відновити, причому, як правило, в автоматичному режимі, з мінімальною участю користувача [1-5]. Як адсорбенти у них використовують природні (бентоніт, монтморилоніт, торф), штучні (активоване вугілля, штучні цеоліти, полісорби) та синтетичні матеріали (наноструктуровані вуглецеві сорбенти) [6-8].

Важливою особливістю процесу фільтрування є збільшення гідравлічного опору завантаження фільтра через накопичення частинок домішок на поверхні адсорбенту завантаження за рахунок фізичної і хімічної адсорбції, тому підтримка постійної швидкості фільтрування досягається автоматичним регулюванням збільшення відкриття засувки на трубопроводі фільтрату в міру збільшення опору. Імпульсом для збільшення відкриття засувки на трубопроводі фільтрату служить зміна рівня води на фільтрі (контролюється поплавковим пристроєм) або витрати води в трубопроводі фільтрату (контролюється за допомогою дроселюючого пристрою і дифманометра) [9]. Коли засувка відкрита повністю, фільтр вимикається з роботи для регенерації завантаження. Десорбція домішок з поверхні адсорбенту завантаження є необхідною стадією технологічного процесу регенерації швидких фільтрів. Спочатку включається режим зворотньої промивки з високою швидкістю подачі води (в 2-3 рази вища за швидкість режиму фільтрування), який триває 5-20 хв. і дозволяє розпушити фільтруючий матеріал кожного шару завантаження та видалити частинки домішок з поверхні адсорбенту завантаження, накопичені там за рахунок фізичної адсорбції. Далі здійснюється режим хімічної регенерації з високою швидкістю подачі розчину певного реагенту (зазвичай використовується перманганат калію KMnO_4), що запускає процес хімічного відновлення адсорбційної здатності фільтрувального матеріалу кожного шару завантаження і триває 10-30 хв. У розчин реагенту переходять частинки домішки з

поверхні адсорбенту завантаження, накопичені там за рахунок хімічної адсорбції. Останнім здійснюється режим прямої промивки з високою швидкістю подачі води, що триває до 10 хв. і дозволяє ущільнити фільтруючий матеріал кожного шару завантаження та видалити залишки домішок і хімічного розчину реагенту.

Постійно зростаючі потреби у господарстві в очищеній воді та зростання вартості фільтрувальних матеріалів вимагають проведення досліджень з одного боку їх більш оптимального використання і збільшення тривалості роботи швидких адсорбційних фільтрів у режимі фільтрування за рахунок вибору їх форми, висоти шарів, зокрема, з урахуванням впливу температурних ефектів на внутрішню кінетику масопереносу, а з іншого — регенерації пористих завантажень для їх повторного використання.

Аналіз останніх досліджень. Фільтрування води через пористі завантаження швидких адсорбційних фільтрів є окремим випадком руху рідин через пористі матеріали і закономірності цього руху досить повно висвітлені в працях Д. М. Мінца, Л. С. Лейбензона, Н. Н. Павловського, С. А. Шуберта та інших дослідників [10-12]. Установлено, що при фільтруванні води через пористі матеріали справедливим є лінійний закон Дарсі, тобто має місце ламінарний режим руху води. Швидкість процесу адсорбції залежить від концентрації, природи і структури домішок, швидкості фільтрування, температури фільтраційного потоку, виду і властивостей адсорбенту [13]. В якості математичних моделей для прогнозування процесів «фільтрування-регенерація» пористих завантажень вітчизняними дослідниками найчастіше використовується модель Д. М. Мінца [10, 12] при дотриманні сталих швидкостей відповідних процесів та температури або деяка її модифікація (вдосконалена модель). У [14] запропоновано її просторове узагальнення для прогнозування процесу адсорбційної доочистки води від домішок у швидких фільтрах конусоподібної форми при додержанні сталої швидкості фільтрування, яка є більш ефективною для проведення теоретичних досліджень, націлених на оптимізацію параметрів процесу фільтрування (тривалості роботи, форми, розмірів фільтра, висоти шарів тощо) за рахунок введення додаткових рівнянь для визначення зміни коефіцієнта фільтрації та активної пористості завантаження вздовж висоти фільтру, врахування дифузії у фільтраційному потоці і на поверхні адсорбенту завантаження. Актуальною задачею є узагальнення відповідної моделі для прогнозування технологічних режимів фільтрування, зворотної промивки, хімічної регенерації та прямої промивки швидких конусоподібних адсорбційних фільтрів з кусково-однорідними пористими завантаженнями з урахуванням впливу температурних ефектів

на внутрішню кінетику масопереносу. Ці моделі у комплексі за рахунок урахування впливу зміни температури та швидкості фільтраційного потоку вздовж висоти фільтру на коефіцієнти, які характеризують швидкості масообміну при фізичній і хімічній адсорбції та десорбції, коефіцієнт фільтрації, дадуть змогу шляхом проведення комп'ютерних експериментів спрогнозувати більш оптимальні варіанти використання адсорбентів кожного шару завантаження і часові інтервали роботи фільтрів відповідно у режимах фільтрування, зворотньої промивки, хімічної регенерації та прямої промивки при дотриманні сталих швидкостей відповідних режимів.

Постановка математичних моделей. Сформуємо модельні задачі прогнозування процесів «фільтрування-регенерація» швидкого конусоподібного адсорбційного фільтру з кусково-однорідними пористими завантаженнями для області $G = G_z \times (0, \infty)$, G_z — просторова однозв'язна область ($z = (x, y, z)$), обмежена заданими гладкими, ортогональними між собою по ребрах, двома еквіпотенціальними поверхнями S_* , S^* і однією поверхнею течії S^{**} та розділена на p підобластей G_z^r ($r = \overline{1, p}$) деякими заданими $p - 1$ еквіпотенціальними поверхнями S_{*r}^* ($r = \overline{1, p - 1}$) (рис. 1).

Вважаючи, що у режимі фільтрування конвективні складові тепло- і масоперенесення та фізична і хімічна адсорбція переважають над вкладом дифузії й фізичної десорбції, у режимі хімічної регенерації конвективні складові тепло- і масоперенесення та фізична і хімічна десорбція переважають над вкладом дифузії й фізичної адсорбції, а у режимах зворотньої та прямої промивок конвективні складові тепло- і масоперенесення та фізична десорбція і хімічна адсорбція переважають над вкладом дифузії й фізичної адсорбції, просторові модельні задачі для прогнозування технологічних режимів швидкого конусоподібного адсорбційного фільтру при дотриманні сталих швидкостей відповідних режимів з урахуванням зворотного впливу характеристик процесу (концентрацій домішок відповідно у фільтраційному потоці і на поверхні адсорбенту завантаження, температури фільтраційного потоку) на характеристики завантаження (коефіцієнти фільтрації, активної пористості, коефіцієнти, які характеризують швидкості масообміну при фізичній і хімічній адсорбції та десорбції домішок) складатимуться з рівнянь руху фільтраційного потоку згідно із законом Дарсі та рівняння нерозривності:

$$\{\vec{v} = \kappa_0^0 \cdot \text{grad } \varphi, \text{div } \vec{v} = 0, (x, y, z) \in G_z^r, r = \overline{1, p}, \quad (1)$$

рівнянь для визначення зміни концентрацій домішок у фільтраційному потоці і на поверхні адсорбенту завантаження, утримуваних за

рахунок фізичної та хімічної адсорбції, температури фільтраційного потоку, коефіцієнта фільтрації та активної пористості вздовж висоти фільтру відповідно для режимів фільтрування:

$$\left\{ \begin{array}{l} (\sigma \cdot C)'_t = \operatorname{div} (D_1 \cdot \operatorname{grad} C) - \vec{v} \cdot \operatorname{grad} C - \alpha \cdot C + \beta \cdot U - \alpha^* \cdot C, \\ (\sigma \cdot U)'_t = \operatorname{div} (D_2 \cdot \operatorname{grad} U) + \alpha \cdot C - \beta \cdot U, \\ (\sigma \cdot W)'_t = \operatorname{div} (D_3 \cdot \operatorname{grad} W) + \alpha^* \cdot C, \\ (\sigma \cdot T)'_t = \operatorname{div} (D^* \cdot \operatorname{grad} T) - \vec{v} \cdot \operatorname{grad} T + \gamma \cdot (\alpha \cdot C - \beta \cdot U) + \gamma^* \cdot \alpha^* \cdot C, \\ \kappa'_t = -\mu \cdot U - \mu^* \cdot W, \\ \sigma'_t = -\lambda \cdot U - \lambda^* \cdot W, (x, y, z) \in G_z^r, r = \overline{1, p}, \end{array} \right. \quad (2)$$

хімічної регенерації:

$$\left\{ \begin{array}{l} (\sigma \cdot C)'_t = \operatorname{div} (D_1 \cdot \operatorname{grad} C) - \vec{v} \cdot \operatorname{grad} C + \beta \cdot U - \alpha \cdot C + \beta^* \cdot W, \\ (\sigma \cdot U)'_t = \operatorname{div} (D_2 \cdot \operatorname{grad} U) - \beta \cdot U + \alpha \cdot C, \\ (\sigma \cdot W)'_t = \operatorname{div} (D_3 \cdot \operatorname{grad} W) - \beta^* \cdot W, \\ (\sigma \cdot T)'_t = \operatorname{div} (D^* \cdot \operatorname{grad} T) - \vec{v} \cdot \operatorname{grad} T + \gamma \cdot (\beta \cdot U - \alpha \cdot C) + \gamma^* \cdot \beta^* \cdot W, \\ \kappa'_t = \mu \cdot U + \mu^* \cdot W, \\ \sigma'_t = \lambda \cdot U + \lambda^* \cdot W, (x, y, z) \in G_z^r, r = \overline{1, p}, \end{array} \right. \quad (3)$$

зворотньої та прямої промивок:

$$\left\{ \begin{array}{l} (\sigma \cdot C)'_t = \operatorname{div} (D_1 \cdot \operatorname{grad} C) - \vec{v} \cdot \operatorname{grad} C + \beta \cdot U - \alpha \cdot C - \alpha^* \cdot C, \\ (\sigma \cdot U)'_t = \operatorname{div} (D_2 \cdot \operatorname{grad} U) - \beta \cdot U + \alpha \cdot C, \\ (\sigma \cdot W)'_t = \operatorname{div} (D_3 \cdot \operatorname{grad} W) + \alpha^* \cdot C, \\ (\sigma \cdot T)'_t = \operatorname{div} (D^* \cdot \operatorname{grad} T) - \vec{v} \cdot \operatorname{grad} T + \gamma \cdot (\beta \cdot U - \alpha \cdot C) + \gamma^* \cdot \alpha^* \cdot C, \\ \kappa'_t = \mu \cdot U - \mu^* \cdot W, \\ \sigma'_t = \lambda \cdot U - \lambda^* \cdot W, (x, y, z) \in G_z^r, r = \overline{1, p}, \end{array} \right. \quad (4)$$

які доповнюються наступними крайовими умовами:

$$\left\{ \begin{array}{l} \varphi|_{S_*} = \varphi_*, \varphi|_{S^*} = \varphi^*, \varphi'_n|_{S^{**}} = 0, \end{array} \right. \quad (5)$$

$$\left\{ \begin{array}{l} C|_{S_*} = c_*, C'_n|_{S^*} = 0, C'_n|_{S^{**}} = 0, \\ U|_{S_*} = u_*, U'_n|_{S^*} = 0, U'_n|_{S^{**}} = 0, \\ W|_{S_*} = u_*, W'_n|_{S^*} = 0, W'_n|_{S^{**}} = 0, \\ T|_{S_*} = T_*, T'_n|_{S^*} = 0, T'_n|_{S^{**}} = 0, \end{array} \right. \quad (6)$$

початковими умовами:

$$\begin{cases} C|_{t=0} = c_0^0, U|_{t=0} = u_0^0, W|_{t=0} = w_0^0, \\ T|_{t=0} = T_0^0, \kappa|_{t=0} = \kappa_0^0, \sigma|_{t=0} = \sigma_0^0 \end{cases} \quad (7)$$

та умовами узгодженості на поверхнях розділу S_{*r}^* ($r = \overline{1, p-1}$):

$$\begin{cases} \varphi|_{S_{r-}^*} = \varphi|_{S_{r+}^*} = \varphi_{*r}^*, \kappa_r^0 \cdot \varphi_{\vec{n}}'|_{S_{r-}^*} = \kappa_r^0 \cdot \varphi_{\vec{n}}'|_{S_{r+}^*}, \quad r = \overline{1, p-1}, \\ \left\{ \begin{array}{l} C|_{S_{r-}^*} = C|_{S_{r+}^*}, U|_{S_{r-}^*} = U|_{S_{r+}^*}, W|_{S_{r-}^*} = W|_{S_{r+}^*}, T|_{S_{r-}^*} = T|_{S_{r+}^*}, \\ D_{1,r} \cdot C_{\vec{n}}' - v_{rn}^0 \cdot C|_{S_{r-}^*} = D_{1,r+1} \cdot C_{\vec{n}}' - v_{rn}^0 \cdot C|_{S_{r+}^*}, \\ D_{2,r} \cdot U_{\vec{n}}'|_{S_{r-}^*} = D_{2,r+1} \cdot U_{\vec{n}}'|_{S_{r+}^*}, \\ D_{3,r} \cdot W_{\vec{n}}'|_{S_{r-}^*} = D_{3,r+1} \cdot W_{\vec{n}}'|_{S_{r+}^*}, \\ D_r^* \cdot T_{\vec{n}}' - v_{rn}^0 \cdot T|_{S_{r-}^*} = D_{r+1}^* \cdot T_{\vec{n}}' - v_{rn}^0 \cdot T|_{S_{r+}^*}, \\ D_{1,r} \cdot C_{\vec{n}}' - v_{rn} \cdot C + D_{2,r} \cdot U' + D_{3,r} \cdot W_{\vec{n}}'|_{S_{r-}^*} = \\ = D_{1,r+1} \cdot C_{\vec{n}}' - v_{rn} \cdot C + D_{2,r+1} \cdot U' + D_{3,r+1} \cdot W_{\vec{n}}'|_{S_{r+}^*}, \\ (\sigma \cdot (C + U + W))_t'|_{S_{r-}^*} = (\sigma \cdot (C + U + W))_t'|_{S_{r+}^*}, \quad r = \overline{1, p-1}, \end{array} \right. \end{cases} \quad (9)$$

де $\varphi = \varphi(x, y, z)$ і $\vec{v} = \vec{v}(v_x, v_y, v_z)$ — відповідно потенціал і вектор швидкості фільтрації, $0 \leq \varphi_* = \varphi_{*0}^* < \varphi_{*1}^* < \dots < \varphi_{*p-1}^* < \varphi_{*p}^* = \varphi^* < \infty$, φ_* і φ^* — задані дійсні числа, $\varphi_{*1}^*, \varphi_{*2}^*, \dots, \varphi_{*(p-1)}^*$ — невідомі значення потенціалу, $v = |\vec{v}| = \sqrt{v_x^2(x, y, z) + v_y^2(x, y, z) + v_z^2(x, y, z)} \gg 0$, κ_*^* — початковий коефіцієнт фільтрації, $\kappa_*^* = \{\kappa_r^*, (x, y, z) \in G_z^r, r = \overline{1, p}\}$, $\kappa_r^* > 0$ ($r = \overline{1, p}$), \vec{n} — зовнішня нормаль до відповідної поверхні, $C = C(x, y, z, t)$ — концентрація домішок у фільтраційному потоці, $U = U(x, y, z, t)$ і $W = W(x, y, z, t)$ — концентрації домішок на поверхні адсорбенту завантаження, утримуваних відповідно за рахунок фізичної та хімічної адсорбції, $T = T(x, y, z, t)$ — температура фільтраційного потоку, $\kappa = \kappa(x, y, z, t)$ — коефіцієнт фільтрації, $\sigma = \sigma(x, y, z, t)$ — активна пористість в точці (x, y, z) у момент часу t , D_1 — коефіцієнт дифузії домішок у фільтраційному потоці, D_2 і

D_3 — коефіцієнти дифузії домішок на поверхні адсорбенту завантаження, утримуваних відповідно за рахунок фізичної та хімічної адсорбції, $D_s = \{D_{s,r}, (x, y, z) \in G_z^r, r = \overline{1, p}\}$, $D_{s,r} = \varepsilon \cdot d_{s,r,0}$, $d_{s,r,0} > 0$ ($s = \overline{1, 3}$, $r = \overline{1, p}$), D^* — коефіцієнт температуропровідності, $D^* = \{D_r^*, (x, y, z) \in G_z^r, r = \overline{1, p}\}$, $D_r^* = \varepsilon \cdot d_{r,0}^*$, $d_{r,0}^* > 0$ ($r = \overline{1, p}$), α і β — коефіцієнти, які характеризують швидкість масообміну відповідно при фізичній адсорбції та десорбції домішок, для модельної задачі прогнозування режиму фільтрування

$$\alpha = \{\alpha_r, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \alpha_r = \sum_{s_1=0}^2 \sum_{s_2=0}^{2-s_1} \varepsilon^{s_1+s_2} \cdot \alpha_{r,s_1,s_2} \cdot \nu^{s_1} \cdot T^{s_2},$$

$$\alpha_{r,s_1,s_2} \in \mathbb{R} \quad (r = \overline{1, p}, s_1, s_2 = \overline{0, 2}), \beta = \{\beta_r, (x, y, z) \in G_z^r, r = \overline{1, p}\},$$

$$\beta_r = \varepsilon \cdot \sum_{s_1=0}^2 \sum_{s_2=0}^{2-s_1} \varepsilon^{s_1+s_2} \cdot \beta_{r,s_1,s_2} \cdot \nu^{s_1} \cdot T^{s_2}, \beta_{r,s_1,s_2} \in \mathbb{R} \quad (r = \overline{1, p}, s_1, s_2 = \overline{0, 2}),$$

а модельних задач прогнозування режимів хімічної регенерації, зворотньої та прямої промивок

$$\alpha = \{\alpha_r, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \alpha_r = \varepsilon \cdot \sum_{s_1=0}^2 \sum_{s_2=0}^{2-s_1} \varepsilon^{s_1+s_2} \cdot \alpha_{r,s_1,s_2} \cdot \nu^{s_1} \cdot T^{s_2},$$

$$\alpha_{r,s_1,s_2} \in \mathbb{R} \quad (r = \overline{1, p}, s_1, s_2 = \overline{0, 2}), \beta = \{\beta_r, (x, y, z) \in G_z^r, r = \overline{1, p}\},$$

$$\beta_r = \sum_{s_1=0}^2 \sum_{s_2=0}^{2-s_1} \varepsilon^{s_1+s_2} \cdot \beta_{r,s_1,s_2} \cdot \nu^{s_1} \cdot T^{s_2}, \beta_{r,s_1,s_2} \in \mathbb{R} \quad (r = \overline{1, p}, s_1, s_2 = \overline{0, 2}),$$

α^* — коефіцієнт, який характеризує швидкість масообміну при хімічній адсорбції домішок, для модельних задач прогнозування режимів фільтрування, зворотньої та прямої промивок $\alpha^* = \{\alpha_r^*, (x, y, z) \in G_z^r,$

$$r = \overline{1, p}\}, \alpha_r^* = \sum_{s_1=0}^2 \sum_{s_2=0}^{2-s_1} \varepsilon^{s_1+s_2} \cdot \alpha_{r,s_1,s_2}^* \cdot \nu^{s_1} \cdot T^{s_2}, \alpha_{r,s_1,s_2}^* \in \mathbb{R} \quad (r = \overline{1, p}, s_1,$$

$s_2 = \overline{0, 2}), \beta^*$ — коефіцієнт, який характеризує швидкість масообміну при хімічній десорбції домішок, для модельної задачі прогнозування режиму хімічної регенерації, $\beta^* = \{\beta_r^*, (x, y, z) \in G_z^r, r = \overline{1, p}\},$

$$\beta_r^* = \sum_{s_1=0}^2 \sum_{s_2=0}^{2-s_1} \varepsilon^{s_1+s_2} \cdot \beta_{r,s_1,s_2}^* \cdot \nu^{s_1} \cdot T^{s_2}, \beta_{r,s_1,s_2}^* \in \mathbb{R} \quad (r = \overline{1, p}, s_1, s_2 = \overline{0, 2}),$$

γ — коефіцієнт, що характеризує швидкість зміни температури фі-

льтраційного потоку при фізичній адсорбції та десорбції домішок, $\gamma = \{\gamma_r, (x, y, z) \in G_z^r, r = \overline{1, p}\}$, $\gamma_r \in \mathbb{R}$ ($r = \overline{1, p}$), γ^* — коефіцієнт, що характеризує швидкість зміни температури фільтраційного потоку при хімічній адсорбції та десорбції домішок, $\gamma^* = \{\gamma_r^*, (x, y, z) \in G_z^r, r = \overline{1, p}\}$, $\gamma_r^* \in \mathbb{R}$ ($r = \overline{1, p}$), μ і λ — коефіцієнти, що характеризують швидкість зміни відповідно коефіцієнта фільтрації й активної пористості завантаження при фізичній адсорбції та десорбції домішок, $\mu = \{\mu_r, (x, y, z) \in G_z^r, r = \overline{1, p}\}$, $\mu_r = \varepsilon \cdot \sum_{s=0}^2 \varepsilon^s \cdot \mu_{r,s} \cdot T^s$, $\mu_{r,s} \in \mathbb{R}$ ($r = \overline{1, p}$, $s = \overline{0, 2}$), $\lambda = \{\lambda_r, (x, y, z) \in G_z^r, r = \overline{1, p}\}$, $\lambda_r = \varepsilon \cdot \bar{\lambda}_r$ ($r = \overline{1, p}$), μ^* і λ^* — коефіцієнти, що характеризують швидкість зміни відповідно коефіцієнта фільтрації й активної пористості завантаження при хімічній адсорбції і десорбції домішок,

$$\begin{aligned}
 \mu^* &= \{\mu_r^*, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \mu_r^* = \varepsilon \cdot \sum_{s=0}^2 \varepsilon^s \cdot \mu_{r,s}^* \cdot T^s, \mu_{r,s}^* \in \mathbb{R} \\
 (r = \overline{1, p}, s = \overline{0, 2}), \lambda^* &= \{\lambda_r^*, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \lambda_r^* = \varepsilon \cdot \bar{\lambda}_r^* \\
 (r = \overline{1, p}), \alpha_r &= \alpha_r(x, y, z, t), \beta_r = \beta_r(x, y, z, t), \alpha_r^* = \alpha_r^*(x, y, z, t), \\
 \beta_r^* &= \beta_r^*(x, y, z, t), \mu_r = \mu_r(x, y, z, t), \bar{\lambda}_r = \bar{\lambda}_r(x, y, z, t), \\
 \mu_r^* &= \mu_r^*(x, y, z, t), \bar{\lambda}_r^* = \bar{\lambda}_r^*(x, y, z, t)
 \end{aligned}$$

— неперервні обмежені функції ($r = \overline{1, p}$), ε — малий параметр ($\varepsilon > 0$),

$$\begin{aligned}
 c_0^0 &= \{c_r^0, (x, y, z) \in G_z^r, r = \overline{1, p}\}, u_0^0 = \{u_r^0, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \\
 w_0^0 &= \{w_r^0, (x, y, z) \in G_z^r, r = \overline{1, p}\}, T_0^0 = \{T_r^0, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \\
 \kappa_0^0 &= \{\kappa_r^0, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \sigma_0^0 = \{\sigma_r^0, (x, y, z) \in G_z^r, r = \overline{1, p}\}, \\
 c_*^* &= c_*^*(M, t), c_r^0 = c_r^0(x, y, z), u_*^* = u_*^*(M, t), u_r^0 = u_r^0(x, y, z), \\
 w_*^* &= w_*^*(M, t), w_r^0 = w_r^0(x, y, z), T_*^* = T_*^*(M, t), T_r^0 = T_r^0(x, y, z), \\
 \kappa_r^0 &= \kappa_r^0(x, y, z), \sigma_r^0 = \sigma_r^0(x, y, z) \quad (r = \overline{1, p})
 \end{aligned}$$

— достатньо гладкі функції, узгоджені між собою на ребрах області G [15], $M \in S_*$, v_{rn}^0 і v_{rn} ($r = \overline{1, p-1}$) — відповідно початкові і поточні нормальні складові швидкості на поверхнях розділу S_{*r}^* ($r = \overline{1, p-1}$).

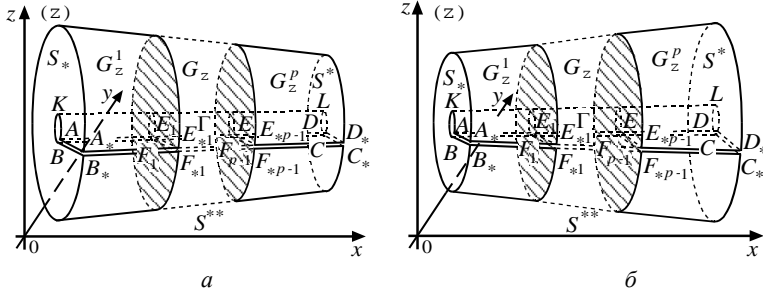


Рис. 1. Просторові області фільтрації G_z з умовним розрізом Γ для модельних задач прогнозування технологічних режимів: а — фільтрування та прямої промивки; б — зворотної промивки та хімічної регенерації

Аналогічно [14, 15] шляхом виконання умовного розрізу $\Gamma = \Gamma_1 \cup \Gamma_2$, де $\Gamma_1 = AKLDBKLC$ і $\Gamma_2 = ADD_*A_*BCC_*B_*$ розрізи вздовж відповідних поверхонь течії, модельні задачі для прогнозування режимів фільтрування (1), (2) (5)-(9), хімічної регенерації (1), (3), (5)-(9), зворотної та прямої промивок (1), (4)-(9) швидких конусоподібних фільтрів з кусково-однорідними пористими завантаженнями зводяться до розв'язання в однозв'язній кусково-однорідній області $G_z \setminus \Gamma$ — криволінійному паралелепіпеді $ABCD_*B_*C_*D_*$, обмеженому гладкими, ортогональними між собою в кутових точках та вздовж ребер, двома еквіпотенціальними поверхнями ABB_*A_* , CDD_*C_* і чотирма поверхнями течії $ABCD = AKLD \cup BKLC$, $A_*B_*C_*D_*$, $ADD_*A_* = BCC_*B_*$ та розділеному на p підобластей $G_z^1 \setminus \Gamma^1 = ABF_1E_1A_*B_*F_{*1}E_{*1}$, ..., $G_z^p \setminus \Gamma^p = E_{p-1}F_{p-1}CDE_{*(p-1)}F_{*(p-1)}C_*D_*$ ($\Gamma^1 = AE_1E_{*1}A_*BF_1F_{*1}B_*$, ..., $\Gamma^p = E_{p-1}DD_*E_{*(p-1)}F_{p-1}CC_*F_{*(p-1)}$) еквіпотенціальними поверхнями $E_rF_rF_{*r}E_{*r}$ ($r = 1, p-1$) (рис. 2), з додаванням умови непроникності $\varphi'_n|_{\Gamma} = 0$ вздовж умовного розрізу Γ задач, що описуються системою рівнянь (1) та відповідно системами рівнянь (2), (3) і (4) з крайовими умовами:

$$\left\{ \begin{aligned} \varphi \Big|_{ABB_*A_*} = \varphi_*, \varphi \Big|_{CDD_*C_*} = \varphi^*, \varphi'_n \Big|_{ABCD \cup A_*B_*C_*D_* \cup ADD_*A_* \cup BCC_*B_*} = 0, \\ \left\{ \begin{aligned} C \Big|_{ABB_*A_*} = C_*^*, C'_n \Big|_{CDD_*C_*} = 0, C'_n \Big|_{ADD_*A_* \cup BCC_*B_* \cup ABCD \cup A_*B_*C_*D_*} = 0, \\ U \Big|_{ABB_*A_*} = u_*^*, U'_n \Big|_{CDD_*C_*} = 0, U'_n \Big|_{ADD_*A_* \cup BCC_*B_* \cup ABCD \cup A_*B_*C_*D_*} = 0, \\ W \Big|_{ABB_*A_*} = w_*^*, W'_n \Big|_{CDD_*C_*} = 0, W'_n \Big|_{ADD_*A_* \cup BCC_*B_* \cup ABCD \cup A_*B_*C_*D_*} = 0, \\ T \Big|_{ABB_*A_*} = T_*^*, T'_n \Big|_{CDD_*C_*} = 0, T'_n \Big|_{ADD_*A_* \cup BCC_*B_* \cup ABCD \cup A_*B_*C_*D_*} = 0, \end{aligned} \right. \end{aligned} \right. \quad (10)$$

початковими умовами (7), умовами узгодженості на еквіпотенціальних поверхнях розділу $E_r F_r F_{*r} E_{*r}$ ($r = 1, p-1$):

$$\begin{cases} \varphi \Big|_{E_r F_r F_{*r} E_{*r}} = \varphi \Big|_{E_r F_r F_{*r} E_{*r}} = \varphi_{*r}^*, \\ \kappa_r^0 \cdot \varphi'_n \Big|_{E_r F_r F_{*r} E_{*r}} = \kappa_{r+1}^0 \cdot \varphi'_n \Big|_{E_r F_r F_{*r} E_{*r}}, \quad r = 1, p-1, \end{cases} \quad (12)$$

$$\begin{cases} C \Big|_{E_r F_r F_{*r} E_{*r}} = C \Big|_{E_r F_r F_{*r} E_{*r}}, U \Big|_{E_r F_r F_{*r} E_{*r}} = U \Big|_{E_r F_r F_{*r} E_{*r}}, \\ W \Big|_{E_r F_r F_{*r} E_{*r}} = W \Big|_{E_r F_r F_{*r} E_{*r}}, T \Big|_{E_r F_r F_{*r} E_{*r}} = T \Big|_{E_r F_r F_{*r} E_{*r}}, \\ D_{1,r} \cdot C'_n - v_{rn}^0 \cdot C \Big|_{E_r F_r F_{*r} E_{*r}} = D_{1,r+1} \cdot C'_n - v_{rn}^0 \cdot C \Big|_{E_r F_r F_{*r} E_{*r}}, \\ D_{2,r} \cdot U'_n \Big|_{E_r F_r F_{*r} E_{*r}} = D_{2,r+1} \cdot U'_n \Big|_{E_r F_r F_{*r} E_{*r}}, \\ D_{3,r} \cdot W'_n \Big|_{E_r F_r F_{*r} E_{*r}} = D_{3,r+1} \cdot W'_n \Big|_{E_r F_r F_{*r} E_{*r}}, \\ D_r^* \cdot T'_n - v_{rn}^0 \cdot T \Big|_{E_r F_r F_{*r} E_{*r}} = D_{r+1}^* \cdot T'_n - v_{rn}^0 \cdot T \Big|_{E_r F_r F_{*r} E_{*r}}, \\ D_{1,r} \cdot C'_n - v_{rn} \cdot C + D_{2,r} \cdot U'_n + D_{3,r} \cdot W'_n \Big|_{E_r F_r F_{*r} E_{*r}} = \\ = D_{1,r+1} \cdot C'_n - v_{rn} \cdot C + D_{2,r+1} \cdot U'_n + D_{3,r+1} \cdot W'_n \Big|_{E_r F_r F_{*r} E_{*r}}, \\ (\sigma \cdot (C + U + W))'_t \Big|_{E_r F_r F_{*r} E_{*r}} = (\sigma \cdot (C + U + W))'_t \Big|_{E_r F_r F_{*r} E_{*r}}, \\ r = 1, p-1 \end{cases} \quad (13)$$

і подальшого «склеювання» берегів умовного розрізу Γ із використанням умов:

$$\begin{cases} \varphi \Big|_{AKLD} = \varphi \Big|_{BKLC}, \varphi \Big|_{ADD,A} = \varphi \Big|_{BCC,B}, \\ \varphi'_n \Big|_{AKLD} = \varphi'_n \Big|_{BKLC}, \varphi'_n \Big|_{ADD,A} = \varphi'_n \Big|_{BCC,B}, \end{cases} \quad (14)$$

та узгодженості величин концентрацій домішок у фільтраційному потоці і на поверхні адсорбенту завантаження, утримуваних за рахунок фізичної і хімічної адсорбції, значень температури фільтраційного потоку на умовних поверхнях розрізу Γ із використанням умов:

$$\begin{cases} C \Big|_{AKLD} = C \Big|_{BKLC}, C \Big|_{ADD,A} = C \Big|_{BCC,B}, \\ U \Big|_{AKLD} = U \Big|_{BKLC}, U \Big|_{ADD,A} = U \Big|_{BCC,B}, \\ W \Big|_{AKLD} = W \Big|_{BKLC}, W \Big|_{ADD,A} = W \Big|_{BCC,B}, \\ T \Big|_{AKLD} = T \Big|_{BKLC}, T \Big|_{ADD,A} = T \Big|_{BCC,B}, \\ C'_n \Big|_{AKLD} = C'_n \Big|_{BKLC}, C'_n \Big|_{ADD,A} = C'_n \Big|_{BCC,B}, \\ U'_n \Big|_{AKLD} = U'_n \Big|_{BKLC}, U'_n \Big|_{ADD,A} = U'_n \Big|_{BCC,B}, \\ W'_n \Big|_{AKLD} = W'_n \Big|_{BKLC}, W'_n \Big|_{ADD,A} = W'_n \Big|_{BCC,B}, \\ T'_n \Big|_{AKLD} = T'_n \Big|_{BKLC}, T'_n \Big|_{ADD,A} = T'_n \Big|_{BCC,B}. \end{cases} \quad (15)$$

хімічної регенерації:

$$\begin{cases}
 (\tilde{\sigma} \cdot c)'_t = D_1 \cdot (b_1 \cdot c''_{\varphi\varphi} + b_2 \cdot c''_{\psi\psi} + b_3 \cdot c''_{\eta\eta} + b_4 \cdot c'_{\psi} + b_5 \cdot c'_{\eta}) - \\
 - \tilde{v}^2 / \kappa_*^* \cdot c'_{\varphi} + \tilde{\beta} \cdot u - \tilde{\alpha} \cdot c + \beta^* \cdot w, \\
 (\tilde{\sigma} \cdot u)'_t = D_2 \cdot (b_1 \cdot u''_{\varphi\varphi} + b_2 \cdot u''_{\psi\psi} + b_3 \cdot u''_{\eta\eta} + b_4 \cdot u'_{\psi} + b_5 \cdot u'_{\eta}) - \tilde{\beta} \cdot u + \tilde{\alpha} \cdot c, \\
 (\tilde{\sigma} \cdot w)'_t = D_3 \cdot (b_1 \cdot w''_{\varphi\varphi} + b_2 \cdot w''_{\psi\psi} + b_3 \cdot w''_{\eta\eta} + b_4 \cdot w'_{\psi} + b_5 \cdot w'_{\eta}) - \tilde{\beta}^* \cdot w, \\
 (\tilde{\sigma} \cdot \tilde{T})'_t = D^* \cdot (b_1 \cdot \tilde{T}''_{\varphi\varphi} + b_2 \cdot \tilde{T}''_{\psi\psi} + b_3 \cdot \tilde{T}''_{\eta\eta} + b_4 \cdot \tilde{T}'_{\psi} + b_5 \cdot \tilde{T}'_{\eta}) - \\
 - \tilde{v}^2 / \kappa_*^* \cdot \tilde{T}'_{\varphi} + \tilde{\gamma} \cdot (\tilde{\beta} \cdot u - \tilde{\alpha} \cdot c) + \tilde{\gamma}^* \cdot \beta^* \cdot w, \\
 \tilde{\kappa}'_t = \tilde{\mu} \cdot u + \tilde{\mu}^* \cdot w, \\
 \tilde{\sigma}'_t = \tilde{\lambda} \cdot u + \tilde{\lambda}^* \cdot w, (\varphi, \psi, \eta) \in G^r_{\omega}, r = \overline{1, p},
 \end{cases} \quad (17)$$

зворотньої та прямої промивок:

$$\begin{cases}
 (\tilde{\sigma} \cdot c)'_t = D_1 \cdot (b_1 \cdot c''_{\varphi\varphi} + b_2 \cdot c''_{\psi\psi} + b_3 \cdot c''_{\eta\eta} + b_4 \cdot c'_{\psi} + b_5 \cdot c'_{\eta}) - \\
 - \tilde{v}^2 / \kappa_*^* \cdot c'_{\varphi} + \tilde{\beta} \cdot u - \tilde{\alpha} \cdot c - \alpha^* \cdot c, \\
 (\tilde{\sigma} \cdot u)'_t = D_2 \cdot (b_1 \cdot u''_{\varphi\varphi} + b_2 \cdot u''_{\psi\psi} + b_3 \cdot u''_{\eta\eta} + b_4 \cdot u'_{\psi} + b_5 \cdot u'_{\eta}) - \tilde{\beta} \cdot u + \tilde{\alpha} \cdot c, \\
 (\tilde{\sigma} \cdot w)'_t = D_3 \cdot (b_1 \cdot w''_{\varphi\varphi} + b_2 \cdot w''_{\psi\psi} + b_3 \cdot w''_{\eta\eta} + b_4 \cdot w'_{\psi} + b_5 \cdot w'_{\eta}) + \alpha^* \cdot c, \\
 (\tilde{\sigma} \cdot \tilde{T})'_t = D^* \cdot (b_1 \cdot \tilde{T}''_{\varphi\varphi} + b_2 \cdot \tilde{T}''_{\psi\psi} + b_3 \cdot \tilde{T}''_{\eta\eta} + b_4 \cdot \tilde{T}'_{\psi} + b_5 \cdot \tilde{T}'_{\eta}) - \\
 - \tilde{v}^2 / \kappa_*^* \cdot \tilde{T}'_{\varphi} + \tilde{\gamma} \cdot (\tilde{\beta} \cdot u - \tilde{\alpha} \cdot c) + \tilde{\gamma}^* \cdot \alpha^* \cdot c, \\
 \tilde{\kappa}'_t = \tilde{\mu} \cdot u - \tilde{\mu}^* \cdot w, \\
 \tilde{\sigma}'_t = \tilde{\lambda} \cdot u - \tilde{\lambda}^* \cdot w, (\varphi, \psi, \eta) \in G^r_{\omega}, r = \overline{1, p}
 \end{cases} \quad (18)$$

з крайовими умовами:

$$\begin{cases}
 c \Big|_{\varphi=\varphi_*} = \tilde{c}_*^*, c'_{\varphi} \Big|_{\varphi=\varphi_*} = 0, \\
 c_{\psi} \Big|_{\psi=0} = c'_{\psi} \Big|_{\psi=Q_*} = c'_{\eta} \Big|_{\eta=0} = c'_{\eta} \Big|_{\eta=Q_*} = 0, \\
 u \Big|_{\varphi=\varphi_*} = \tilde{u}_*^*, u'_{\varphi} \Big|_{\varphi=\varphi_*} = 0, \\
 u'_{\psi} \Big|_{\psi=0} = u'_{\psi} \Big|_{\psi=Q_*} = u'_{\eta} \Big|_{\eta=0} = u'_{\eta} \Big|_{\eta=Q_*} = 0, \\
 w \Big|_{\varphi=\varphi_*} = \tilde{w}_*^*, w'_{\varphi} \Big|_{\varphi=\varphi_*} = 0, \\
 w'_{\psi} \Big|_{\psi=0} = w'_{\psi} \Big|_{\psi=Q_*} = w'_{\eta} \Big|_{\eta=0} = w'_{\eta} \Big|_{\eta=Q_*} = 0, \\
 \tilde{T} \Big|_{\varphi=\varphi_*} = \tilde{T}_*^*, \tilde{T}'_{\varphi} \Big|_{\varphi=\varphi_*} = 0, \\
 \tilde{T}'_{\psi} \Big|_{\psi=0} = \tilde{T}'_{\psi} \Big|_{\psi=Q_*} = \tilde{T}'_{\eta} \Big|_{\eta=0} = \tilde{T}'_{\eta} \Big|_{\eta=Q_*} = 0,
 \end{cases} \quad (19)$$

початковими умовами:

$$\begin{cases} c|_{t=0} = \tilde{c}_0^0, u|_{t=0} = \tilde{u}_0^0, w|_{t=0} = \tilde{w}_0^0, \\ \tilde{T}|_{t=0} = \tilde{T}_0^0, \tilde{\kappa}|_{t=0} = \tilde{\kappa}_0^0, \tilde{\sigma}|_{t=0} = \tilde{\sigma}_0^0, \end{cases} \quad (20)$$

умовами узгодженості на поверхнях розділу $E'_r F'_r E'_r F'_r$ ($r = \overline{1, p-1}$) та на умовних поверхнях розрізу:

$$\left\{ \begin{array}{l} \lim_{\varphi \rightarrow \varphi_{r-0}^+} c = \lim_{\varphi \rightarrow \varphi_{r+0}^+} c, \quad \lim_{\varphi \rightarrow \varphi_{r-0}^+} u = \lim_{\varphi \rightarrow \varphi_{r+0}^+} u, \\ \lim_{\varphi \rightarrow \varphi_{r-0}^+} w = \lim_{\varphi \rightarrow \varphi_{r+0}^+} w, \quad \lim_{\varphi \rightarrow \varphi_{r-0}^+} \tilde{T} = \lim_{\varphi \rightarrow \varphi_{r+0}^+} \tilde{T}, \\ \lim_{\varphi \rightarrow \varphi_{r-0}^+} (D_{1,r} \cdot c'_\varphi - \kappa_r^* \cdot c) = \lim_{\varphi \rightarrow \varphi_{r+0}^+} (D_{1,r+1} \cdot c'_\varphi - \kappa_{r+1}^* \cdot c), \\ \lim_{\varphi \rightarrow \varphi_{r-0}^+} D_{2,r} \cdot u'_\varphi = \lim_{\varphi \rightarrow \varphi_{r+0}^+} D_{2,r+1} \cdot u'_\varphi, \\ \lim_{\varphi \rightarrow \varphi_{r-0}^+} D_{3,r} \cdot w'_\varphi = \lim_{\varphi \rightarrow \varphi_{r+0}^+} D_{3,r+1} \cdot w'_\varphi, \\ \lim_{\varphi \rightarrow \varphi_{r-0}^+} (D_r^* \cdot \tilde{T}'_\varphi - \kappa_r^* \cdot \tilde{T}) = \lim_{\varphi \rightarrow \varphi_{r+0}^+} (D_{r+1}^* \cdot \tilde{T}'_\varphi - \kappa_{r+1}^* \cdot \tilde{T}), \\ \lim_{\varphi \rightarrow \varphi_{r-0}^+} (D_{1,r} \cdot c'_\varphi - \tilde{\kappa} \cdot c + D_{2,r} \cdot u'_\varphi + D_{3,r} \cdot w'_\varphi) = \\ = \lim_{\varphi \rightarrow \varphi_{r+0}^+} (D_{1,r+1} \cdot c'_\varphi - \tilde{\kappa} \cdot c + D_{2,r+1} \cdot u'_\varphi + D_{3,r+1} \cdot w'_\varphi), \\ \lim_{\varphi \rightarrow \varphi_{r-0}^+} (\tilde{\sigma} \cdot (c + u + w))'_t = \lim_{\varphi \rightarrow \varphi_{r+0}^+} (\tilde{\sigma} \cdot (c + u + w))'_t, \quad r = \overline{1, p-1}, \end{array} \right. \quad (21)$$

$$\left\{ \begin{array}{l} c|_{\eta=0, \psi=\tilde{\psi}} = c|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad c|_{\psi=0} = c|_{\psi=Q}, \\ u|_{\eta=0, \psi=\tilde{\psi}} = u|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad u|_{\psi=0} = u|_{\psi=Q}, \\ w|_{\eta=0, \psi=\tilde{\psi}} = w|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad w|_{\psi=0} = w|_{\psi=Q}, \\ \tilde{T}|_{\eta=0, \psi=\tilde{\psi}} = \tilde{T}|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad \tilde{T}|_{\psi=0} = \tilde{T}|_{\psi=Q}, \\ c'_n|_{\eta=0, \psi=\tilde{\psi}} = c'_n|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad c'_n|_{\psi=0} = c'_n|_{\psi=Q}, \\ u'_n|_{\eta=0, \psi=\tilde{\psi}} = u'_n|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad u'_n|_{\psi=0} = u'_n|_{\psi=Q}, \\ w'_n|_{\eta=0, \psi=\tilde{\psi}} = w'_n|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad w'_n|_{\psi=0} = w'_n|_{\psi=Q}, \\ \tilde{T}'_n|_{\eta=0, \psi=\tilde{\psi}} = \tilde{T}'_n|_{\eta=0, \psi=Q, -\tilde{\psi}}, \quad \tilde{T}'_n|_{\psi=0} = \tilde{T}'_n|_{\psi=Q}, \end{array} \right. \quad (22)$$

де

$$\begin{aligned} c &= c(\varphi, \psi, \eta, t) = \overline{C(x(\varphi, \psi, \eta), y(\varphi, \psi, \eta), z(\varphi, \psi, \eta), t)}, \dots, \\ \tilde{v}_r &= \tilde{v}_r(\varphi, \psi, \eta) \quad (r = \overline{1, p}), \quad b_{r,s} = b_{r,s}(\varphi, \psi, \eta) \quad (r = \overline{1, p}, s = \overline{1, 5}), \\ b_{r,1} &= \varphi_x'^2 + \varphi_y'^2 + \varphi_z'^2 = \tilde{v}_r^2 / (\kappa_r^0)^2, \quad b_{r,2} = \psi_x'^2 + \psi_y'^2 + \psi_z'^2, \end{aligned}$$

$$b_{r,3} = \eta_x'' + \eta_y'' + \eta_z'', \quad b_{r,4} = \psi_{xx}'' + \psi_{zz}'', \quad b_{r,5} = \eta_{xx}'' + \eta_{yy}'' + \eta_{zz}'', \quad \widehat{\psi} \in [0, Q_*/2].$$

Алгоритмізація розв'язання. Аналогічно до [14, 15] для області $G_w \times (0, \infty)$ знайдено числово-асимптотичні наближення

$$c = \{c_r, (\varphi, \psi, \eta) \in G_w^r, r = \overline{1, p}\}, \quad u = \{u_r, (\varphi, \psi, \eta) \in G_w^r, r = \overline{1, p}\}, \\ w = \{w_r, (\varphi, \psi, \eta) \in G_w^r, r = \overline{1, p}\}, \quad \tilde{T} = \{\tilde{T}_r, (\varphi, \psi, \eta) \in G_w^r, r = \overline{1, p}\}, \\ \tilde{\kappa} = \{\tilde{\kappa}_r, (\varphi, \psi, \eta) \in G_w^r, r = \overline{1, p}\}, \quad \tilde{\sigma} = \{\tilde{\sigma}_r, (\varphi, \psi, \eta) \in G_w^r, r = \overline{1, p}\}$$

розв'язків модельних задач прогнозування відповідно режимів фільтрування (16), (19)-(22), хімічної регенерації (17), (19)-(22), зворотної та прямої промивок (18)-(22) з точністю $O(\varepsilon^{n+1})$ у вигляді таких рядів:

$$c_r = \sum_{i=0}^n \varepsilon^i \cdot c_{r,i} + P_{1,r}, \quad u_r = \sum_{i=0}^n \varepsilon^i \cdot u_{r,i} + P_{2,r}, \quad w_r = \sum_{i=0}^n \varepsilon^i \cdot w_{r,i} + P_{3,r}, \\ \tilde{T}_r = \sum_{i=0}^n \varepsilon^i \cdot \tilde{T}_{r,i} + P_{4,r}, \quad \tilde{\kappa}_r = \sum_{i=0}^n \varepsilon^i \cdot \tilde{\kappa}_{r,i} + P_{5,r}, \quad \tilde{\sigma}_r = \sum_{i=0}^n \varepsilon^i \cdot \tilde{\sigma}_{r,i} + P_{6,r},$$

$$\text{де } P_{s,r} = \sum_{i=0}^{n+1} \varepsilon^i \cdot (\tilde{P}_{s,r,i} + \tilde{\tilde{P}}_{s,r,i}) + \sum_{i=0}^{2n+1} \varepsilon^{i/2} \cdot (\hat{P}_{s,r,i} + \hat{\hat{P}}_{s,r,i} + \check{P}_{s,r,i} + \check{\check{P}}_{s,r,i}) + R_{s,r,n+1}$$

($s = \overline{1, 6}$), $c_{r,i} = c_{r,i}(\varphi, \psi, \eta, t)$, $u_{r,i} = u_{r,i}(\varphi, \psi, \eta, t)$, $w_{r,i} = w_{r,i}(\varphi, \psi, \eta, t)$,

$\tilde{T}_{r,i} = \tilde{T}_{r,i}(\varphi, \psi, \eta, t)$, $\tilde{\kappa}_{r,i} = \tilde{\kappa}_{r,i}(\varphi, \psi, \eta, t)$, $\tilde{\sigma}_{r,i} = \tilde{\sigma}_{r,i}(\varphi, \psi, \eta, t)$ ($r = \overline{1, p}$,

$i = \overline{0, n}$) — члени регулярних частин асимптотик, $\tilde{P}_{s,r,i} = \tilde{P}_{s,r,i}(\tilde{\varphi}_{r-1}, \psi, \eta, t)$,

$\tilde{\tilde{P}}_{s,r,i} = \tilde{\tilde{P}}_{s,r,i}(\tilde{\tilde{\varphi}}_r, \psi, \eta, t)$ ($s = \overline{1, 6}$, $r = \overline{1, p}$, $i = \overline{0, n+1}$) — функції типу

примежового шару відповідно в околах $\varphi = \varphi_{*r}^*$ ($r = \overline{0, p}$) (поправки на вході, виході і в околі поверхонь розділу шарів фільтру),

$\hat{P}_{s,r,i} = \hat{P}_{s,r,i}(\varphi, \tilde{\psi}, \eta, t)$, $\hat{\hat{P}}_{s,r,i} = \hat{\hat{P}}_{s,r,i}(\varphi, \tilde{\tilde{\psi}}, \eta, t)$, $\check{P}_{s,r,i} = \check{P}_{s,r,i}(\varphi, \psi, \tilde{\eta}, t)$,

$\check{\check{P}}_{s,r,i} = \check{\check{P}}_{s,r,i}(\varphi, \psi, \tilde{\tilde{\eta}}, t)$ ($s = \overline{1, 6}$, $r = \overline{1, p}$, $i = \overline{0, 2n+1}$) — функції типу

примежового шару відповідно в околах $\psi = 0$, $\psi = Q_*$, $\eta = 0$, $\eta = Q^*$

(поправки на бічній стінці фільтру і берегах умовного розрізу Γ),

$\tilde{\varphi}_r = (\varphi - \varphi_{*r}^*)/\varepsilon$ ($r = \overline{0, p-1}$), $\tilde{\tilde{\varphi}}_r = (\varphi_{*r}^* - \varphi)/\varepsilon$ ($r = \overline{1, p}$), $\tilde{\psi} = \psi/\sqrt{\varepsilon}$,

$\tilde{\tilde{\psi}} = (Q_* - \psi)/\sqrt{\varepsilon}$, $\tilde{\eta} = \eta/\sqrt{\varepsilon}$, $\tilde{\tilde{\eta}} = Q^* - \eta/\sqrt{\varepsilon}$ — відповідні їм регуляри-

ризуючі перетворення (розтяги), $R_{s,r,n+1}(\varphi, \psi, \eta, t, \varepsilon)$ ($s = \overline{1, 6}$,

$r = \overline{1, p}$) — залишкові члени.

Результати числових розрахунків. У ході комп'ютерного експерименту для прогнозування процесів «фільтрування-регенерація» шви-

дкого конусоподібного адсорбційного фільтру довжиною 1 м з двома однорідними шарами пористого завантаження проведено дослідження його основних технологічних режимів фільтрування, зворотної промивки, хімічної регенерації та прямої промивки. Форму модельного фільтру та поверхню розділу EFF_*E_* описано поверхнями:

$$f_*(x, y, z) = x^2 + y^2 + z^2 - 4, \quad f^*(x, y, z) = x^2 + y^2 + z^2 - 1, \quad f^{**}(x, y, z) = 7,54863 \cdot x^2 - y^2 - z^2, \quad f_*^*(x, y, z) = x^2 + y^2 + z^2 - 2,25.$$

Функції $f_*(x, y, z)$, $f^*(x, y, z)$ та $f^{**}(x, y, z)$ підібрані так, щоб у режимі фільтрування забезпечити «монотонне звуження» в напрямку від його входу до виходу та взаємну ортогональність граней уздовж ребер. При цьому кожен шар фільтру характеризується різними початковими значеннями коефіцієнтів фільтрації $\kappa_1^* = 8,5$ м/добу, $\kappa_2^* = 5,6$ м/добу, активної пористості $\tilde{\sigma}_1^* = 0,41$, $\tilde{\sigma}_2^* = 0,38$, дифузії домішок $D_{1,1} = 0,0003$ м²/добу, $D_{1,2} = 0,0002$ м²/добу, $D_{2,1} = 0,000005$ м²/добу, $D_{2,2} = 0,000002$ м²/добу, $D_{3,1} = 0,0000007$ м²/добу, $D_{3,2} = 0,0000003$ м²/добу і температуропровідності фільтраційного потоку $D_1^* = 0,0005$ м²/добу, $D_2^* = 0,0003$ м²/добу.

У результаті побудовано розрахункову сіткову область G_z (рис. 3 а) при $n = n_1 + n_2 = 33$, $n_1 = 22$, $n_2 = 11$, $m = 17$, $l = 9$, $\varphi_* = \varphi_0 = 0$, $\varphi^* = \varphi_{33} = 14,5$. Значення φ^* вибиралося так, щоб середня швидкість фільтрації $v_{\text{сер}}$ вздовж модельного фільтра становила 5 м/год. Для області G_z знайдено фільтраційну витрату $Q = 84,1$ м³/год і значення потенціалу на поверхні розділу $\varphi_*^* = \varphi_{22} = 8,2$ та обчислено величини швидкостей фільтрації (рис. 3 б) і функції $b_i(\varphi, \psi, \eta)$ ($i = 1, 5$).

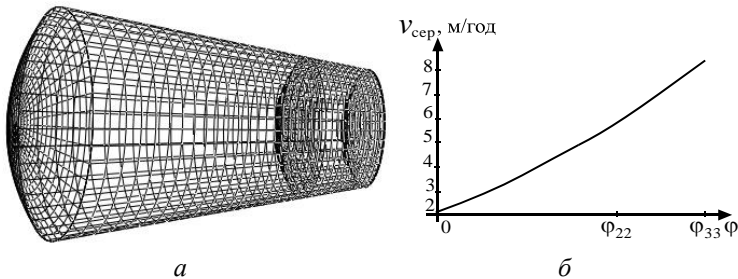


Рис. 3. Результати побудови фільтраційного поля конусоподібного фільтру для модельної задачі прогнозування технологічного режиму фільтрування:

а — розрахована сіткова область;
б — розподіл середніх величин швидкості фільтрації вздовж висоти фільтру

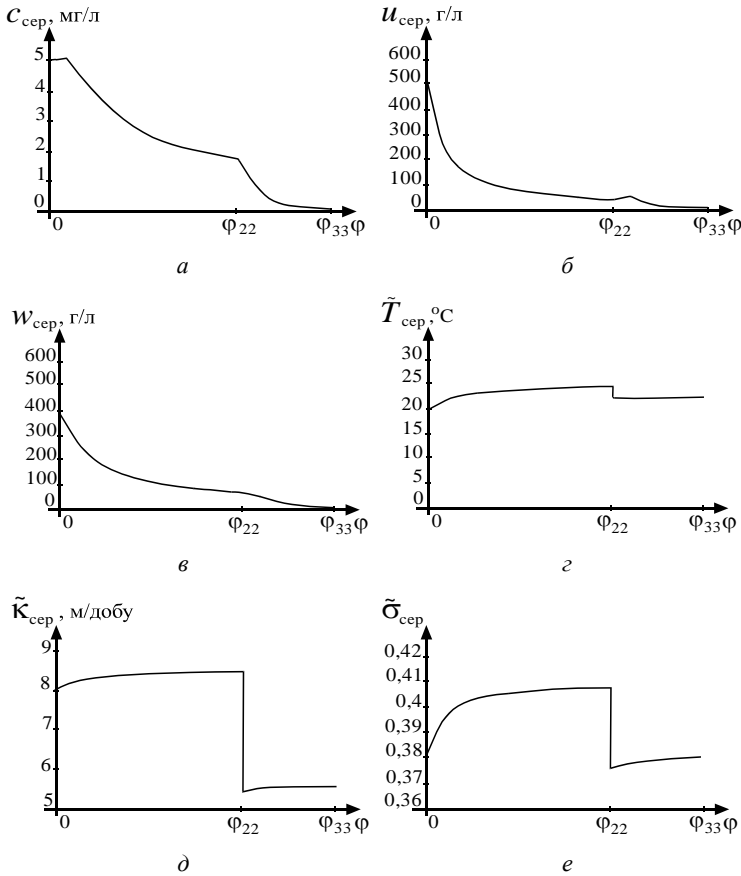


Рис. 4. Розподіл середніх величин концентрацій домішок відповідно у фільтраційній течії (а), адсорбованих на поверхні адсорбенту завантаження при фізичній (б) та хімічній (в) адсорбції, температури фільтраційного потоку (г), коефіцієнта фільтрації (д) і активної пористості (е) вздовж висоти фільтру у момент часу $t_3 = 15$ год для модельної задачі прогнозування режиму фільтрування

На рис. 4 представлено розподіли середніх величин $c_{\text{сеп}}(\varphi, t)$, $u_{\text{сеп}}(\varphi, t)$, $w_{\text{сеп}}(\varphi, t)$, $\tilde{T}_{\text{сеп}}(\varphi, t)$, $\tilde{\kappa}_{\text{сеп}}(\varphi, t)$ і $\tilde{\sigma}_{\text{сеп}}(\varphi, t)$ вздовж висоти завантаження фільтру у момент часу $t_3 = 15$ год для модельної задачі прогнозування режиму фільтрування при $c_{\text{кр}} = 0,00005$ г/л, $c_* = 0,005$ г/л, $c_0^0 = 0$, $u_0^0 = 0$, $w_0^0 = 0$, $\tilde{T}_0^0 = 20$ °С, $\kappa_0^0 = \kappa_*^*$, $\sigma_0^0 = \sigma_*^*$,

$$\begin{aligned} \tilde{\alpha}_1 &= 0,3 - 0,01 \cdot \tilde{v} - 0,008 \cdot \tilde{T} \text{ c}^{-1}, & \tilde{\alpha}_2 &= 0,38 - 0,01 \cdot \tilde{v} - 0,011 \cdot \tilde{T} \text{ c}^{-1}, & \tilde{\beta}_s &= \\ &= 0,00001 \cdot \tilde{\alpha}_s \text{ c}^{-1} \quad (s = 1, 2), & \tilde{\alpha}_1^* &= 0,25 - 0,002 \cdot \tilde{v} - 0,01 \cdot \tilde{T} \text{ c}^{-1}, & \tilde{\alpha}_2^* &= 0,35 - \\ & - 0,001 \cdot \tilde{v} - 0,011 \cdot \tilde{T} \text{ c}^{-1}, & \tilde{\gamma}_1 &= 1200 \text{ }^\circ\text{C} \cdot \text{л/г}, & \tilde{\gamma}_2 &= 1500 \text{ }^\circ\text{C} \cdot \text{л/г}, & \tilde{\gamma}_1^* &= \\ &= 60000 \text{ }^\circ\text{C} \cdot \text{л/г}, & \tilde{\gamma}_2^* &= 80000 \text{ }^\circ\text{C} \cdot \text{л/г} & \tilde{\mu}_1 &= 0,000025 \text{ л/(г} \cdot \text{с)}, & \tilde{\mu}_2 &= \\ &= 0,000023 \text{ л/(г} \cdot \text{с)}, & \tilde{\mu}_1^* &= 0,000025 \text{ л/(г} \cdot \text{с)}, & \tilde{\mu}_2^* &= 0,000023 \text{ л/(г} \cdot \text{с)}, & \tilde{\lambda}_1 &= \\ &= 0,000031 \text{ л/(г} \cdot \text{с)}, & \tilde{\lambda}_2 &= 0,000028 \text{ л/(г} \cdot \text{с)}, & \tilde{\lambda}_1^* &= 0,000021 \text{ л/(г} \cdot \text{с)}, & \tilde{\lambda}_2^* &= \\ &= 0,00002 \text{ л/(г} \cdot \text{с)}. \end{aligned}$$

Висновки. Запропоновані моделі для прогнозування процесів «фільтрування-регенерація» швидких конусоподібних адсорбційних фільтрів з кусково-однорідними пористими завантаженнями у комплексі за рахунок урахування впливу температурних ефектів на внутрішню кінетику масопереносу аналогічно [17, 18] дають змогу шляхом проведення комп'ютерних експериментів більш якісно дослідити зміну концентрацій домішок у фільтраційному потоці і на поверхні адсорбенту завантаження, утримуваних за рахунок фізичної і хімічної адсорбції, температури фільтраційного потоку, коефіцієнта фільтрації та активної пористості у кожному шарі завантаження вздовж висоти фільтру та на їх основі спрогнозувати більш оптимальні варіанти використання адсорбентів кожного шару завантаження і часові інтервали роботи фільтрів відповідно у режимах фільтрування, зворотньої промивки, хімічної регенерації та прямої промивки при дотриманні сталих швидкостей відповідних режимів. У перспективі планується розповсюдження даної методики на нанофільтри [19].

Список використаних джерел:

1. Двадненко М. В., Привалова Н. М., Кудасева І. Ю., Степура А. Г. Адсорбційна очистка сточних вод. *Современные наукоемкие технологии*. 2010. № 10. С. 214-215.
2. Корнева Д. А., Куров Л. Н. Адсорбційна очистка — ефективний метод очистки сточних вод і підготовки води для господарсько-питвєвого водопользования. *Успехи современного естествознания*. 2011. № 7. С. 129.
3. Edzwald J. Water Quality & Treatment. A Handbook on Drinking Water. McGraw-Hill Professional, 2010. 1996 p.
4. Hendricks D. W. Fundamentals of water treatment unit processes: physical, chemical, and biological. Boca Raton: CRC Press, 2011. 883 p.
5. Орлов В. О., Мартинов С. Ю., Зошук А. М. Проектування станцій прояснення та знебарвлення води. Рівне: НУВГП, 2007. 252 с.
6. Sabadash V., Mylanyk O., Matsuska O., Gumnytsky J. Kinetic regularities of copper ions adsorption by natural zeolite. *Chemistry and chemical technology*. 2017. Vol. 11, № 4. P. 459-462.
7. Мальований М. С., Сакалова Г. В., Василінич Т. М., Братушак А. О. Дослідження здатності глинистих сорбентів до адсорбції іонів важких мета-

- лів. *Сталий розвиток XXI століття: управління, технології, моделі* — *Дискусії 2016* : колективна монографія / за наук. ред. проф. Є. В. Хлобистова. Черкаси: Видавець Чабаненко Ю., 2016. 590 с.
8. Сакалова Г. В., Василеніч Т. М. Дослідження ефективності очищення стічних вод від іонів важких металів з використанням природних адсорбентів : монографія. Вінниця: ТОВ «Твори», 2019. 92 с.
 9. Невзорова А. Б. Основы автоматизации систем водоснабжения и водоотведения : пособие. Гомель: УО «БелГУТ», 2005. 115 с.
 10. Бомба А. Я., Сафоник А. П. Моделювання нелінійно-збурених процесів очищення рідин від багатоконпонентних забруднень : монографія. Рівне: НУВГП, 2017. 296 с.
 11. Душкин С. С. Физические методы водоподготовки : учеб. пособие. Киев: УМК ВО, 1989. 151 с.
 12. Минц Д. М. Теоретические основы технологии очистки воды. М.: Стройиздат, 1964. 156 с.
 13. Макаревич Н. А., Богданович Н. И. Теоретические основы адсорбции : учебное пособие. Архангельск: САФУ, 2015. 362 с.
 14. Климяк Ю. Є., Бомба А. Я. Комп'ютерне прогнозування адсорбційної доочистки води від домішок у швидких багатощарових фільтрах конусо-подібної форми. *Математичне та комп'ютерне моделювання. Серія: Технічні науки* : зб. наук. праць. Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2019. Вип. 20. С. 51-68. DOI: 10.32626/2308-5916.2019-20.51-68.
 15. Бомба А. Я., Климяк Ю. Є. Математичне моделювання просторових сингулярно-збурених процесів типу фільтрація-конвекція-дифузія : монографія. Рівне: ТЗОВ фірма «Ассоль», 2014. 273 с.
 16. Klimjuk Ju. Je. Modelling of spatial filtration fields for one class of rapid multilayer filters of cone-shaped form with piecewise-homogeneous porous loads. *Proceedings of III International scientific conference «Science: new goals»*. London: SI Universum, 2017. P. 4-10.
 17. Бомба А. Я., Климяк Ю. Є. Прогнозування зміни характеристик кусково-однорідних пористих завантажень швидких багатощарових фільтрів. *Вісник НТУ «ХП»*. Серія: *Математичне моделювання в техніці та технологіях*. Харків: НТУ «ХП», 2018. № 27 (1303). С. 8-15.
 18. Бомба А. Я., Климяк Ю. Є. Прогнозування оптимального використання фільтруючих матеріалів у швидких багатощарових фільтрах із кусково-однорідними пористими завантаженнями. *Журнал обчислювальної та прикладної математики* : наук. журн. Київ. нац. ун-т ім. Тараса Шевченка. Київ: ТВіМС, 2017. Вип. 1 (124). С. 3-14.
 19. Petryk M. R., Khimichb A., Petryk M. M., Fraissard J. Experimental and computer simulation studies of dehydration on microporous adsorbent of natural gas used as motor fuel. *Fuel*. 2019. Vol. 239. P. 1324-1330.

COMPUTER PREDICTION OF «FILTRATION-REGENERATION» PROCESSES OF RAPID CONE-SHAPED ADSORPTION FILTERS WITH PIECEWISE-HOMOGENEOUS POROUS LOADS

To predict the processes of «filtration-regeneration» of fast cone-shaped adsorption filters with piecewise homogeneous porous loads, math-

emational models of their main technological modes are formed: filtration, backwashing, chemical regeneration and direct washing, taking into account the influence of temperature effects on the internal kinetics of mass transfer while maintaining constant speeds of the respective modes. Given that in the filtration mode the convective components of heat and mass transfer and physical and chemical adsorption prevail over the contribution of diffusion and physical desorption, in the chemical regeneration mode the convective components of heat and mass transfer and physical and chemical desorption prevail over the contribution of diffusion and physical adsorption. modes of reverse and direct washing convective components of heat and mass transfer and physical desorption and chemical adsorption prevail over the contribution of diffusion and physical adsorption, algorithms of numerical-asymptotic approximations of solutions of corresponding nonlinear singularly perturbed boundary value problems, bounded by given, smooth, orthogonal to each other along the edges, two equipotential surfaces and one flow surface and divided into several subregions by some given equipotential surfaces. The proposed models in the complex by taking into account the effect of changes in temperature and filtration flow rate along the filter height on the coefficients that characterize the mass transfer rates during physical and chemical adsorption and desorption, filtration coefficient, allow computer experiments to better investigate changes in impurity concentrations in filtration flow and on the surface of the loading adsorbent, retained by physical and chemical adsorption, filtration flow temperature, filtration coefficient and active porosity in each loading layer along the filter height and based on them predict more optimal use of adsorbents of each loading layer and time intervals of filters, respectively, in filtration, backwash, chemical regeneration and direct washing with constant speeds of the respective modes.

Key words: *mathematical model, prediction, filtration, washing, chemical regeneration, impurity, adsorption, desorption, temperature, rapid cone-shaped filter, piecewise homogeneous porous load.*

Отримано: 28.05.2020

УДК 004.78;004.891.2;007.3

DOI: 10.32626/2308-5916.2020-21.102-113

О. Є. Коваленко, канд. техн. наук,**В. Л. Косолапов**, канд. техн. наук

Інститут проблем математичних машин і систем НАН України, м. Київ

МОДЕЛІ ОПТИМІЗАЦІЇ ЗАВАНТАЖЕННЯ МЕРЕЖІ РОЗПОДІЛЕНИХ СИТУАЦІЙНИХ ЦЕНТРІВ

У статті обґрунтовано актуальність створення та сформульовано математичну постановку задачі забезпечення технічної інтероперабельності в мережі розподілених ситуаційних центрів на основі конвергентних технологічних рішень. Технічна інтероперабельність розглядається в контексті оптимізації розподілу завдань ситуаційного управління між окремими ситуаційними центрами в розподіленій мережі та мінімізації часу обробки інформації. Для цього запропоновано модель та алгоритми оптимізації виконання завдань учасниками процесу ситуаційного управління в мережі розподілених ситуаційних центрів. Модель оптимізації технічної інтероперабельності представлена у формі задачі булевого цілочисельного програмування з пошуком локального екстремуму цільової функції. Інформаційно-аналітичний процес формується як запит, який визначає потребу в інформаційних ресурсах, аналітичних та експертних спроможностях та часі обслуговування в кожному локальному ситуаційному центрі. Задача оптимізації завантаження мережі розподілених ситуаційних центрів полягає у перерозподілі навантаження між вузлами мережі шляхом вирівнювання використання організаційних, людських, технічних та інформаційних ресурсів для своєчасного та якісного прийняття рішень в ситуаційних центрах, які обслуговують учасників процесу ситуаційного управління на різних рівнях. Задача мінімізації часу обслуговування в мережі розподілених ситуаційних центрів є двоїстою до задачі перерозподілу навантаження і розв'язується з використанням стохастичного квазіградієнтного методу проєкції. Розроблені алгоритми розв'язання сформульованих задач та наведено тестовий приклад застосування. Показано потенціал подальшого вдосконалення ситуаційних систем управління стратегічним плануванням на основі управління спроможностями цільових систем.

Ключові слова: *система ситуаційного управління, мережа розподілених ситуаційних центрів, конвергентна система, технічна інтероперабельність, оптимізація систем.*

Вступ. Актуальною проблемою розвитку інформаційних технологій та систем є проблема створення *мережі розподілених ситуаційних центрів* (МРСЦ) на основі конвергентних технологічних рішень. У зв'язку з цим особливого значення набуває розвиток інформаційних

сервісів інформаційно-аналітичних систем на базі МРСЦ для підтримки та прийняття рішень на різних рівнях державного управління. Розробка та впровадження на практиці мережі ситуаційних центрів (СЦ) є невід'ємною складовою створення ефективних систем для вирішення завдань державного управління різної складності.

Стандарт ISO/IEC 10746 [1] визначає основні поняття, необхідні для опису еталонної моделі систем відкритої розподіленої обробки (EMCBPO, Reference Model of Open Distributed Processing (RM-ODP) Systems.) з п'яти встановлених точок зору. Цей стандарт забезпечує ґрунтовний базис для структуризації специфікацій широкомасштабних розподілених систем.

Еталонна модель для опису системи, що пропонується ISO/IEC 10746, включає чотири основні елементи:

- об'єктно-модельний підхід до специфікації системи;
- специфікацію системи в термінах окремих, але взаємопов'язаних специфікацій з певних точок зору;
- визначення системної інфраструктури, що забезпечує прозорість поширення (дистрибуції) системних додатків;
- шаблон (модель) для оцінки відповідності системи.

Концепція RM-ODP встановлює п'ять загальних та взаємодоповнюючих точок зору на систему та її середовище:

- точка зору на організацію, яка фокусується на призначенні, обсязі та політиці системи і описує вимоги діяльності та шляхи їх виконання;
- точка зору інформації, яка зосереджена на семантиці інформації та виконаній обробці інформації і описує інформацію, якою керує система, та структуру та тип вмісту допоміжних даних;
- обчислювальна точка зору, яка, шляхом функціональної декомпозиції, дозволяє розділити систему на об'єкти, що взаємодіють між собою через інтерфейси, і описує функціональність, що надається системою, та її функціональну декомпозицію;
- інженерна точка зору, яка фокусується на механізмах та функціях, необхідних для підтримки розподіленої взаємодії між об'єктами в системі і описує розподіл обробки, що виконується системою для управління інформацією та забезпечення функціональності;
- технологічна точка зору, яка фокусується на виборі технології системи і описує технології, обрані для забезпечення обробки, функціональності та представлення інформації.

У контексті стандарту ISO/IEC 10746 поняття інтероперабельності визначається як здатність об'єктів до взаємодії (співпраці), тобто здатність взаємно передавати інформацію для обміну подіями, пропозиціями, запитами, результатами, зобов'язаннями та потоками [2].

При впровадженні інформаційно-аналітичного процесу обробки інформації під час підготовки альтернатив та прийняття рішень у МРСЦ

виникає завдання управління послідовністю потоків обробки інформації та управління таким чином, щоб система СЦ, пов'язана з вирішенням однієї проблеми, могла забезпечити необхідний рівень інтероперабельності технологій інформаційного, аналітичного та експертного обслуговування учасників процесів ситуаційного управління (УПСУ) з мінімальними витратами часу. Множина УПСУ включає експертів, аналітиків, осіб, які приймають рішення, та інших зацікавлених сторін та осіб. Інформаційно-аналітичний процес ситуаційного управління формується у вигляді застосунку, що визначає потребу в інформаційних ресурсах та вимоги до аналітичних та експертних спроможностей, а також часових обмежень обслуговування запитів окремими СЦ. Конвергентний підхід передбачає композицію гетерогенних компонентів в єдиній системі [3], забезпечуючи дотримання 12 основних принципів інтероперабельності [4]:

- 1) субсидіарність (виконання дій в рамках кооперації на основі оцінювання їх обґрунтованості з точки зору наявних *спроможностей* на відповідних рівнях, тобто керівний орган повинен виконувати допоміжну функцію, виконуючи лише ті завдання, які неможливо виконати на більш локальному рівні) та пропорційність;
- 2) відкритість;
- 3) прозорість;
- 4) повторне використання;
- 5) технологічна нейтральність та можливість взаємодії з даними;
- 6) орієнтація на користувача;
- 7) включність (інклюзивність) та доступність;
- 8) безпека та конфіденційність;
- 9) багатомовність;
- 10) спрощення адміністрування;
- 11) збереженість інформації;
- 12) оцінка продуктивності та ефективності.

Ці основні принципи поділяються на наступні 4 категорії:

- принцип, що встановлює контекст для заходів управління щодо інтероперабельності (1-й принцип);
- основні принципи взаємодії (принципи 2-5);
- принципи, що стосуються загальних потреб та очікувань користувачів (принципи 6-9);
- фундаментальні принципи співробітництва між адміністративними органами (принципи 10-12).

Проблема ефективного розподілу завдань в мережі автоматизованих комп'ютерних систем (АСУ) була розглянута в різний час в роботах В. М. Глушкова [5], А. О. Морозова [6] та інших авторів. МРСЦ — це розвиток та узагальнення концепції мережі АСУ у сучасних умовах. Одним із ключових механізмів ефективного функціонування МРСЦ є забезпечення інтероперабельності між його складовими.

Відповідно до визначення інтерооперабельності у стандарті ISO/IEC 24765 [2] та європейської моделі інтерооперабельності (European Interoperability Framework, EIF), прийнятого в ЄС [4], базовим рівнем інтерооперабельності є технічна інтерооперабельність (рис. 1).

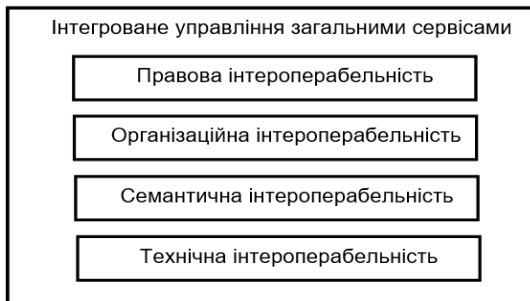


Рис. 1. Чотири рівні інтерооперабельності

На основі EIF [4] була розроблена Європейська еталонна архітектура інтерооперабельності (EIRA) [7]. Різні аспекти інтерооперабельності були розглянуті в роботах [8-12].

Формальна постановка задачі. Розглядається гетерогенна мережа СЦ. Нехай M — множина користувачів (УПСУ) МРСЦ; n кількість СЦ (в нашому випадку, $|M| > n$, тобто, багатокористувальницькі та різні за змістом вузли моніторингу та аналізу інформації, що обробляється (економічна, екологічна, військова тощо); c_k оцінка інформаційних потреб k -го ($k \in M$) користувача; R_i оцінка пропускної здатності i -го СЦ з МРСЦ; P_{ij} — потік перерозподілу інформації між i -м та j -м СЦ ($i, j = 1, 2, \dots, n$) в МРСЦ; q — максимально допустимий потік інформації між двома СЦ.

Введемо змінну

$$x_{ki} = \begin{cases} 1, & \text{якщо } k\text{-й користувач обслуговується у } i\text{-му СЦ з МРСЦ, } i = 1, \dots, n \\ 0, & \text{в іншому випадку.} \end{cases}$$

Формалізована модель фактичного завантаження H_i i -го СЦ ($i = 1, 2, \dots, n$) в МРСЦ в процесі обслуговування користувачів має вид:

$$H_i = \max_i \sum_{k \in M} c_k x_{ki} \rightarrow \min, \quad (1)$$

$$\sum_{k \in M} c_k x_{ki} \leq R_i, i = 1, 2, \dots, n, \quad (2)$$

$$\frac{1}{M} \sum_{i=1}^n x_{ki} \leq 1, k \in M, \quad (3)$$

$$P_{ij}(x_{ki}, x_{kj}) \leq q; k \in M; i \neq j; i, j = 1, 2, \dots, n. \quad (4)$$

Умова (2) виражає обмеження на інформаційну пропускну здатність i -го СЦ, умова (3) — це вимога обов'язкового обслуговування всіх користувачів (УПСУ) та належність користувача до групи обслуговування локальним СЦ, умова (4) регулює потоки послуг між користувачами (УПСУ) МРСЦ.

Задача представлена співвідношеннями (1)-(4) — це задача булевого цілочисельного програмування. Для розв'язання задачі (1)-(4) запропоновано алгоритм, який реалізує метод пошуку з локальною оптимізацією [13], що дозволяє отримати розв'язок задачі (1)-(4) з прийнятною точністю для практичної роботи МРСЦ.

Модель оптимізації завантаження МРСЦ. Розглянемо проблему обслуговування УПСУ в розподіленій системі, яка може складатися з декількох локальних (регіональних) СЦ і спрямована на вирішення різних за змістом проблем. Як правило, при організації взаємодії кількох СЦ проводяться відповідні експертні оцінювання та розрахунки їх завантаження з метою своєчасного забезпечення потреб УПСУ. Однак з часом завантаженість може суттєво змінюватися (наприклад, через появу нових завдань та управлінських потоків інформації), і спроможності МРСЦ перестають відповідати вимогам УПСУ стосовно часу обробки та кількості запитів. Задача полягає в тому, щоб перерозподілити навантаження МРСЦ для забезпечення рівномірного навантаження на організаційні, людські, технічні та інформаційні ресурси і своєчасного та якісного прийняття рішень в СЦ, які обслуговують УПСУ на різних рівнях.

Вихідні умови для розв'язання задачі (1)-(4) формуються у вигляді матриці A . Рядки матриці являють собою оцінку інформаційних потреб УПСУ, які обслуговуються у МРСЦ. Стовпці матриці A — це фактичне завантаження МРСЦ, тобто H_i ($i = 1, 2, \dots, n$).

Алгоритм вирішення проблеми обслуговування клієнтів у МРСЦ складається з п'яти кроків.

Крок 1. Якщо всі стовпці матриці A переглянуті, переходимо до кроку 5. В іншому випадку виконуємо такі дії. При аналізі в матриці A шукаємо стовпчик з максимальною сумою рядків, тобто з максимальним завантаженням конкретного СЦ. Нехай таким стовпцем буде стовпець i . Наступним кроком є закріплення області з трьох стовпців $i, i + 1, i - 1$ ($i = 1, 2, \dots, n$). Якщо $i + 1$ або $i - 1$ не існують (для кінцевих елементів), тоді стовпець та його окіл будуть усічені.

Крок 2. У i -му стовпці ненульовий елемент $c_k x_{ki}$ і перевіряється можливість зсуву цього елемента вправо або вліво відповідно до умов (2)-(4). Якщо такі зсуви можливі, то вони виконуються. Для запам'ятовування переміщень елементів матриці A на наступному кроці алгоритму (з метою визначення можливих переміщень на наступному кроці) вводимо допоміжну матрицю B , елементами якої є числа з множини $(-1, 0, 1)$. Початкові значення елементів матриці рівні 0.

При переміщенні елемента $c_k x_{ki}$ до розміщення елемента $c_k x_{ki-1}$, елемент e_{ki} матриці B приймає значення -1 ; при переміщенні $c_k x_{ki}$ до розміщення $c_k x_{ki+1}$ елемент e_{ki} приймає значення $+1$; інакше значення елемента e_{ki} залишається рівним 0 . Якщо наступні ітерації алгоритму потребують зсуву деякого елемента $-c_k x_{ki-1}$ матриці A на місце елемента $c_k x_{ki}$, і елемент $-e_{ki}$ матриці B отримує значення -1 , що також вимагає переміщення елемента $-c_k x_{ki+1}$ матриці A до розміщення елемента $c_k x_{ki}$, а елемент $-e_{ki}$ матриці B стає рівним $+1$, то ці переміщення не робляться і переходять до ряду $k + 1$ матриці A .

Крок 3. На цьому кроці перевіряється, який із трьох варіантів (оригінал та два отримані) дає найкращий результат за критерієм (1). Якщо зміщення було можливе лише вліво або лише вправо, то обидва варіанти, і початковий і отриманий, підлягають верифікації. Якщо на m -й ітерації виявиться, що $H^m = H^{m-1}$, то процес завершується і H^m приймається як наближений розв'язок для стовпця i . Відповідні зміни вносяться в матрицю A , матрицю B і вектор $H = \{H_i\}$ ($i = 1, 2, \dots, n$). Таким чином, i -й стовпець виключається з розгляду і переходимо до кроку 1 алгоритму. В іншому випадку виконується крок 4.

Крок 4. Переходимо до рядка $k + 1$ матриці A та кроку 2 алгоритму. Якщо всі рядки були переглянуті, то процес закінчується, значення H_i^k приймається як наближене рішення для задачі з розв'язаним стовпцем i і відповідні зміни вносяться в матрицю A , матрицю B і вектор $H = \{H_i\}$ ($i = 1, 2, \dots, n$). Таким чином, i -й стовпець виключається з розгляду і виконується крок 1 алгоритму.

Крок 5. Після перегляду всіх стовпців матриці A розв'язок задачі (1)-(4) завершується. Отримано нову розмітку для обробки запитів користувачів локального СЦ.

Вищенаведений алгоритм розв'язку задачі (1)-(4) обслуговування УПСУ в МРСЦ є скінченим завдяки кінцевій кількості СЦ в мережі.

Результати алгоритму перерозподілу завантаження між 9 вузлами МРСЦ для перших 5 ітерацій представлені на рис. 2. Номер графіка на діаграмі відповідає номеру ітерації. Горизонтальна вісь показує номери вузлів МРСЦ, вертикальна вісь відображає об'єм номінального потоку інформації, який слід обробити протягом регламентованого проміжку часу, для прийняття окремого рішення стосовно заданої проблеми.

Таким чином, алгоритм розв'язку задачі (1)-(4) оптимізації завантаження МРСЦ на основі методу булевого програмування дозволяє якісно підвищити рівень інформаційного обслуговування користувачів у МРСЦ і тим самим підвищити ефективність його функціонування. Цей приклад є прототипом моделі обслуговування УПСУ в МРСЦ на основі запитів експертів у цільових областях ситуаційного управління, що здійснюються в розподілених інформаційних середовищах.

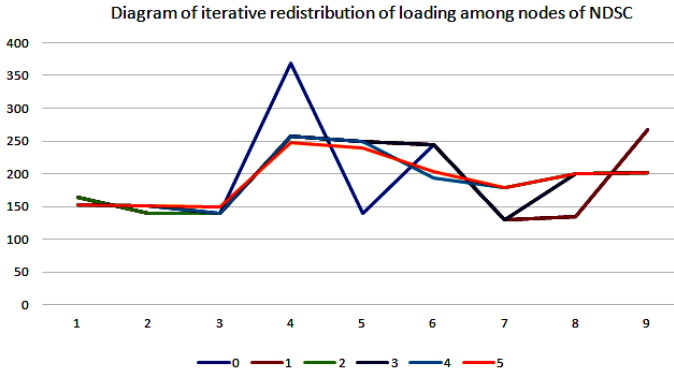


Рис. 2. Результати виконання 5 ітерацій перерозподілу завантаження для МРСЦ з 9 вузлами

Модель мінімізації часу обслуговування в МРСЦ. При здійсненні інформаційно-аналітичного процесу обробки інформації при завантаженні відповідного СЦ виникає задача управління послідовністю обробки інформаційних та керуючих потоків, завдяки чому комплекс СЦ забезпечує максимальну ефективність інформаційного, аналітичного та експертного обслуговування користувачів (УПСУ) з мінімальними витратами часу [7]. Така задача є двоїстою до прямої задачі, обговореної вище. Інформаційно-аналітичний процес ініціюється як запит, що визначає потребу в інформаційних ресурсах, аналітичних та експертних спроможностях, які надаються впродовж певного часу обслуговування в кожному локальному СЦ. Така задача формулюється наступним чином.

Розглянемо спільні розподілені інформаційні ресурси МРСЦ, які існують в n СЦ A_1, A_2, \dots, A_n , які можуть надавати різні інформаційні, аналітичні, експертні чи прогнозні сервіси. Система СЦ виконує деякий запит на використання інформаційного, аналітичного чи експертного сервісу для УПСУ. Запит вважається обслуговуваним, якщо він обслуговується кожним локальним СЦ послідовно, починаючи з СЦ A_1 . Припустимо, що час обслуговування запиту в i -му СЦ є незалежною випадковою величиною з функцією розподілу $F^i(t)$ та щільністю розподілу $\varphi_i(t) = F^i(t)$. Система працює так. Запит обслуговується в СЦ A_1 одразу після його надходження. На момент надходження запиту до СЦ A_2 можливі два варіанта:

- 1) запит був обслугований у СЦ A_1 раніше ніж у залученому до обслуговування СЦ A_2 і очікує відповіді від СЦ A_2 . За затримку обслуговування запиту накладається певний штраф;
- 2) запит був обслугований у СЦ A_1 пізніше ніж запит було надіслано до СЦ A_2 , тобто, СЦ A_2 чекає завершення обслуговування запиту в

СЦ A_1 . У такому випадку за затримку СЦ A_2 накладається штраф. В такому випадку здійснюється взаємодія між іншими СЦ.

Очевидно, що штраф f_i є випадковою величиною. Необхідно визначити такі моменти x_1, x_2, \dots, x_n підключення СЦ A_2, A_3, \dots, A_n , коли математичне сподівання загального штрафу, який дорівнює сумі накладених штрафів через невідповідність моментів x_i підключення i -го СЦ моментам η_{i-1} припинення обслуговування запиту в i -му СЦ, буде мінімальним.

Нехай α — питома вартість простою запиту, а β — питома вартість простою СЦ. Тоді штрафні санкції пов'язані з розбіжністю між моментами x_i підключення та моментами η_{i-1} завершення обслуговування запиту користувача в $(i - 1)$ -му СЦ визначається як:

$$f_1(\eta_1, x_2) = \begin{cases} \alpha(x_2 - \eta_1), \eta_1 \leq x_2, \\ \beta(\eta_1 - x_2), \eta_1 \geq x_2, \end{cases} \quad (5)$$

де $\eta_1 = \tau_1$,

$$f_2(\eta_2, x_3) = \begin{cases} \alpha(x_3 - \eta_2), \eta_2 \leq x_3, \\ \beta(\eta_2 - x_3), \eta_2 \geq x_3, \end{cases} \quad (6)$$

де $\eta_2 = \max(\eta_1, x_2) + \tau_2$,

$$f_{n-1}(\eta_{n-1}, x_n) = \begin{cases} \alpha(x_n - \eta_{n-1}), \eta_{n-1} \leq x_n, \\ \beta(\eta_{n-1} - x_n), \eta_{n-1} \geq x_n, \end{cases} \quad (7)$$

де $\eta_{n-1} = \max(\eta_{n-2}, x_{n-1}) + \tau_{n-1}$,

$$Mf_1(\eta_1, x_2) = (\alpha + \beta) \int_0^{x_2} F(u_1) du_1 + \beta(M_{\tau_1} - x_2), \quad (8)$$

$$Mf_2(\eta_2, x_3) = (\alpha + \beta) \int_{x_2}^{x_3} \left(\int_{x_2}^{u_3} F(u_1) \varphi_2(u_2 - u_1) du_1 + \beta(M_{\tau_1} - x_2) \right) du_2 + \\ + \beta \left(M_{\tau_1} + M_{\tau_2} + \int_0^{x_2} F(u_1) du_1 - x_3 \right), \quad (9)$$

$$Mf_3(\eta_3, x_4) = (\alpha + \beta) \int_{x_3}^{x_4} \left(\int_{x_3}^{u_3} \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) \varphi_3(u_3 - u_2) du_2 \right) du_3 + \\ + \beta \left(M_{\tau_1} + M_{\tau_2} + M_{\tau_3} + \int_0^{x_2} F(u_1) du_1 + \int_{x_2}^{x_3} \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) du_2 - x_4 \right), \quad (10)$$

$$\begin{aligned}
 Mf_{n-1}(\eta_{n-1}, x_n) = & (\alpha + \beta) \int_{x_{n-1}}^{x_n} \left(\dots \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) \dots \right) du_{n-1} + \\
 & + \beta \left(M_{\tau_1} + \dots + M_{\tau_n} + \int_0^{x_2} F(u_1) du_1 + \right. \\
 & \left. + \int_{x_{n-2}}^{x_{n-1}} \left(\dots \left(\int_{x_2}^{u_2} F(u_1) \varphi_2(u_2 - u_1) du_1 \right) \dots \right) du_{n-2} - x_n \right).
 \end{aligned} \tag{11}$$

Загальний штраф визначається за формулою

$$F(x) = \sum_{i=2} Mf_{i-1}(\eta_{i-1}, x_i). \tag{12}$$

Використання класичного підходу для мінімізації функції $F(x)$ вимагає обчислення множинних інтегралів і значень випадкового розподілу $F_i(t)$ за відповідними законами, що не завжди можливо. Однак ці труднощі можна усунути, використовуючи методи стохастичного програмування [14], з точки зору яких переформулюємо задачу. Задача, у новій постановці, полягає у пошуку мінімуму функції

$$F(x) = Mf(\eta, x) \tag{13}$$

за умов

$$x_i + 1 \geq x_i \geq 0, \quad i = 2, 3, \dots, n, \tag{14}$$

де

$$f(\eta, x) = \alpha \sum_{i \in I} (x_i - \eta_{i-1}) + \beta \sum_{j \in J} (\eta_{j-1} - x_j), \tag{15}$$

причому

$$\eta_i = \max(\eta_{i-1}, x_i) + \tau_i, \quad i = 2, 3, \dots, n.$$

Зуважимо, що функція (15) опукла відносно x_i для фіксованих η_{i-1} .

Задача у постановці (13)-(15) є окремим випадком задачі стохастичного програмування. Для її розв'язання може бути використаний стохастичний квазіградієнтний метод проекції [14], який описується процедурою

$$x_i^{s+1} = \pi_x(x^s - \rho_s W_i^s), \quad s = 0, 1, \dots, i; \quad i = 2, 3, \dots, n, \tag{16}$$

де s — номер ітерації; X — область проекції, що задовольняє обмеженням (14); $\pi_x(\bullet)$ оператор, що відображає кожену точку $x \in R^n$ у точку $\pi_x(x) \in X$ так, що $\|x - \pi_x(x)\|^2 \leq \|y - x\|^2$ для будь-якого $y \in X$; ρ_s коефіцієнт (множник) кроку; W_i^s випадковий вектор такий, що

$M(W_i^s | x_i^0, x_i^1, \dots, x_i^s) = F_x^A(x_i^s), s = 0, 1, \dots, i; i = 2, \dots, n$, де $F_x^A(x_i^s)$ — градієнт (узагальнений градієнт) функції $F(x)$. Послідовність $\{x_i^s\}, s = 0, 1, \dots, i$, визначена процедурою (16), збігається з ймовірністю 1 до розв'язку задач (13), (14).

Обчислювальна схема вирішення задачі формулюється наступним чином. Нехай x^s отримано на s -й ітерації (початкове значення x^0 задається). Потрібно зробити наступні кроки для отримання $(s + 1)$ -го наближення.

Крок 1. Спостерігаємо реалізацію x_i^s значення x_i .

Крок 2. Обчислюємо вектор W_i^s .

Крок 3. В результаті виконання операції проєкції на область $X = \{x_i : x_i \geq x_{i-1} \geq 0\}$, наступне наближення визначається як:

$$x_i^{s+1} = \max\left(0, \max\left(x_{i-1}^s; x_i^s - \rho_s W_i^s\right)\right), s = 0, 1, \dots, i; i = 2, 3, \dots, n.$$

Наведений вище алгоритм легко реалізувати і він не залежить від виду закону розподілу випадкового часу обслуговування запитів УПСУ.

Використання значення згладженого середнього

$$Z^{s+1} = Z^s + 1/W^s \left(f(\eta^s, x^s) - Z^s \right)$$

дає оцінку математичного сподівання випадкової функції $f(\eta^s, x^s)$.

Висновки. Запропоновані моделі дозволяють вирішити проблеми технічної інтероперабельності з точки зору забезпечення ефективного завантаження та мінімізації часу обслуговування запитів користувачів у розподілених інформаційних системах шляхом розв'язання прямої та двоїстої задач математичного програмування.

Для розв'язання прямої задачі обґрунтовано використання методу булевого програмування та запропоновано алгоритм його реалізації. Для розв'язання двоїстої задачі обґрунтовано використання методу стохастичного програмування та запропоновано алгоритм його реалізації. Алгоритм управління завантаженням СЦ, є інструментом управління МРСЦ, що забезпечує ефективне надання інформаційних, аналітичних та експертних послуг УПСУ з мінімізацією часу обробки інформації.

Запропоновані моделі підтримки технічної інтероперабельності та алгоритми управління інформаційно-аналітичним середовищем МРСЦ створюють підґрунтя для подальшого вдосконалення систем ситуаційного управління на основі оптимізації використання ресурсів і сервісів МРСЦ.

Список використаних джерел:

1. ISO/IEC 10746-2:2009. Information technology. Open distributed processing. URL: <https://www.iso.org/ru/standard/55723.html>.

2. ISO/IEC 24765:2017. Systems and Software Engineering. Vocabulary. URL:<https://www.iso.org/standard/71952.html>.
3. Коваленко О. Є. Композиційна конвергенція інформаційних технологій у системах ситуаційного управління на основі моделі знань предметної області. *Математичне моделювання в економіці*. 2019. Т. 16. №2. С. 39-44. DOI: 10.35350/2409-8876-2019-15-2-40-45.
4. New European Interoperability Framework (EIF). ISA². Interoperability solutions for public administrations, businesses and citizens. URL: https://ec.europa.eu/isa2/eif_en, https://ec.europa.eu/isa2/sites/isa/files/eif_brochure_final.pdf.
5. Глушков В. М. Введение в АСУ 2-е изд., испр. и доп. Киев : Техника, 1972. 320 с.
6. Морозов А. О., Косолапов В. Л., Отришко О. В. Засоби інформаційного сервісу об'єктів інтелектуальної власності. *Математичні машини і системи*. 2002. № 3. С. 136-147.
7. European Interoperability Reference Architecture (EIRA). ISA². Interoperability solutions for public administrations, businesses and citizens. URL: <https://joinup.ec.europa.eu/asset/eia/description>.
8. Koulou A., Zemzami M., El Hami N., Elmir A., Hmina N. Optimization in collaborative information systems for an enhanced interoperability network. *Int. J. Simul. Multidisci. Des. Optim.* 2020. Vol. 11, 2. DOI: 10.1051/smdo/2019021.
9. Leal G., Guedria W., Panetto H. A semi-automated system for interoperability assessment: an ontology-based approach. *Enterprise Information Systems*. 2020. Vol. 14 (3). P. 308-333. DOI: 10.1080/17517575.2019.1678767.
10. Guédria W., Naudet Y., Chen D. Maturity Model for Enterprise Interoperability. *Enterprise Information Systems*. 2015. Vol. 9 (1). P. 1-28. DOI: 10.1080/17517575.2013.805246.
11. Gabriel da Silva Serapião Leal, Wided Guédria, Hervé Panetto. Interoperability Assessment: A Systematic Literature Review. *Computers in Industry*. 2019. Vol. 106. P. 111-132. DOI:10.1016/j.compind.2019.01.002.
12. Khisro J., Sundberg H. Enterprise interoperability development in multi relation collaborations: Success factors from the Danish electricity market. *Enterprise Information Systems*. 2018. DOI: 10.1080/17517575.2018.1528633.
13. Забродский В. А., Скурихин В. И. Оптимизация функционирования АСУ предприятием. Киев; Донецк: Вища школа, 1978. 136 с.
14. Ермольев Ю. М. Методы стохастического програмування. М.: Наука, 1976. 239 с.

MODELS OF OPTIMIZATION OF LOADING OF THE NETWORK OF DISTRIBUTED SITUATIONAL CENTERS

The article substantiates the relevance of the creation and formulates a mathematical statement of the problem of technical interoperability maintenance in a network of distributed situational centers based on convergent technological solutions. Technical interoperability is considered in the context of optimizing the distribution of situational management tasks between individual situation centers in a distributed network and minimizing information processing time. For this purpose the model and algorithms of optimization of performance of tasks by participants of process of situational management in a network of the distributed situational centers are offered. The model of tech-

nical interoperability optimization is presented in the form of a zero-one integer programming problem with the search for the local extremum of the objective function. The information-analytical process is formed as a request that determines the need for information resources, analytical and expert capabilities and service time in each local situation center. The task of optimizing the load of the network of distributed situational centers is to redistribute the load between network nodes by equalizing the use of organizational, human, technical and information resources for timely and quality decision-making in situational centers that serve participants in situational management at different levels. The problem of minimizing service time in a network of distributed situational centers is dual to the primal problem of load redistribution and is solved using a stochastic quasigradient projection method. Algorithms for solving the formulated problems are developed and a test example of application is given. The potential for further improvement of situational systems of strategic planning management based on capability management for target systems is shown.

Keywords: *system on situational management, network of distributed situational centers, convergent system, technical interoperability, system optimization.*

Отримано: 16.09.2020

УДК 004.942:621.3.047.42

DOI: 10.32626/2308-5916.2020-21.113-125

О. О. Ситник, д-р техн. наук, професор,

К. М. Ключка, канд. техн. наук,

Г. О. Кисельова, старший викладач,

В. Б. Кисельов, старший викладач

Черкаський державний технологічний університет, м. Черкаси

МОДЕЛЮВАННЯ МІСТКОВОЇ ЕРОЗІЇ СЛАБКОСТРУМОВИХ ЕЛЕКТРИЧНИХ КОНТАКТІВ ЗАСОБАМИ MATLAB

У статті розглядається можливість побудови математичної моделі процесу механічної ерозії слабкострумових електричних контактних пар з використанням моделюючої системи MATLAB методом ідентифікації, тобто отримання математичної моделі реального об'єкта на основі експериментальних даних, представлених у вигляді алгебраїчних рівнянь.

Місткова ерозія електричних контактних пар і пов'язаний із нею перенос металу з однієї контактної поверхні на іншу суттєво впливає на зносостійкість електричних контактів реле, потенціометрів, потенціометричних датчиків, еncoderів та іншої слабкострумової комутаційної апаратури. Однак на розмір містка при ерозії контактів окрім величини струму впливає багато факторів, а саме: швидкість замикання та розмикання, сила притискання, температура, тиск та щільність навколишнього середовища, геометрія контактів, параметри контактних матеріалів

та інші процеси. Тому створення математичної моделі, яка б враховувала всі фактори, які впливають на місткову ерозію контактів є достатньо складною математичною задачею. Це призводить до широкого використання емпіричних залежностей у вигляді достатньо простих степеневих функцій. Проведене дослідження виявило, що похибка, яку дають моделі з використанням степеневих функцій, є досить високою і практично не відрізняється похибки моделей з використанням лінійних функцій. Тому для вибору оптимальних по вартості та зносостійкості контактних матеріалів, дослідники змушені використовувати результати отримані методом комутаційних випробувань контактних матеріалів. Однак в результаті проведених досліджень було виявлено, що для зменшення похибки та отримання більш адекватної моделі є доцільним застосувати квадратичну функцію, використання якої зменшує похибку приблизно вдвічі.

В результаті дослідження контактних пар з різних контактних матеріалів виявлено наступне: експоненціальна функція не може бути використана в якості моделі місткової ерозії; лінійна та степенева функції дають цілком прийнятні результати і можуть бути рекомендовані для використання на початкових етапах моделювання процесу місткової ерозії; більш адекватною математичною моделлю місткової ерозії слабкострумівих електричних контактних пар є квадратична функція.

Ключові слова: *контактні пари, математична модель, динаміка електричних процесів, регресійний аналіз, електрична ерозія.*

Вступ. Основними вузлами, що визначають ресурс комутаційного електричного обладнання є контакти. Вони здійснюють замикання, розмикання і перемикання електричних ланцюгів і є осередками переходу струму з однієї струмоведучої контакт-деталі в іншу [1]. Вид електричного розряду між контактами, а отже, різновид процесу електричного зносу контактів залежить від параметрів комутуваного ланцюга (напруги і струму). За величиною комутуваного струму контакти можуть класифікуватися на слабкоструміві (струми від часток до одиниць ампер), середньоструміві (струми від одиниць до сотень ампер), сильноструміві (струми від сотень до десятків тисяч ампер) [2]. Комутація контактами малопотужних ланцюгів з активним навантаженням супроводжується ерозійними явищами, що викликаються утворенням розплавлених містків в зоні контактування в певні моменти руху контактів. Визначення максимальних довжин містків має практичне значення не тільки для вивчення місткової ерозії, але і для оцінки швидкості замикання-розмикання та величини міжконтактних відстаней, необхідних для правильного вибору іскрогасних елементів [3-4].

Функціонування електричних контактних пар супроводжується зношуванням контактуючих матеріалів внаслідок тертя і електричної ерозії. Протягом останніх десятиліть проводилися трибологічні випробу-

вання, математичне моделювання та моделювання для вивчення природи і наслідків механічного зносу матеріалів з покриттям і без покриття [6-9]. Дослідження були переважно зосереджені на різних механізмах зношування: стирання, адгезії, ерозії, корозійному зносі, втомному зносі, а також на різних їх комбінаціях [7], однак вплив місткової ерозії на процес електричного контактування залишається ще недостатньо вивченим [9].

На основі проведеного аналізу літературних джерел було з'ясовано, що розбіжності результатів вимірювання довжин і діаметрів містків, отриманих різними дослідниками і різними методами, досить значні. Такі розбіжності можуть бути пов'язані як з методикою проведення експериментів, так і з підходами до опрацювання отриманих результатів. Станом на теперішній час методи обробки результатів експериментальних даних та подання їх у вигляді математичних моделей набули значного розвитку. Зокрема побудова моделей багатофакторних процесів, яким є місткова ерозія, де використання апріорної інформації, заснованої на законах фізики викликає певні труднощі, може бути проведено методами ідентифікації.

Метою статті є побудова математичної моделі місткової ерозії електричних контактних пар з використанням моделюючої системи MATLAB методом ідентифікації, тобто отримання за експериментальними даними математичної моделі реального об'єкта, представленої у вигляді алгебраїчних рівнянь (ідентифікацією, при побудові математичної моделі, будемо вважати визначення числових значень коефіцієнтів синтезованих моделей [10, 11]).

Основна частина. Відомо, що в слабкострумових електричних контактних парах переважає місткова ерозія. У роботі [3] автором проводилися визначення довжин містків по осцилограмам напруги на контактах для різних струмів і контактних матеріалів, стосовно до вимог малопотужних контактних систем, використовуваних в автоматичності. Вимірювання проводилися на поляризованих реле, що відрізняються між собою матеріалом контактів. Для проведення вимірювань при зниженому тиску повітря і різних газів, реле розміщувалися в герметизованих балонах, де створювалася необхідне середовище.

Максимальна довжина містка s , в момент підривання його струмом i , знаходиться зі співвідношення [3, 4]

$$s = v_{cp} \cdot \tau,$$

де v_{cp} — середня швидкість розмикання контактів поляризованих реле, яка визначається для кожного реле при заданих напрузі живлення і регулюванні, τ — відрізок часу від початку роз'єднання контактів до моменту вибуху містка, який визначається по осцилограмам напруги на містку.

Вимірами s при різних параметрах ланцюга було виявлено характерну властивість містків, а саме — відносна незалежність довжини

містка s від напруги живлення, індуктивності і ємності ланцюга в деяких (досить широких) межах зміни цих параметрів при незмінному струмі містка i . Експериментальне визначення максимальних довжин містків для різних контактних матеріалів, середовищ і струмів i представлено на рис. 1. Крім того, визначено, що максимальні довжини містків збільшуються зі збільшенням i , що мінімальним струмом, при якому з достатньою повторюваністю вдається виміряти s , є струм величиною 10 мА. Результати досліджень довжини містків ерозії (s , 10^{-4} см) від струму (i , мА) за даними [3] представлено в табл. 1.

Таблиця 1

Результати досліджень довжини містків ерозії
(s , 10^{-4} см) від струму (i , мА) за даними [3]

Сплав	Струм i , мА										
	10	100	200	300	400	500	600	700	800	900	1000
1 — ПЛН-10	0,5	1,14	1,52	1,87	2,21	2,56	2,9	3,23	3,56	3,88	4,21
2 — ПЛН-5	0,2	1,0	1,3	1,48	1,62	1,75	1,84	1,93	2,0	2,09	2,17
3 — ЗлН-5	0,15	0,84	1,06	1,19	1,28	1,34	1,41	1,46	1,52	1,57	1,63
4, 5, 6 — ПЛН-5	0,05	0,59	0,86	1,03	1,15	1,27	1,34	1,41	1,46	1,51	1,57
7 — Ср 999	0,2	0,35	0,45	0,53	0,62	0,69	0,75	0,81	0,86	0,91	0,97
8 — мідь	0,05	0,09	0,12	0,12	0,13	0,13	0,14	0,14	0,13	0,12	0,12

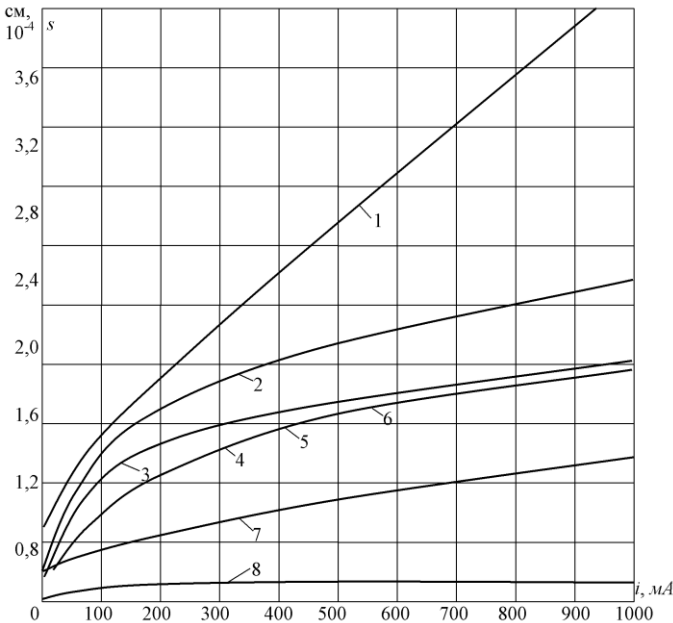


Рис. 1. Графік залежності довжини містка ерозії від величини комутованого струму для ряду контактних матеріалів [3]

Усереднені значення довжин містків при струмі до 1 А для різних контактних матеріалів: 1 — платино-іридієвий сплав з 10% іридію (ПЛІ-10); 2 — платино-нікелевий сплав з 5% нікелю (ПЛН-5); 3 — золото-нікелевий сплав з 5% нікелю; 4 — Паладій-Срібний сплав з 40% срібла (ПдСр-40); 5 — ПдСр-40 (анод) — ПЛН-5 (катод); 6 — пдср-40 (катод) — ПЛН-5 (анод); 7 — срібло; 8 — безкиснева мідь у водні.

Далі проведемо ідентифікацію моделі місткової ерозії методом регресійного аналізу [11]. Мінімізованою функцією похибки (різниця між прогнозованою моделлю і даними експерименту) при регресійному аналізі є сума квадратів похибок. Завдяки цьому вдається застосувати метод найменших квадратів [10]. Тобто знайдемо апроксимуючу функцію залежності величини містка ерозії від струму так, щоб функція $s(i)$ проходила максимально близько до всіх точок отриманих експериментальним шляхом.

Ключовий момент полягає у виборі класу наближених функцій. Основна вимога до цих функцій — це незалежність результатів відліку від початку відліку, тобто від зсуву по послідовності значень аргументу. Іншими словами, необхідно, щоб кінцева множина функцій вибраного для апроксимації класу переходила сама в себе при заміні i на $i + k$. Такими властивостями володіють:

- лінійні комбінації статичних функцій $1, i, i^2, \dots, i^n$;
- експоненціальні функції $e^{-a \cdot i}$;
- степеневі функції i^a .

Використання будь-якої іншої кінцевої множини функцій, крім перерахованих, вимагає наявності природного початку відліку, бо його вибір впливає на результат.

Оскільки при дослідженнях довжини містків ерозії від струму найбільш часто зустрічаються незмінні або повільно мінливі криві, то доцільно використовувати аналогічні залежності при виборі методів апроксимації:

- лінійна апроксимація

$$s(i) = a_0 + a_1 \cdot i; \quad (1)$$

- квадратична апроксимація

$$s(i) = a_0 + a_1 \cdot i + a_2 \cdot i^2; \quad (2)$$

- експоненціальна апроксимація

$$s(i) = A \cdot e^{a \cdot i}; \quad (3)$$

- степенева апроксимація

$$s(i) = A \cdot i^a. \quad (4)$$

Для обробки даних MATLAB використовує різні функції апроксимації даних поліномами за методом найменших квадратів — поліноміальної регресії. Однією з таких функцій є `polyfit(x, y, n)`,

яка повертає вектор коефіцієнтів полінома $p(x)$ степеня n , який з найменшою середньоквадратичною похибкою апроксимує функцію $y(x)$, результатом є вектор-рядок довжиною $n + 1$, що містить коефіцієнти полінома в порядку зменшення степенів [12].

Для проведення розрахунків в MATLAB місткової ерозії сплаву ПЛІ-10 вводимо вектори-стовбці експериментальних даних струму та довжини містка ерозії

```
>> i=[10 100 200 300 400 500 600 700 800 900
1000];
>> s=[0.5 1.14 1.52 1.87 2.21 2.56 2.9 3.23
3.56 3.88 4.21];
```

Для полінома першого степеня (лінійна апроксимація) в MATLAB

```
>> coeff1=polyfit(i,s,1)
coeff1 = 0.003562938870820 0.722564256525727
```

Для полінома другого степеня (квадратична апроксимація) в MATLAB

```
>> coeff2=polyfit(i,s,2)
coeff2 = -0.000000817889248 0.004385599991898
0.596754494347372
```

Подальше дослідження викликає певні труднощі, оскільки в MATLAB немає функцій експоненціальної та степеневі апроксимації.

Для проведення експоненціальної апроксимації в MATLAB прологарифмуємо праву і ліву частини рівняння (3). В результаті отримаємо

$$\ln(s) = \ln(A \cdot e^{a \cdot i}) \text{ або } \ln(s) = \ln(A) + a \cdot i.$$

Якщо ввести заміну $S = \ln(s)$, $B = \ln(A)$ то експоненціальна функція перетворюється на поліном першого порядку:

$$S = B + a \cdot i. \quad (5)$$

Врахувавши, що $A = e^B$ для експоненціальної апроксимації в MATLAB

```
>> S=log(s);
coeffe=polyfit(i,S,1);
coeffe=[coeffe(1) exp(coeffe(2))];
coeffe = 0.001772650510313 0.892458860657700
```

Для проведення степеневі апроксимації в MATLAB прологарифмуємо праву і ліву частини рівняння (4). В результаті отримаємо

$$\ln s = \ln(A \cdot i^a) \text{ або } \ln(s) = \ln(A) + a \cdot \ln i.$$

Якщо ввести заміну $S = \ln(s)$, $B = \ln(A)$, $I = \ln i$ то степенева функція перетворюється на поліном першого порядку:

$$S = B + a \cdot I. \quad (6)$$

Враховавши, що $A = e^B$ для експоненціальної апроксимації в MATLAB

```
>> S=log(s);
I=log(i);
coeffs=polyfit(I,S,1);
coeffs=[coeffs(1) exp(coeffs(2))]
coeffs = 0.464957479263563 0.148352348369164
```

Результати апроксимації зведені в табл. 2.

Середня похибка апроксимації Δs для кожної моделі відповідного контактного матеріалу розраховується за формулою

$$\Delta s = \frac{\sum_{j=1}^n |s_j - s'_j|}{n} \cdot 100\%, \quad (7)$$

де s'_j — розрахункове значення в точці i_j за рівнянням апроксимації;
 n — кількість точок апроксимації.

Таблиця 2

*Результати апроксимації дослідження залежності
довжини містків ерозії від струму*

Сплав	1 — ПлН10	2 — ПлСр40	3 — ЗлН5	4,5,6 — ПлН5	7 — Ср 999	8 — мідь
Лінійна регресія $s(i) = a_0 + a_1 \cdot i$ [10; 1000]						
a_1	0.0036	0.0016	0.0011	0.0013	0.00073	0.0000498
a_0	0.72	0.79	0.66	0.47	0.283	0.092
Δs %	3.42	12.88	13.53	19.15	5.15	13.58
Квадратична регресія $s(i) = a_0 + a_1 \cdot i + a_2 \cdot i^2$ [10; 1000]						
a_2	-0.00000082	-0.0000022	-0.00000176	-0.0000018	-0.00000042	-0.0000002
a_1	0.00439	0.00376	0.00289	0.003089	0.00115	0.0002555
a_0	0.59675	0.452	0.389	0.1958	0.218	0.0608
Δs %	2.45	7.46	8.65	8.41	1.88	4.5
Експоненціальна регресія $s(i) = A \cdot e^{a \cdot i}$ [10; 1000]						
a	0.00177	0.001525	0.00145	0.00212	0.00133	0.0005567
A	0.892	0.6437	0.5189	0.305	0.304	0.08586
Δs %	13.77	22.73	23.26	33.84	11.33	15.55
Степенева регресія $s(i) = A \cdot i^a$ [10; 1000]						
a	0.465	0.496	0.4898	0.717	0.35	0.2057
A	0.148	0.0788	0.0636	0.014	0.0786	0.0346
Δs %	8.05	9.09	11.56	16.88	6.2	9.21

Графіки відповідних функцій для Ср 999 представлено на рис. 2-5.

Середня похибка апроксимації ΔS для кожної моделі різних контактних матеріалів розраховується за формулою

$$\Delta S = \frac{\sum_{k=1}^m \Delta s_k}{m}, \quad (8)$$

де Δs_k — похибка апроксимації (7) для кожної моделі відповідного контактного матеріалу за рівнянням апроксимації; m — кількість досліджуваних функцій, $m = 6$.

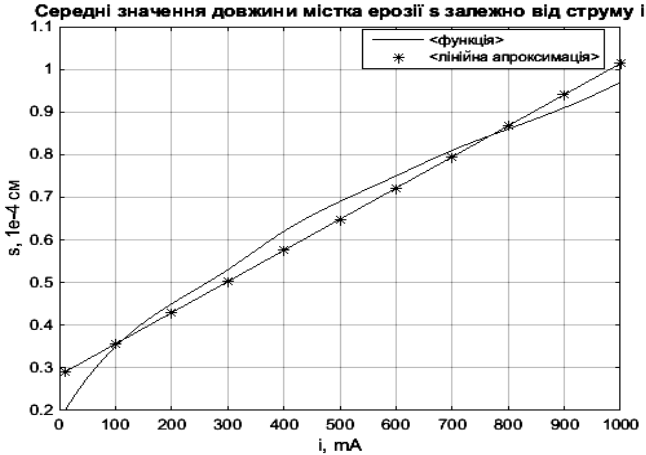


Рис. 2. Середні значення довжини містка ерозії $s(i)$ для сплаву Cr 999 за результатами експерименту та апроксимованої функцією $s(i) = a_0 + a_1 \cdot i$

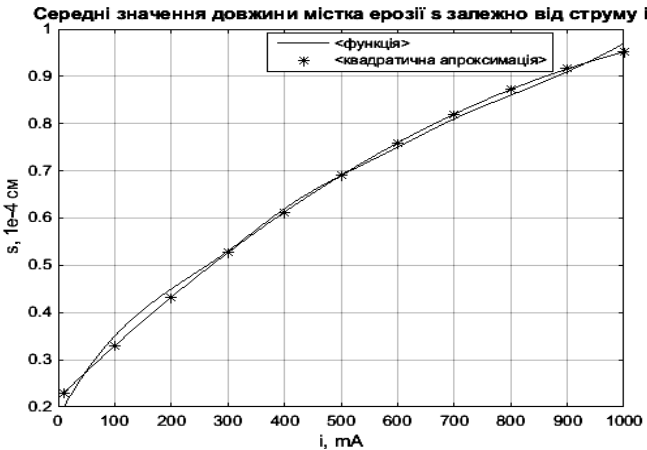


Рис. 3. Середні значення довжини містка ерозії $s(i)$ для сплаву Cr 999 за результатами експерименту та апроксимованої функцією $s(i) = a_0 + a_1 \cdot i + a_2 \cdot i^2$

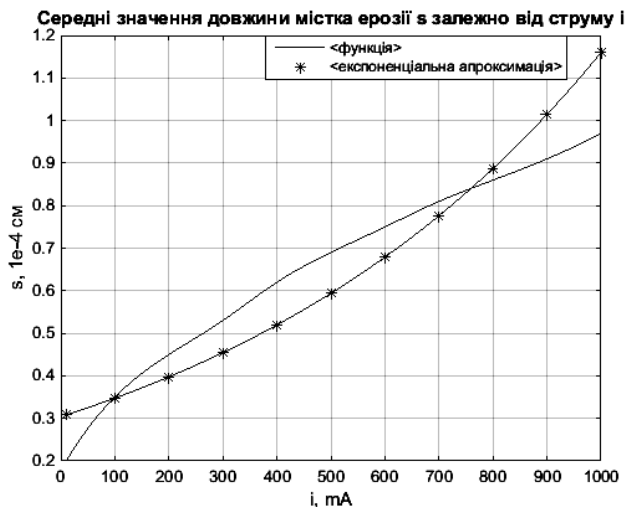


Рис. 4. Середні значення довжини містка ерозії $s(i)$ для сплаву Cr 999 за результатами експерименту та апроксимованої функцією $s(i) = A \cdot e^{a \cdot i}$

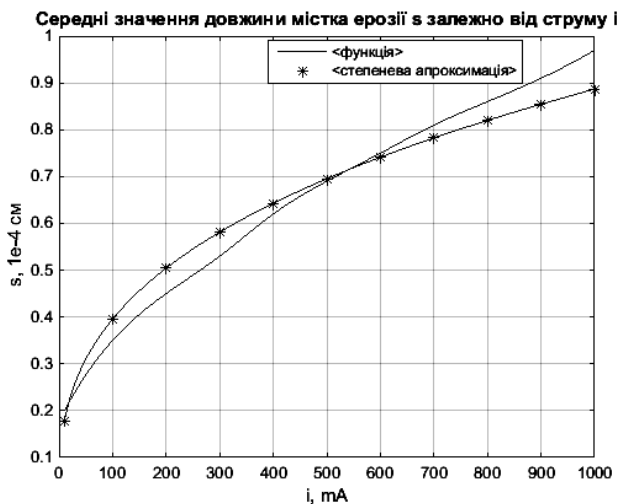


Рис. 5. Середні значення довжини містка ерозії $s(i)$ для сплаву Cr 999 за результатами експерименту та апроксимованої функцією $s(i) = A \cdot e^{a \cdot i}$

Середні похибки моделей місткової ерозії (8):

- лінійна модель

$$\Delta S = \frac{\sum_{k=1}^m \Delta s_k}{m} = \frac{3.42 + 12.88 + 13.53 + 19.15 + 5.15 + 13.58}{6} = 12.95\%;$$

- квадратична модель

$$\Delta S = \frac{\sum_{k=1}^m \Delta s_k}{m} = \frac{2.45 + 7.46 + 8.65 + 8.41 + 1.88 + 4.5}{6} = 5.56\%;$$

- експоненціальна модель

$$\Delta S = \frac{\sum_{k=1}^m \Delta s_k}{m} = \frac{13.77 + 22.73 + 23.26 + 33.84 + 11.33 + 15.55}{6} = 20.08\%;$$

- степенева модель

$$\Delta S = \frac{\sum_{k=1}^m \Delta s_k}{m} = \frac{8.05 + 9.09 + 11.56 + 16.88 + 6.2 + 9.21}{6} = 10.17\%.$$

Значення середньої похибки апроксимації від 7% до 15% свідчить про задовільно підібрану модель, а до 7% — добре підібрану модель [13].

Аналіз результатів визначення середньої похибки показав, що тільки апроксимація експоненціальною функцією дає похибку більшу за 15%, тому така модель не може бути рекомендована в якості моделі місткової ерозії контактів.

В результаті дослідження отримано моделі місткової ерозії:

- лінійна модель

$$s(i) = a_0 + a_1 \cdot i, \quad (9)$$

де $a_0 = (1 \div 8) \cdot 10^{-1}$, $a_1 = (5 \div 360) \cdot 10^{-5}$;

- квадратична модель

$$s(i) = a_0 + a_1 \cdot i + a_2 \cdot i^2, \quad (10)$$

де $a_0 = (1 \div 8) \cdot 10^{-1}$, $a_1 = (2,6 \div 44) \cdot 10^{-4}$, $a_2 = -(2,0 \div 22) \cdot 10^{-7}$;

- степенева модель

$$s(i) = A \cdot i^a, \quad (11)$$

де $A = (1,4 \div 15) \cdot 10^{-2}$, $a = (2,1 \div 7,1) \cdot 10^{-1}$.

Висновки. В результаті проведеного дослідження було отримано вирази (9)-(11) для математичних моделей місткової ерозії слабко-струмових електричних контактних пар у вигляді алгебраїчних рівнянь лінійної квадратичної та степеневі регресії та визначено їх похибки з використанням моделюючої системи MATLAB. Оскільки

середня похибка лінійної (12,95%) квадратичної (5,56%), та степеневі (10,17%) апроксимації не перевищує 15%, то отримані рівняння можуть бути використані в якості математичних моделей місткової ерозії контактів. Також необхідно відмітити, що квадратична функція більш якісною математичною моделлю місткової ерозії слабкострумних електричних контактних пар, оскільки її похибка практично вдвічі менша за похибки степеневі та лінійної моделей.

Список використаних джерел:

1. Григорьев А. А., Ваткина М. А. Анализ физических процессов износа электрических контактов коммутационных низковольтных аппаратов. *Вестник Чувашского государственного педагогического университета имени И. Я. Яковлева*. 2014. № 2 (82). С. 3-13.
2. Декабрун И. Е. Состояние инженерно-технических исследований в области слаботочных релейных контактов. *Электрические контакты*. М.: Наука, 1973. С. 15-19.
3. Разумихин М. А. Эрозионная устойчивость маломощных контактов М.: Энергия, 1964.
4. Holm R. Electric contacts: theory and application / by Ragnar Holm with Else Holm ; preface by J.B.P. Williamson. 4th ed. Berlin ; London: Springer, 2011. 484 p.
5. Montgomery S., Kennedy D., O'Dowd N. Analysis of Wear Models for Advanced Coated Materials. *Matrib: International Conference on Materials, Tribology, Recycling, Lipanj*, Croatia, June 24-26, 2009.
6. Мышкин Н. К., Кончиц В. В. Электрические контакты. Долгопрудный: Интеллект. 2008. 560 с.
7. Недорезов В. Г., Цыганков А. И. Влияние продуктов износа контактной пары потенциометра на надежность его работы. *Труды Международного симпозиума «Надежность и качество»*, в 2 т. Пенза: ПГУ, 2015. Т. 2. С. 153-154.
8. Мрачковський А. М. Дослідження електричної ерозії поверхонь різних контактних пар низьковольтних комутаційних апаратів. *Науковий вісник Національного університету біоресурсів і природокористування України. Серія: Техніка та енергетика АПК*. 2016. № 242. С. 155-158.
9. Мрачковський А. М. Дослідження закономірностей електричної ерозії дослідних зразків контакт — деталей на основі срібла і міді. *Енергетика і автоматика*. 2016. № 1. С. 82-89.
10. Мокін Б. І., Мокін В. Б., Мокін О. Б. Математичні методи ідентифікації динамічних систем : навчальний посібник. Вінниця: ВНТУ, 2010. 260 с.
11. Радченко С. Г. Статус математических моделей, получаемых с использованием регрессионного анализа. *Математичні машини і системи*. 2016. № 2. С. 138-147.
12. Лазарев Ю. Ф. Довідник з MATLAB : електронний навчальний посібник з курсового і дипломного проектування. Київ: НТУУ «КПІ», 2013. 132 с.
13. Петрянин Д. Л., Юрков Н. К., Романенко Ю. А. Повышение точности расчетов методов аппроксимации. *Труды международного симпозиума Надежность и качество*. 2015. № 1. С. 123-127.

MODELING OF CAPACITIVE EROSION OF LOW-CURRENT ELECTRICAL CONTACTS USING MATLAB

The article considers the possibility of constructing a mathematical model of the process of mechanical erosion of low-current electrical contact pairs using the MATLAB modeling system by identification, that is, obtaining a mathematical model of a real object based on experimental data presented in the form of algebraic equations.

Capacious erosion of electrical contact pairs and the associated transfer of metal from one contact surface to another significantly affects the wear resistance of electrical contacts of relays, potentiometers, potentiometric sensors, encoders, and other low-current switching equipment. However, the size of the bridge during contact erosion, in addition to the current value, is affected by many factors, namely: the speed of closing and opening, clamping force, temperature, pressure and density of the environment, contact geometry, contact material parameters, and other processes. Therefore, creating a mathematical model that would take into account all the factors that affect the capacious erosion of contacts is quite a complex mathematical task. This leads to a wide use of empirical dependencies in the form of fairly simple power functions. The study revealed that the error given by models using power functions is quite high and practically does not differ from the error of models using linear functions. Therefore, to select the optimal cost and wear resistance of contact materials, researchers are forced to use the results obtained by switching tests of contact materials. However, as a result of the research, it was found that in order to reduce the error and obtain a more adequate model, it is advisable to use a quadratic function, the use of which reduces the error by about half.

As a result of the study of contact pairs from different contact materials, the following was revealed: the exponential function cannot be used as a model of low-current electrical contact pairs; the linear and power functions give quite acceptable results and can be recommended for use at the initial stages of modeling the process of low-current electrical contact pairs; the quadratic function is a more adequate mathematical model of erosion of electrical contact pairs and the associated transfer of metal from one contact surface to another significantly affects the wear resistance of electrical contacts of relays, potentiometers, potentiometric sensors, encoders, and other low-current switching equipment. However, the size of the bridge during contact erosion, in addition to the current value, is affected by many factors, namely: the speed of closing and opening, clamping force, temperature, pressure and density of the environment, contact geometry, contact material parameters, and other processes. Therefore, creating a mathematical model that would take into account all the factors that affect the mystical erosion of contacts is quite a complex mathematical task. This leads to a wide use of empirical dependencies in the form of fairly simple power functions. The study revealed that the error given by models using power functions is quite high and practically does not differ from the error of models using linear functions. Therefore, to select the optimal cost

and wear resistance of contact materials, researchers are forced to use the results obtained by switching tests of contact materials. However, as a result of the research, it was found that in order to reduce the error and obtain a more adequate model, it is advisable to use a quadratic function, the use of which reduces the error by about half.

As a result of the study of contact pairs from different contact materials, the following was revealed: the exponential function cannot be used as a model of low-current electrical contact pairs; the linear and power functions give quite acceptable results and can be recommended for use at the initial stages of modeling the process of low-current electrical contact pairs; the quadratic function is a more adequate mathematical model of.

Key words: *contact pairs, mathematical model, dynamics of electrical processes, regression analysis, electrical erosion.*

Отримано: 17.09.2020

УДК 004.832

DOI: 10.32626/2308-5916.2020-21.125-139

С. І. Шаповалова, канд. техн. наук

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», м. Київ

ФОРМАЛІЗАЦІЯ ПРЕДСТАВЛЕННЯ ПРОДУКЦІЙНИХ ПРАВИЛ В ERLANG

У статті запропоновано формалізацію представлення в Erlang продукційної моделі представлення знань та відповідне представлення продукційних правил, умовна частина яких відповідає логіці першого порядку. Метою роботи є створення функції Erlang, яка не тільки представляє в базі знань правило висновування, а також виконує його активізацію при виклику.

Наведено відомості про існуючі реалізації логічного висновування в Erlang за обома підходами до формування міркувань: логічним — Erlog та продукційним: ERESYE, SERESYE та RUNES II. Обґрунтовано доцільність розроблення для Erlang власного механізму міркувань, що базується саме на властивостях цієї мови програмування.

Визначено базові принципи концепції логічного висновування за продукційною моделлю, орієнтовані на ефективне використання вбудованого механізму співставлення Erlang, для прискорення логічного висновування. В запропонованій формалізації кожна одиниця представлення має два визначення за синтаксисами логіки та Erlang. Формалізацію відповідно до рівню об'єктів представлення розподілено на три частини: визначення базових елементів логіки, представлення умов логічного висновування (зразків і фактів), представлення безпо-

середньо компонентів продукційної моделі (правила, робочої пам'яті, конфліктного набору).

На основі запропонованої формалізації продукційної моделі для Erlang розроблено функцію активації правила бази знань для ефективного логічного висновування. При виклику — у випадку успішної активації — ця функція повертає кортеж зі списку екземплярів поточного правила та показника його пріоритетності. Кожен елемент списку відповідає комбінації фактів робочої пам'яті, які були успішно узгоджені зі зразками умовної частини поточного правила. У випадку, коли жодного варіанту узгодження не існує, функція повертає кортеж з пустим списком.

Наведено приклад створення та застосування функції активації для поточного правила.

Ключові слова: *Erlang, Erlog, ERESYE, SERESYE, RUNES II, логічні міркування, експертні системи, механізм виведення.*

Сучасне програмне забезпечення неможливе без реалізації задач штучного інтелекту, зокрема послідовних міркувань. Це необхідно в багатьох прикладних галузях, насамперед, в системах діагностування та моніторингу, бортових системах керування, сервісах забезпечення інформації та послуг. Для вирішення задач логічного висновування використовується спеціальний програмний інструментарій, який містить механізм виведення заключень та надає можливість представляти правила висновування на декларативній мові. Найбільш розповсюдженими реалізаціями такого інструментарію є обгортки експертних систем та мови логічного програмування сімейства Prolog. Однак застосування такого інструментарію найбільш ефективно для створення автономних експертних систем. У багатьох випадках, наприклад, коли міркування необхідні для вирішення однієї з багатьох задач, що паралельно вирішуються, і при цьому тісно пов'язані з їх результатами, доцільно використовувати механізм логічного висновування, «вбудований» в базову мову програмного комплексу.

Erlang — одна з сучасних мов програмування, яка була створена для забезпечення телекомунікаційних систем, але в подальшому набула статусу мови загального призначення. Початкова специфіка мови Erlang зумовила властиві їй концептуальну відмовистість, здатність до паралельної обробки запитів в режимі реального часу, миттєвих операцій зі створення/знищення процесів та обміну повідомлень між ними. Завдяки цьому, застосування Erlang затребувано для реалізації паралельних розподілених систем, що здатні обслуговувати мільйони підключень.

Тенденція щодо інтелектуалізації всього сучасного програмного забезпечення зумовлює актуальність розроблення для Erlang власного механізму міркувань. Задача створення моделі представлення правил логічного висновування, що базується на декларативній та функціональній концепції Erlang, є актуальною і має практичну значущість.

Аналіз останніх досліджень. Для реалізації логічного висновування в Erlang використовувались обидва підходи до формування міркувань. За логічною моделлю було створено Erlog — інтерпретатор мови Prolog в Erlang; за продукційною — обгортки експертних систем: ERESYE (ERlang Expert SYstem Engine), SERESYE (Swarm oriented ERlang Expert SYstem Engine) та RUNES II. В таблицю 1 зведено відомості про ці реалізації. Останній стовпчик містить посилання як на публікації, так і на ресурси представлення програмного забезпечення. Відповідно вказується рік публікації або останнього оновлення програмного забезпечення.

Таблиця 1

Засоби реалізації логічного висновування в Erlang

№	Назва	Реалізація	Автори	Організація, в якій створено проєкт	Посилання
1	Erlog	Інтерпретатор мови Prolog, що інтегрується у вузол Erlang.	R. Virding	Erlang Solutions Ltd, Sweden	[1], останнє оновлення 2019
2	ERESYE	Бібліотека Erlang для розробки експертних систем та механізмів обробки правил	C. Santoro, A. Di Stefano, F. Gangemi	University of Catania, Italy; Erlang Training and Consulting, UK	[2], 2005; [3], останнє оновлення 2015
3	SERESYE	Бібліотека Erlang для розробки експертних систем та механізмів обробки правил	E. Mirrett, P. Nimreez, Yu. Rashkovskii	Afiniate, Inc., USA	[4], останнє оновлення 2017
4	RUNES II	Система висновування на правилах для хмарних обчислень	R. Zhou, G. Wang, J. Li, Jinghan Wang, Rui Zhou, Jing Li, and Guowei Wang	University of Science and Technology of China	[5], 2014

Всі зазначені засоби пропонують логічне висновування за класичними моделями, пристосовуючи для його реалізації вбудовані механізми Erlang. Так, всі обгортки розробки експертних систем [2-5], реалізують Rete алгоритм співставлення зі зразком. Алгоритми Rete [6] та TREAT [7] є базовими методами співставлення зразків antecedentів правил з фактами робочої пам'яті на етапі активації.

Саме цей етап є найбільш затратним за часом виконання. Не існує формального доведення оптимальності одного з них для поточної задачі [8]. За обома методами співставлення проводиться прекомпіляція бази знань, в результаті якої будується спеціальна графова структура представлення умов висновування. Співставлення на поточному кроці здійснюється як пошук на графі.

Однак Erlang має власний ефективний механізм співставлення, який фактично здійснює обмежену уніфікацію. Враховуючи специфіку Erlang щодо організації паралельних процесів та можливостей функціонального програмування, доцільно розробити орієнтовану на вбудований механізм співставлення модель представлення правил для пришвидшення їх активізації.

Мета і задачі. Метою роботи є створення функції, яка представляє в базі знань правило висновування і виконує його активізацію при виклику. Для виконання даної мети необхідно було вирішити такі завдання:

1. Визначити концептуальні вимоги до логічного висновування в Erlang.
2. Розробити формалізацію продукційної моделі для Erlang.
3. Створити функцію представлення правил в Erlang.
4. Реалізувати та навести приклад виконання запропонованої функції.

Концепція продукційної моделі орієнтована на ефективне використання вбудованого механізму співставлення Erlang та його властивостей функціональної мови програмування.

Концепція полягає в наступному:

1. Доведення істинності умови відбувається співставленням зі зразком: якщо зразок продукційного правила уніфікується з фактом, який знаходиться в робочій пам'яті WM (Working Memory), цей зразок вважається істинним.
2. Для співставлення використовується виключно вбудований механізм Erlang: умови логічного висновування (зразки, факти) обробляються безпосередньо і не потребують прекомпіляції, тобто не використовуються спеціальні алгоритми співставлення, такі як RETE та Treac, які потребують побудови додаткових графових структур зв'язків умов та їх аргументів.
3. Умовна частина кожного правила бази знань KB (Knowledge Base) представляється кон'юнктивною зв'язкою зразків: це відповідає приведенню антецеденту до диз'юнктивної нормальної форми і дозволяє представити кожен кон'юнкт окремою фразою Erlang.
4. Умовна частина — антецедент (antecedent) — та заключення — консеквент (consequent) — правила KB представляються в одній функції, виконання якої призводить до одного з двох результатів:

- 1) доведення істинності умов з одночасною уніфікацією їх змінних з відповідними аргументами консеквента,
 - 2) доведення хибності умов.
5. Передбачено можливість наявності в WM декількох фактів, які можна співставити з одним й тим самим зразком на поточному кроці висновування, що призводить до занесення в конфліктний набір декількох екземплярів (instantiation) одного й того самого правила KB.
 6. Модель містить базові критерії визначення пріоритету правила при розв'язанні конфлікту.

Формалізацію продукційної моделі в Erlang відповідно до рівню об'єктів представлення розподілено на три частини: визначення базових елементів логіки, представлення умов логічного висовування: зразків і фактів, представлення компонентів продукційної моделі: правила, робочої пам'яті, конфліктного набору. Формалізацію скомпоновано таким чином, щоб визначення кожного поточного об'єкту не містило раніш невизначених аргументів, тобто посилалось на попередні визначення та додавало нові властивості.

Erlang концептуально пов'язаний з мовою логічного програмування Prolog насамперед механізмом співставлення, синтаксисом та використанням змінних. Це дозволяє провести пряму аналогію між логічним поданням умов і правил висновування та їх представленням в Erlang. Тому в запропонованій формалізації кожна одиниця представлення має два визначення — в логіці (ліворуч) і в Erlang (праворуч) — та одне посилання формули. У визначеннях дотримано відповідні синтаксиси. Зокрема в Erlang позначення змінних починаються з великої літери, а констант — з маленької; квадратні дужки визначають список. Зауважимо, що фігурні дужки в логіці предикатів позначають множину, а в Erlang — кортеж. У такому сенсі ці позначення використовуються в подальшій формалізації.

Введемо **базові елементи логіки**, необхідні для формалізації моделі.

Константа може бути атомом або колекцією атомів. Будь-який інший вираз, аргументами якого є тільки константи, константою не являється. Це обмеження впливає з умови, що константа не має бути результатом виразу, що виконується в функціональному програмуванні:

$$\begin{array}{l} \text{atomic constant} \mid \\ \text{atomic constants} \\ \text{collection} \end{array} \mid \begin{array}{l} \text{atom} \mid \text{number} \mid \\ \text{list of atomic constants} \mid \\ \text{tuple of atomic constants} \end{array} \quad (1)$$

Обмеження: `constant ≠ expression from constants`

Оскільки визначення атомарної константи, атому, числа, списку та кортежу повністю відповідають своїм визначенням в логіці предикатів та Erlang, в даній моделі вони не наводяться.

Змінна може бути зв'язаною або незв'язаною:

$$\text{Variable} \equiv \begin{array}{l|l} \text{bound} \mid \text{unbound} & \text{constant} \mid V, \\ & \text{де } V \text{ — позначення змінної.} \end{array} \quad (2)$$

Позначення змінної призначається за синтаксисом Erlang.

Терм може бути константою або змінною:

$$\text{Term} \equiv \text{constant} \mid \text{Variable}, \quad (3)$$

де constant — константа (1); Variable — змінна (2).

Атомарна формула містить ім'я предикату та перелік аргументів — термів. Завдяки тій самій умові, що призвела до обмеження визначення константи, терм не може бути складеним.

$$\text{Atomic Formula} \equiv \begin{array}{l|l} \text{predicate}(X_1, X_2, \dots, X_n) & \{\text{predicate}, [X_1, X_2, \dots, X_n]\} \\ \text{де predicate — предикативний символ; } X_i \text{ — терм} & (3). \end{array} \quad (4)$$

Обмеження: term ≠ compound term

В Erlang атомарна формула представлена кортежем тому, що традиційний запис предикату в Erlang є функцією, що виконується, і не може бути застосований для представлення даних.

Літерал є позитивною або негативною атомарною формулою:

$$\text{Literal} \equiv \text{Atomic Formula} \mid \neg \text{Atomic Formula} = \begin{array}{l|l} \langle \neg \rangle \text{predicate}(X_1, X_2, \dots, X_n) & \{\text{predicate}, [X_1, X_2, \dots, X_n], \text{Sign}\} \end{array} \quad (5)$$

де Atomic Formula — атомарна формула (4); predicate — предикативний символ; X_i — терм (3);

$$\langle \neg \rangle \text{ — знак логічного заперечення, який використовується тільки в негативних формулах; } \begin{array}{l|l} \text{Sign — показник, що визначає} & \text{представлена атомарна формула} \\ & \text{або її заперечення:} \\ \text{Sign} \equiv \text{pos} \mid \text{neg} & (6) \end{array}$$

В логіці предикатів підстановкою θ є кінцева множина:

$$\theta = \{t_1/X_1, t_2/X_2, \dots, t_n/X_n\}, \quad (7)$$

де X_i — змінна, яка позначає i -й терм, при чому всі X_1, X_2, \dots, X_n — різні; t_i — значення i -го терму (константа).

Застосування підстановки θ до деякої формули означає заміну всіх змінних X_i на t_i . Позначимо через $\text{Literal}\theta$ частинний випадок літерала, отриманий після підстановки θ .

Приклад літерала після підстановки θ :

$$\text{Literal}\theta = \begin{array}{l|l} \langle \neg \rangle \text{predicate}(t_1, t_2, \dots, t_n) & \{\text{predicate}, [t_1, t_2, \dots, t_n], \text{Sign}\}, \end{array} \quad (8)$$

де predicate — предикативний символ; t_i — i -тий терм літерала після підстановки ($t_i = \text{constant}$); $\langle \rightarrow \mid \text{Sign} \rangle$ — наявність| відсутність заперечення (6).

Результат застосування множини підстановок $\{\theta_1, \dots, \theta_k\}$ до літерала позначимо $\text{Literal}\{\theta_i\}$:

$$\text{Literal}\{\theta_i\} = \left\{ \begin{array}{l} \text{Literal}\theta_1, \text{Literal}\theta_2, \dots, \\ \text{Literal}\theta_k \end{array} \mid \begin{array}{l} [\text{Literal}\theta_1, \text{Literal}\theta_2, \dots, \\ \text{Literal}\theta_k] \end{array} \right. \quad (9)$$

де $\text{Literal}\theta_j$ — приклад літерала після j -тої підстановки θ_j (8), $j=1..k$.

Останній результат стосується однієї з проблем реалізації логічного висновування за продукційною моделлю — необхідністю врахування декількох (більш ніж одного) прикладів підстановки для одного й того ж зразку. В разі існування декількох комбінацій фактів WM, які призводять до істинності умовної частини поточного продукційного правила, всі екземпляри цього правила заносяться до конфліктного набору.

Представлення умов логічного висновування. В продукційній моделі доведення істинності умов здійснюється на основі співставлення зі зразком. Кожна умова в правилі є зразком P (Pattern), який зіставляється з заданими або доведеними фактами робочої пам'яті. У випадку успішної уніфікації відповідних зразка та факту поточна умова вважається істинною. Структура зразків відповідає фактам WM. На відміну від фактів, аргументи зразків можуть мати незв'язані змінні.

Зразки представляються літералами з додатковим аргументом — позначкою часу доведення факту TimeStamp . В продукційному правилі цей аргумент є незв'язаною змінною. У випадку успішного співставлення на поточному кроці даного зразка з фактом робочої пам'яті, TimeStamp зв'язується відповідним значенням. Цей параметр є визначальним в стратегіях розв'язання конфлікту. Зразок визначається таким чином:

$$P \equiv \text{Literal} |_{\text{TimeStamp}} = \left\{ \begin{array}{l} \langle \rightarrow \rangle \text{predicate} (X_1, X_2, \dots, X_n) \\ \text{TimeStamp} \end{array} \mid \begin{array}{l} \{ \text{predicate}, [X_1, X_2, \dots, X_n], \\ \text{Sign}, \text{TimeStamp} \} \end{array} \right. \quad (10)$$

де Literal — літерал (9), predicate — предикативний символ; X_i — i -тий терм літерала (3), $\langle \rightarrow \mid \text{Sign} \rangle$ — наявність| відсутність заперечення (6), TimeStamp — незв'язана змінна показнику часу доведення факту, відповідного поточному зразку.

Факт F (Fact) є — прикладом літерала після підстановки θ , тобто факт містить тільки константи:

$$F \equiv \text{Literal}\theta = \begin{matrix} \langle \rightarrow \rangle \text{predicate}(t_1, \dots, t_n) | \\ \text{timeStamp} \end{matrix} \left| \begin{matrix} \{ \text{predicate}, [t_1, \dots, t_n], \\ \text{Sign}, \text{timeStamp} \}, \end{matrix} \right. \quad (11)$$

де $\text{Literal}\theta$ — приклад літерала після підстановки θ (8), predicate — предикативний символ; t_i — i -тий терм літерала (константа), $\langle \rightarrow \rangle$ | Sign — наявність| відсутність заперечення (6), TimeStamp — зв'язана змінна показнику часу доведення факту, відповідного поточному зразку.

Всі аргументи факту є зв'язаними.

В робочій пам'яті всі факти з одним й тим самим предикативним символом і структурою представляються записами формату (11), кожен з яких має унікальний набір аргументів. Таким чином представляється множина підстановок (9).

Одиницею представлення WM є факт (11).

Представлення продукційних правил. За загальноприйнятою продукційною моделлю правило представляє зв'язок передумов та заключення:

$$\text{Antecedent} \rightarrow \text{Consequent}, \quad (12)$$

де Antecedent — умовна частина; Consequent — частина дії.

Представлення антецеденту має відповідати логіці 1-го порядку.

Консеквент — перелік фактів, які доводяться або спростовуються за умови істинності антецеденту продукційного правила. Тобто кожен факт консеквенту — відповідно — додається або видаляється з WM . Випадки додавання доведених фактів найпоширеніші, тому такі факти в Erlang запропоновано представляти безпосередньо — без додаткових вказівок — за визначенням (11). Якщо факт має бути видаленим з робочої пам'яті, він представляється кортежем з тегом del .

Для виокремлення різниці між фактами робочої пам'яті і фактами консеквенту для останніх введемо позначення \hat{F} . Різниця полягає не тільки в наявності дії щодо зміни робочої пам'яті, але й в наявності в фактах консеквенту незв'язаних змінних, на відміну від фактів WM . Факт консеквенту у представленні правила є неосновним фактом. Безпосередньо фактом він стає після підстановки змінних, які визначені в зразках умовної частини і зв'язані на основі співставлення з фактами WM , відповідними умовам.

Тому у визначенні факту консеквенту використовується визначення зразку:

$$\hat{F} \equiv \begin{matrix} P^+ | P^- \\ \end{matrix} \left| \begin{matrix} P | \{ \text{del}, P \} \end{matrix} \right. \quad (13)$$

де P — зразок (10),

P^+ — факт, що додається, P^- — факт, що видаляється, $\{ \text{del}, P \}$ — факт, що видаляється.

Консеквент є множиною доведених/спростованих фактів:

$$\text{Consequent} \equiv \{ \dot{F}_1, \dot{F}_2, \dots, \dot{F}_w \} \quad | \quad [\dot{F}_1, \dot{F}_2, \dots, \dot{F}_w] \quad (14)$$

де F^*_{j} — j -й факт з визначенням дії щодо нього (13), $j=1..w$

Враховуючи декларативність Erlang та особливості його вбудованих механізмів, в запропонованій моделі для спрощення та прискорення процесу обробки продукційних правил, умовна частина кожного з них має бути представлена у диз'юнктивній нормальній формі DFN (disjunctive normal form). Тобто антецедент є диз'юнктивною зв'язкою кон'юнктив:

$$\text{Antecedent} = \text{Conjunction}_1 \vee \text{Conjunction}_2 \vee \dots \vee \text{Conjunction}_d \quad (15)$$

де $\text{Conjunction } i$ — i -й кон'юнкт, $i=1..d$.

Кон'юнкт в моделі є кон'юнктивною зв'язкою зразків:

$$\text{Conjunction} \equiv \{ P_1 \wedge P_2 \wedge \dots \wedge P_m \} \quad | \quad [P_1, P_2, \dots, P_m] \quad (16)$$

де P_j — j -й зразок (13) кон'юнкту, $j=1..m$.

Таким чином, одне продукційне правило представляється декількома клозами — фразами Erlang, кількість яких дорівнює кількості кон'юнктив:

$$\text{Production} \equiv \begin{array}{l} (\text{Conjunction}_1 \rightarrow \text{Consequent}_1) \vee \\ (\text{Conjunction}_2 \rightarrow \text{Consequent}_2) \vee \\ \dots \\ \vee (\text{Conjunction}_d \rightarrow \text{Consequent}_d) \end{array} \quad | \quad \begin{array}{l} \text{Clause}_1 ; \\ \text{Clause}_2 ; \\ : \\ \text{Clause}_d. \end{array} \quad (17)$$

де

Conjunction_i — i -й кон'юнкт (16), $i=1..d$; Consequent_i — консеквент (14), який відповідає i -му кон'юнкту; Clause_i — фаза Erlang, яка відповідає i -й зв'язці умов та заключення, $i=1..d$.

Обмеження: Antecedent \equiv logic connective in DFN

В стратегіях розв'язання конфлікту, що базуються на сортуванні активацій за пріоритетами, окрім критеріїв новизни та специфічності, використовують значущість Saliency. Цей параметр дозволяє користувачеві призначати правилам пріоритет.

Таким чином, кожен клоз бази знань має містити таку інформацію:

$$\{ \text{Conjunction}_n, \text{Consequent}_n, \text{Saliency}_n \}, \quad (18)$$

де n — номер правила KB, Conjunction_n — кон'юнкт (16), Consequent_n — консеквент (14), Saliency_n — показник пріоритетності n -го правила.

Кожен клоз може оброблятися автономно.

Функція представлення та активації продукційного правила. Уніфікація змінних логічних умов продукційного правила є основною проблемою реалізації продукційної моделі засобами Erlang.

Оскільки умовна частина кожного правила KB є кон'юнктивною зв'язкою, доведення істинності відбувається за умови успішної уніфікації кожного з представлених зразків з фактами робочої пам'яті.

Уніфікація полягає в пошуку в робочій пам'яті факту з тим самим предикативним символом та здійсненні можливої підстановки множини аргументів. Кожен аргумент зразку умовної частини уніфікується з відповідним аргументом факту WM. Уніфікація успішна, якщо:

- 1) відповідні аргументи — однакові константи;
- 2) аргумент зразку є незв'язною змінною, а аргумент факту — константою.

В останньому випадку всі однойменні змінні антецеденту та консеквенту поточної фрази мають бути зв'язаними цією константою.

Проблема полягає в тому, що вбудований механізм співставлення Erlang, на відміну від Prolog, реалізує лише часткову уніфікацію. В даному випадку це означає, що неможливо безпосередньо представити в правилі зразки з незв'язаними змінними, які заздалегідь не визначаються в тому самому клозі Erlang або при виклику функції, якій він належить. Тому кортеж (18) представити безпосередньо неможливо. Для розв'язання цієї проблеми запропоновано спеціальну структуру, яка базується на використанні генераторів списку. Як спеціальний засіб функціонального програмування, генератор списку дозволяє задавати незв'язані змінні, які згодом будуть уніфіковані на основі співставлення з елементами відомого списку.

При заповненні бази знань програміст визначає всі зразки та факти поточного правила та незв'язані змінні в них. Якщо ці змінні вказують на один й той самий об'єкт, їм призначаються однакові позначення. Для їх зв'язування достатньо помістити обидві частини правила в один вираз, який задасть формат генерації відповіді при активації правила.

Окрім цього у відповідь має входити структура Trace, яка однозначно представить екземпляр правила для занесення в **конфліктний набір** Agenda. Така структура представляється кортежем, який містить номер правила в KB та перелік всіх позначень незв'язаних змінних, які є в Conjunction і Consequent.

Таким чином, при активації правила генерується список відповідей зі структурою:

$$\{ \text{Trace}_n, \text{Conjunction}_n, \text{Consequent}_n \}, \quad (19)$$

де n — номер правила KB, Trace_n — «слід» правила, Conjunction_n — кон'юнкт (16), Consequent_n — консеквент (14).

В кортежі (19) не враховано показник пріоритетності правила $Saliencen$, оскільки цей параметр однаковий для всіх екземплярів правила і доцільніше його повернути в загальній структурі рішення.

При реалізації частини генерації необхідно представити послідовну уніфікацію кожного зразка *Conjunction* з відповідними фактами *WM*. Тобто має бути застосовано m генераторів — за кількістю зразків *Conjunction*. Однак область дії змінної обмежується тим генератором, де вона представлена. Тому при застосуванні декількох генераторів було використано спеціальні умови зв'язування *VarsBinding* (*Variables Binding Conditions*) на основі оператору « $=$ ». Умови *VarsBinding* застосовуються, починаючи з 2-го генератора.

Окрім безпосереднього подання зразків в умовні частині за виразом (10), в модель закладено додаткову можливість здійснення порівнянь значень змінних. Це реалізовано додаванням умов генерації у форматі *Guard* — охоронного виразу, який містить зв'язані та незв'язані змінні поточного зразку P_i .

Назва і позначення охоронного виразу *Guard* запозичені в Erlang. Охоронний вираз — це логічний вираз, який представляє додаткове обмеження в голові функціональних рівнянь, *if*-, *case*- та *receive*-виразах. В запропонованій моделі зі всіх різновидів охоронних виразів Erlang використовуються лише операції порівняння. Для виконання цих операцій використовується безпосередньо механізм Erlang, тому:

- 1) збережено синтаксис Erlang;
- 2) відсутні обмеження на вирази, що порівнюються;
- 3) припускається використовувати логічні зв'язки охоронних виразів.

Таким чином, кожен $Guard_i$ є операцією порівняння або логічною зв'язкою таких операцій.

Вираз *Guard* не є обов'язковим, тому його назва в наведеній нижче функції (20) міститься в символах $\langle \rangle$.

Аргументами цієї функції є номер правила KB, позначка часу активації поточного правила (може вказуватись номер поточного циклу вивчення) та список істинних фактів — *WM*. Номер правила фактично дублюється в «сліді» *Trace*, однак його виокремлено в аргументи функції з міркувань надання можливостей розпаралелювання обробки бази знань. Позначка часу активації поточного правила у разі доведення істинності умов уніфікується з усіма незв'язаними змінними позначок часу доведення фактів консеквенту. Таким чином забезпечується додавання в *WM* фактів з актуальним часом доведення.

Функція, яка представляє фразу KB, має такий формат:

```

Clausen ≡ (20)
rule(N, CurTimeStamp, WM) ->
{
  [ {
    Tracen,          %% {N, UnboundVarsSet}
    Conjunctionn,    %% [ P1, P2, ..., Pm ]
    Consequentn      %% [ F1, F2, ..., Fw ]
    } || P1 <- WM, <Guard>,
      P2 <- WM, VarsBinding2, <Guard>,
      ⋮
      Pm <- WM, VarsBindingm, <Guard>
  ],
  Saliencen
};

```

де N — номер правила KB, $CurTimeStamp$ — показник часу доведення фактів $Consequent_n$, WM — робоча пам'ять, $Trace_n$ — «слід» правила, $Conjunction_n$ — кон'юнкт (16), $Consequent_n$ — консеквент (14), P_i — i -та логічна умова — зразок (10), F_j — j -й факт $Consequent$ з визначенням дії щодо нього (13), $VarsBinding_i$ — умови зв'язування змінних i -го та всіх попередніх генераторів, $Guard$ — умова обмеження аргументів зразків, $Salience_n$ — показник пріоритетності n -го правила.

При виклику ця функція повертає кортеж з двох аргументів: списку рішень у форматі (19) та показника пріоритетності поточного правила. Кожен елемент списку відповідає комбінації фактів робочої пам'яті, які були успішно узгоджені зі зразками умовної частини поточного правила. У випадку, коли жодного варіанту узгодження не існує, функція повертає кортеж з пустим списком.

Таким чином, кортеж (18) перетворено в вираз Erlang, який автоматично виконується при виклику.

Приклад активації правила. Для тестування реалізації запропонованої функції активації (20) створимо правило:

$$f_1(1, X) \wedge f_2(X, Y, 2) \wedge Y > 0 \wedge \neg (f_3(Y, 3, X)) \rightarrow (21) \\ f_{21}(1, X) \wedge f_{22}(X, Y, 2) \wedge \neg (f_{23}(Y, 3, X)),$$

де f_i — зразки.

Це правило містить позитивні $f_1/2$, $f_2/3$ та негативну $f_3/3$ атомарні формули та додаткову умову, що обмежує незв'язану змінну Y .

Правило (21) згідно з функцією (20) відповідає наступний запис KB:


```

rule(1, T, WM) -> (22)
{
[
{
{1, X, Y},
[
{f1, {1, X}, pos, T1},      {f2, {X, Y, 2}, pos, T2},
%% Conjunction,
{f3, {Y, 3, X}, neg, T3}],
[
{f21, {1, X}, pos, T},      {f22, {X, Y, 2}, pos, T},
%% Consequent }
{f23, {Y, 3, X}, neg, T}]
}
||
{f1, {1, X}, pos, T1} <- WM,
{f2, {X1, Y, 2}, pos, T2} <- WM, X1 == X,
Y>0,
{f3, {Y1, 3, X2}, neg, T3} <- WM, Y1 == Y,
X2 == X1],
1}.
%% Saliense

```

Для відображення можливості отримання декількох екземплярів цього правила для кожного зі зразків умовної частини $f_1/2$, $f_2/3$, $f_3/3$ в робочій пам'яті представимо по 3 підстановки:

```

WM = (23)
{ f1(1, x1) |2,      [ {f1, {1, x1}, pos, 2},
  f2(x1, y1, 2) |1,    {f2, {x1, y1, 2}, pos, 1},
 -f3(y1, 3, x1) |0,    {f3, {y1, 3, x1}, neg, 0},
f1(1, x2) |11,        {f1, {1, x2}, pos, 11},
f2(x2, y1, 2) |21,    {f2, {x2, y1, 2}, pos, 21},
-f3(y1, 3, x2) |13,    {f3, {y1, 3, x2}, neg, 13},
  f1(2, x2) |2,        {f1, {2, x2}, pos, 2},
  f2(x2, y2, 2) |2,    {f2, {x2, y2, 2}, pos, 2},
 -f3(y2, 3, x1) |2 }  {f3, {y2, 3, x1}, neg, 2} ]

```

Очевидно, що тільки 2 комбінації фактів WM забезпечують істинність кон'юнктивної зв'язки.

На рисунку 1 приведено результат виконання функції (22) на основі фактів робочої пам'яті (23). Отриманий в середовищі Erlang результат доповнено помітками, які його структурують.

Відповідь містить опис двох екземплярів правила з номером 1, а також значущість цього правила (значення 1 зазвичай встановлюється за замовчуванням). Екземпляри правила відповідають підстановкам: $\{x_1/X, y_2/Y\}$ та $\{x_2/X, y_1/Y\}$.

Отриманий результат містить повну інформацію для доповнення конфліктного набору та виконання на поточному кроці логічного висновування заключних етапів: розв'язання конфлікту та запуску обраного екземпляру правила.

```

3> r:test().
{[{{1,x1,y1},                                     { [ { Trace1
  [{f1,{1,x1},pos,2},                             ---
  {f2,{x1,y1,2},pos,1},                           | Conjunction1
  {f3,{y1,3,x1},neg,0}],                          ---
  [{f11,{1,x1},pos,0},                             ---
  {f12,{x1,y1,2},pos,0},                           | Consequent1 }
  {f13,{y1,3,x1},neg,0}]]},                      ---
  {{1,x2,y1},                                       { Trace2
  [{f1,{1,x2},pos,11},                             ---
  {f2,{x2,y1,2},pos,21},                           | Conjunction2
  {f3,{y1,3,x2},neg,13}],                          ---
  [{f11,{1,x2},pos,0},                             ---
  {f12,{x2,y1,2},pos,0},                           | Consequent2 } ]
  {f13,{y1,3,x2},neg,0}]]}],                    ---
  1}                                               Saliense }

```

Рис. 1. Приклад результату виконання функції активації правила

Таким чином, використання запропонованої функції (20) дозволяє ефективно здійснити етап активації — самий ресурсоємний за витратами часу та пам'яті.

Висновки:

1. Визначено базові принципи концепції логічного висновування за продукційною моделлю, орієнтовані на ефективне використання вбудованого механізму співставлення Erlang та його властивості функціональної мови програмування.
2. Запропоновано формалізацію продукційної моделі для Erlang на основі його вбудованого механізму співставлення зі зразком для прискорення логічного висновування.
3. Запропоновано функцію активації правила бази знань на основі формалізації продукційної моделі для Erlang для ефективного логічного висновування.
4. Наведено приклад створення та застосування функції активації для поточного правила.

Перспективою подальших досліджень є створення механізму висновування на основі запропонованої моделі як додаткового засобу Erlang.

Список використаних джерел:

1. Viriding R. Erlog — Prolog for an Erlang Application. URL: <https://github.com/rviriding/erlog>.
2. Di Stefano A., Gangemi F., Santoro C. Eresye: Artificial intelligence in Erlang programs. *Proc. 2005 ACM SIGPLAN Work.* Erlang — ERLANG '05, Sept. 26-28, 2005. Tallinn, 2005. P. 62-71. DOI:10.1145/1088361.1088373.
3. Santoro C. ERESYE download. URL: <https://sourceforge.net/projects/eresye>.

4. SERESYE — Swarm oriented Erlang Expert System Engine. URL: <https://github.com/afiniate/seresyte>.
5. Zhou R., Wang G., Li J. RUNES II: A Distributed Rule Engine Based on Rete network in Cloud Computing. *International journal of Grid distribution computing*. 2014. Vol. 7. № 6. P. 91-110. DOI:10.14257/ijgcd.2014.7.6.08.
6. Forgy C.L. Rete: A Fast Algorithm for the Many Pattern. *Many Object Pattern Match Problem, Artificial intelligence*. 1982. Vol. 19. № 1. P. 17-37.
7. Miranker D.P. TREAT: A New and Efficient Match Algorithm for AI Production Systems. *Morgan Kaufmann Publishers*. San Francisco, 1990. P. 143.
8. Шаповалова С. І., Мажара О. О. Вибір оптимального алгоритму співставлення зі зразком при проектуванні продукційної системи. *Східно-Європейський журнал передових технологій*. 2014. Вип. 2/2 (68). С. 43-49. DOI: 10.15587/1729-4061.2015.46571.

FORMALIZATION OF THE RULES OF INFERENCE IN ERLANG

The article proposes a method of solving logical puzzles on the basis of machine learning. The method is designed for the preliminary formalization of tasks in the form of description of properties and relations between them. Because each property has a set of possible values, the solution of the puzzle by the methods of search has a combinatorial complexity. With a large number of properties and their values, the time of the solving is rapidly increasing.

In recent years, a separate area of research in machine learning has been the solution to logical tasks of this type. However, existing solutions to this area have a number of shortcomings, first and foremost, they do not always guarantee a correct solution.

The paper presents a special network of connections for learning the solution of logical puzzles, as well as their formalization for the representation of this network. The network contains computing nodes that represent the relationship between properties, and the nodes of the input layers that specify the values of these relationships.

Every task is solved by automatically creating a network of links with its further training until the solution is obtained. The geometric interpretation of the n -dimensional network of bonds and its $(n - 1)$ -dimensional layers is given. The formalization of the presentation of the study sample and the learning algorithm are presented. The mechanism of solving logical combinatorial problems is presented.

The article presents examples of tasks that are traditional tests in systems of logical programming and production (expert) systems, as well as tasks from the resource bAbI of such classes: two supporting facts, two argument relations, positional reasoning.

The efficiency of the proposed method has been experimentally proved.

The prospects of further researches, which are connected with the creation of a lexical-syntactic analyzer for automatic representation of properties, their values and relations between them, are determined.

The proposed method is universal and does not depend on the characteristics of the current task, such as the number of properties and their values.

Key words: *Erlang, Erlog, ERESYE, SERESYE, RUNES II, logical reasoning, rule-based systems, inference engine.*

Отримано: 21.05.2020

ВІДОМОСТІ ПРО АВТОРІВ

Барановський Сергій Віталійович — кандидат технічних наук, доцент, доцент кафедри економічної кібернетики Рівненського державного гуманітарного університету, м. Рівне, svbaranovsky@gmail.com

Бомба Андрій Ярославович — доктор технічних наук, професор, професор кафедри комп'ютерних наук та прикладної математики Національного університету водного господарства та природокористування, м. Рівне, abomba@ukr.net

Верлань Анатолій Федорович — доктор технічних наук, професор, член кореспондент НАПН України, Інститут проблем моделювання в енергетиці імені Г. Є. Пухова НАН України, м. Київ, a.f.verlan@gmail.com

Verlan Andriy — PhD, Associate Professor, Norwegian University of Science and Technology, Gjøvik, Norway, verlandr@gmail.com

Гордєєв Олександр Олександрович — кандидат технічних наук, доцент, завідувач кафедри кібербезпеки Університету банківської справи, м. Київ, Alex.gordeyev@gmail.com

Дячук Олександр Анатолійович — кандидат технічних наук, старший науковий співробітник, провідний науковий співробітник Державної установи «Інститут економіки та прогнозування НАН України», м. Київ, oadyachuk@ukr.net

Іванчук Наталія Віталіївна — кандидат технічних наук, старший викладач кафедри комп'ютерних наук та прикладної математики Національного університету водного господарства та природокористування, м. Рівне, n.v.medvid@nuwm.edu.ua

Кисельов Владлен Борисович — старший викладач кафедри електротехнічних систем Черкаського державного технологічного університету, м. Черкаси, vladkis.777@gmail.com

Кисельова Ганна Олексіївна — старший викладач кафедри електротехнічних систем Черкаського державного технологічного університету, м. Черкаси, annakys.777@gmail.com

Климук Юрій Євгенійович — кандидат технічних наук, доцент, доцент кафедри комп'ютерних наук та прикладної математики Національного університету водного господарства і природокористування, м. Рівне, klimyuk@ukr.net

Ключка Костянтин Миколайович — кандидат технічних наук, доцент, доцент кафедри електротехнічних систем Черкаського державного технологічного університету, м. Черкаси, ux0cx@ukr.net

Коваленко Олексій Єпифанович — кандидат технічних наук, доцент, старший науковий співробітник Інституту проблем математичних машин і систем НАН України, м. Київ, Kovalenko.O.E@nas.gov.ua

Косолапов Володимир Леонідович — кандидат технічних наук, старший науковий співробітник, провідний науковий співробітник Інституту проблем математичних машин і систем НАН України, м. Київ, KosolapovVL1953@gmail.com

Леонтієв Костянтин Петрович — технічний директор науково-виробничого підприємства «Радій», м. Кропивницький

Мартинюк Петро Миколайович — доктор технічних наук, професор, завідувач кафедри комп'ютерних наук та прикладної математики Національного університету водного господарства та природокористування, м. Рівне, p.m.martyniuk@nuwm.edu.ua

Ситник Олександр Олексійович — доктор технічних наук, професор, завідувач кафедри електротехнічних систем Черкаського державного технологічного університету, м. Черкаси, sytnyk.ets@gmail.com

Sterten Jo — PhD, Assistant Professor, Norwegian University of Science and Technology, Gjøvik, Norway

Филипчук Віктор Леонідович — доктор технічних наук, професор, завідувач кафедри охорони праці та безпеки життєдіяльності Національного університету водного господарства та природокористування, м. Рівне, v.l.fylupchuk@nuwm.edu.ua

Фуртат Юрій Олегович — кандидат технічних наук, старший науковий співробітник Інституту проблем моделювання в енергетиці імені Г. Є. Пухова НАН України, м. Київ, saodhar@ipme.kiev.ua

Чмир Ігор Олексійович — доктор технічних наук, професор, професор кафедри інформатика Одеського державного екологічного університету, м. Одеса, chimirigor@gmail.com

Шаповалова Світлана Ігорівна — кандидат технічних наук, доцент, доцент Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», м. Київ, lanashape@gmail.com

АЛФАВІТНИЙ ПОКАЖЧИК АВТОРІВ

Б		Л	
Барановський С. В.	5	Леонтьєв К. П.	51
Бомба А. Я.	5, 83		
В		М	
Верлань А. Ф.	25	Мартинюк П. М.	76
Verlan Andriy	43		
Г		С	
Гордєєв О. О.	51	Ситник О. О.	113
		Sterten Jo	43
Д		Ф	
Дячук О. А.	61	Филипчук В. Л.	76
		Фуртат Ю. О.	61
І		Ч	
Іванчук Н. В.	76	Чмир І. О.	25
К		Ш	
Кисельов В. Б.	113	Шаповалова С. І.	125
Кисельова Г. О.	113		
Климюк Ю. Є.	83		
Ключка К. М.	113		
Коваленко О. Є.	102		
Косолапов В. Л.	102		

ЗМІСТ

Барановський С. В., Бомба А. Я. Узагальнення математичної моделі противірусної імунної відповіді Марчука-Петрова з урахуванням впливу малих просторово розподілених дифузійних збурень	5
Верлань А. Ф., Чмырь И. А. Две модели вопросно-ответной «машины диалога»	25
Verlan Andriy, Sterten Jo Intelligent Object-Oriented Approach to Dynamic Energy Systems' Modelling	43
Гордєєв О. О., Леонтиєв К. П. Модель життєвого циклу дефекту програмного забезпечення	51
Дячук О. А., Фуртат Ю. О. Проблема надійності при участі людини-оператора в процесі прийняття рішень по керуванню об'єктами енергетики.....	61
Іванчук Н. В., Мартинюк П. М., Филипчук В. Л. Математичне моделювання очищення води в біоплато-фільтрі.....	76
Климюк Ю. Є., Бомба А. Я. Комп'ютерне прогнозування процесів «фільтрування-регенерація» швидких конусоподібних адсорбційних фільтрів з кусково-однорідними пористими завантаженнями	83
Коваленко О. Є., Косолапов В. Л. Моделі оптимізації завантаження мережі розподілених ситуаційних центрів	102
Ситник О. О., Ключка К. М., Кисельова Г. О., Кисельов В. Б. Моделювання місткової ерозії слабкострумових електричних контактів засобами MatLab	113
Шаповалова С. І. Формалізація представлення продукційних правил в Erlang	125
Відомості про авторів	140
Алфавітний покажчик авторів	142

Інститут кібернетики імені В. М. Глушкова
Національної академії наук України
Кам'янець-Подільський національний університет
імені Івана Огієнка

НАУКОВЕ ВИДАННЯ

МАТЕМАТИЧНЕ ТА КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ

Серія: Технічні науки

Збірник наукових праць

Випуск 21

Підписано до друку 2.11.2020 р. Гарнітура «Таймс».
Папір офсетний. Друк різнографічний.
Формат 60x84/16. Умовн. друк. арк. 8,4. Обл.-вид. арк. 8,7.
Тираж 50. Зам. № 907.

Кам'янець-Подільський національний університет імені Івана Огієнка,
вул. Огієнка, 61, м. Кам'янець-Подільський, 32300.
Свідоцтво серії ДК № 3382 від 05.02.2009 р.

Надруковано в Кам'янець-Подільському національному
університеті імені Івана Огієнка,
вул. Огієнка, 61, м. Кам'янець-Подільський, 32300.
Свідоцтво серії ДК № 3382 від 05.02.2009 р.