

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ВОДНОГО ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ

# ВІСНИК

**НАВЧАЛЬНО-НАУКОВОГО ІНСТИТУТУ АВТОМАТИКИ,  
КІБЕРНЕТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ НУВГП**

Збірник наукових праць

Випуск 1

Рівне-2014

У збірнику опубліковано наукові статті студентів і викладачів  
ННІ АКОТ НУВГП.

Редакційна колегія:

**Тадєєв П. О.**, к.ф.-м.н., д.пед.н., професор; **Слюсарчук В. Ю.**, д.ф.-м.н., професор; **Древецький В. В.**, д.т.н., професор; **Баховець Б. А.**, к.т.н., професор; **Кованько В. В.**, д.т.н., професор; **Бомба А. Я.**, д.т.н., професор; **Власюк А. П.**, д.т.н., професор (МЕГУ); **Тулашвілі Ю. Й.**, д.пед.н., професор; **Пастушенко В. Й.**, к.т.н., професор; **Турбал Ю. В.**, к.ф.-м.н., професор; **Круліковський Б. Б.**, к.т.н., доцент; **Мартинюк П. М.**, к.ф.-м.н., доцент; **Гладун Л. В.**, к.ф.-м.н., доцент; **Степанченко О. М.**, к.т.н., ст. викладач; **Харів Н. О.**, ст. викладач.

Свідоцтво про внесення суб'єкта видавничої справи до державного реєстру видавців, виготівників і розповсюджувачів видавничої продукції КВ № 20590-10390Р від 20.12.2013 р.

Матеріали збірника розглянуто і рекомендовано до видання на  
Вченій раді Національного університету водного господарства та  
природокористування 29 серпня 2014 р., протокол № 7.

Адреса редколегії: 33022, м. Рівне, вул. Соборна, 11, НУВГП.  
© Національний університет водного господарства та  
природокористування, 2014

Цветкова Т. П., ст. викл., Стрілець Н. І., магістр ННІ АКОТ  
(Національний університет водного господарства та природокористування,  
м. Рівне)

## **МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ МАСОПЕРЕНОСЕННЯ СОЛЕЙ ПРИ ВОЛОГОПЕРЕНОСЕННІ У ШАРУВАТИХ ҐРУНТАХ У НЕЛІНІЙНОМУ ВИПАДКУ**

Проведено математичне моделювання процесу масоперенесення сольових розчинів при вологоперенесенні в шарах області неповного насичення з урахуванням осмотичних явищ у нелінійному випадку. Числові розв'язки відповідних крайових задач знайдено за допомогою методу скінчених різниць. На основі програмної реалізації задачі виконано та проаналізовано низку числових експериментів.

**Ключові слова:** математична модель, масоперенесення, вологоперенесення, неповне насичення, шаруватий ґрунт, концентрація, напір, осмос.

Проведено математическое моделирование процесса массопереноса солевых растворов при влагопереносе в слоях области неполного насыщения с учетом осмотических явлений в нелинейной постановке. Числовые решения данных краевых задач найдено методом конечных разностей. На основе программной реализации задачи проведены численные эксперименты и сделан их анализ.

**Ключевые слова:** математическая модель, массоперенос, влагоперенос, неполное насыщение, слоистый ґрунт, концентрация, напор, осмос.

The mathematical model of the mass transfer during the moisture transfer in partial saturation soils layers including osmotic phenomena is formulated. Numerical solutions of respective boundary problems are obtained by means of finite differences. Using proper software the numerical experiments and their analysis are performed.

**Key words:** mathematical model, mass transfer, moisture transfer, partial saturation soils, layered soils, concentration, pressure, osmosis.

### **Вступ**

Стрімке індустріальне виробництво, розвиток меліорації, будівництво підземних споруд та комунікацій на сучасному етапі актуалізували необхідність дослідження процесів у приповерхневих шарах земної кори. Особливо важливим є значення вивчення вказаних процесів у розрізі розвитку сільського господарства з огляду на те, що проблема встановлення напорів вологи в ґрунтах постає як пріоритетна для визначення кількості

водних ресурсів у засушливих регіонах, для проектування дренажних та зрошувальних систем.

Над вирішенням означених вище проблем працювали багато вчених. Однак процес масоперенесення солей проаналізовано без урахування вологоперенесення [1, 2], що увиразнює актуальність проведення дослідження процесів масоперенесення з урахуванням вологоперенесення та осмотичних явищ у шаруватому ґрунті.

### Постановка задачі

Розглядається шаруватий ґрунт великої протяжності області неповного насичення (рис. 1).

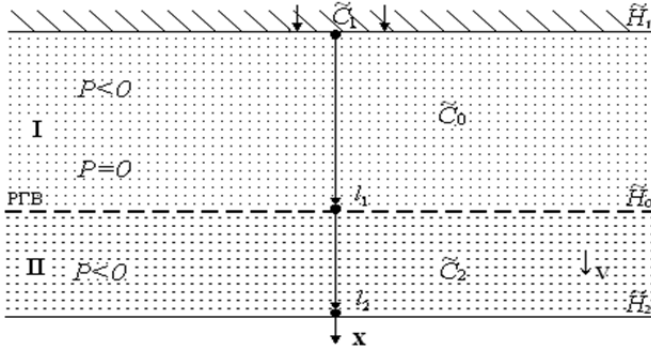


Рис. 1. Масоперенесення солей у шарах області неповного насичення

Досліджувана область вологоперенесення складається з двох шарів ґрунту, різних за типами: I шар – піщаний, II шар – глинистий.

На поверхню ґрунту потрапляють атмосферні опади і поливи з концентрацією  $C_1$ , через що на деякій глибині  $l_1$  від поверхні землі утворюється вільна поверхня рівня ґрунтових вод (РГВ), яку надалі будемо вважати нерухомою, а на глибині  $l_2$  – фронт промочування, який для простоти будемо вважати нерухомим.

Розподіл концентрації в I шарі області неповного насичення –  $\tilde{C}_0$ , в II шарі –  $\tilde{C}_2$ .

Потрібно знайти поля розподілів напорів води та концентрації солей у шарах аналізованої області водонасичення; дослідити вплив масоперенесення солей на вологоперенесення.

### Математична модель задачі

Математична модель задачі масоперенесення солей при вологоперенесенні в I шарі у загальноприйнятих позначеннях має вигляд [1, 2, 4, 5]:

$$\frac{\partial \left( D_1(c_1) \frac{\partial c_1}{\partial x} \right)}{\partial x} - V_1(c_1) \frac{\partial c_1}{\partial x} - \gamma_1(c_1 - C_1^*) = \sigma_1 \frac{\partial c_1}{\partial t}, \quad (1)$$

$$\mu(h_1) \frac{\partial h_1}{\partial t} = \frac{\partial}{\partial x} \left( k_1(c_1, h_1) \frac{\partial h_1}{\partial x} \right) - v_1(c_1) \frac{\partial c_1}{\partial x}, \quad (2)$$

$$V_1 = -k_1(c_1, h_1) \frac{\partial h_1}{\partial x} + v_1 \frac{\partial c_1}{\partial x}, \quad (3)$$

$$c_1(x, 0) = \tilde{C}_0(x), \quad c_1(0, t) = \tilde{C}_1(t), \quad (4)$$

$$h_1(x, 0) = \tilde{H}_0(x), \quad h_1(0, t) = \tilde{H}_1(t). \quad (5)$$

Математична модель задачі у II шарі має вигляд:

$$\frac{\partial \left( D_2(c_2) \frac{\partial c_2}{\partial x} \right)}{\partial x} - V_2(c_2) \frac{\partial c_2}{\partial x} - \gamma_2(c_2 - C_2^*) = \sigma_2 \frac{\partial c_2}{\partial t}, \quad (6)$$

$$\mu(h_2) \frac{\partial h_2}{\partial t} = \frac{\partial}{\partial x} \left( k_2(c_2, h_2) \frac{\partial h_2}{\partial x} \right) - v_2(c_2) \frac{\partial c_2}{\partial x}, \quad (7)$$

$$V_2 = -k_2(c_2, h_2) \frac{\partial h_2}{\partial x} + v_2 \frac{\partial c_2}{\partial x}, \quad (8)$$

$$c_2(x, 0) = \tilde{C}_0(x), \quad c_2(l_2, t) = \tilde{C}_2(t), \quad (9)$$

$$h_2(x, 0) = \tilde{H}_0(x), \quad h_2(0, t) = H_2(t), \quad h_2(l_2, t) = \tilde{H}_2(x). \quad (10)$$

На межі шарів I та II задані умови спряження для напору та концентрації мають такий вигляд:

$$[h]_{x=l_1} = [c]_{x=l_1} = \left[ c \tilde{v} - D(c) \frac{\partial c}{\partial x} \right]_{x=l_1} = 0. \quad (11)$$

У математичних моделях (1)–(11) використано такі позначення:  $D_1(c_1)$ ,  $D_2(c_2)$  – коефіцієнти конвективної дифузії,  $\gamma_1$ ,  $\gamma_2$  – коефіцієнти масообміну,  $C_1^*$ ,  $C_2^*$  – концентрації граничного насичення сольових розчинів;  $\sigma_1$ ,  $\sigma_2$  – пористості ґрунтів;  $k_1(c_1, h_1)$ ,  $k_2(c_2, h_2)$  – коефіцієнти фільтрації;  $V_1$ ,  $V_2$  – швидкості фільтрації сольових розчинів;  $h_1$ ,  $h_2$  – напори вологи;  $v_1(c_1)$ ,  $v_2(c_2)$  – осмотичні функції;  $\mu(h) = \frac{\partial \omega}{\partial h}$ ,  $[h]_{l_1} = [c]_{l_1}$  – стрибки функцій

щодо напорів та концентрації солей, задані на межі двох шарів області.

Потрібно розрахувати розподіли напорів вологи, поля розподілів концентрації солей у шарах області неповного насичення.

### Числовий метод розв'язування задач

Розв'язок поставлених задач (1)–(11) знайдено за допомогою методу скінчених різниць [3]. Розглянемо алгоритм розв'язування задач вологоперенесення (2), (5), (11) та задачі масоперенесення (1), (3), (4), (11) в І шарі області неповного насичення.

*Алгоритм розв'язування задачі вологоперенесення*

Для знаходження розв'язку задачі вологоперенесення (2), (5) та (7), (10) використаємо неявну різницеву схему [5, 6].

Неявна різницєва схема для задачі вологоперенесення (2), (5) має вигляд:

$$\mu_i^k \frac{H_{li}^{k+1} - H_{li}^k}{\tau} = \frac{1}{h_1} \left[ a_{li+1}^k \frac{H_{li+1}^{k+1} - H_{li}^k}{h_1} - a_{li}^k \frac{H_{li}^{k+1} - H_{li-1}^{k+1}}{h_1} \right] - \frac{1}{h_1} \left( Z_{li+1}^k \cdot \frac{C_{li+1}^{k+1} - C_{li}^{k+1}}{h_1} - Z_{li}^k \cdot \frac{C_{li}^{k+1} - C_{li-1}^{k+1}}{h_1} \right), \quad (12)$$

де  $a_{li}^k = 0,5(K(H_{li}^k, C_{li}^k) + K(H_{li-1}^k, C_{li-1}^k))$ ,  $a_{li+1}^k = 0,5(K(H_{li+1}^k, C_{li+1}^k) + K(H_{li}^k, C_{li}^k))$ ,  
 $Z_{li+1}^k = 0,5(v_1(C_{li+1}^k) + v_1(C_{li}^k))$ ,  $Z_{li}^k = 0,5(v_1(C_{li}^k) + v_1(C_{li-1}^k))$ ,  
 $\mu_i^k = a \rho g \left( 1 - \frac{2h_2}{H_{i+1}^k - H_{i-1}^k} \right)$ .

Числовий розв'язок задачі вологоперенесення знайдемо на основі методу прогонки. Різницєва схема (12) у прогончному вигляді буде мати такий вигляд:

$$a_1 H_{li-1}^{k+1} - b_1 H_{li}^{k+1} + c_1 H_{li+1}^{k+1} = -f_1, \quad (13)$$

де  $a_1 = \frac{a_{li}^k}{h_1^2}$ ,  $b_1 = \frac{a_{li+1}^k}{h_1^2}$ ,  $c_1 = \frac{a_{li+1}^k - a_{li}^k}{h_1^2} + \frac{\mu_{li}}{\tau}$ ,  
 $f_1 = -\frac{\mu_{li}^k}{\tau} H_{li}^k + \frac{1}{h_1} \left( Z_{li+1}^k \cdot \frac{C_{li+1}^{k+1} - C_{li}^{k+1}}{h_1} - Z_{li}^k \cdot \frac{C_{li}^{k+1} - C_{li-1}^{k+1}}{h_1} \right)$ ,  
 $i = \overline{1, n}$ ,  $k = \overline{1, m}$ .

Таким чином, згідно з методом прогонки розв'язок (13) має вигляд:

$$H_{li}^{k+1} = \alpha_{i+1}^1 H_{li+1}^{k+1} + \beta_i^1, \quad (14)$$

де  $\alpha_{i+1}^1 = \frac{b_1}{c_1 - \alpha_i^1 a_1}$ ,  $\beta_{i+1}^1 = \frac{a_1 \alpha_i^1 + f_1}{c_1 - \alpha_i^1 a_1}$ ,  $i = \overline{1, n}$ ,  $\alpha_1 = 0$ ,  $\beta_1 = \widetilde{H}_1$ .

*Алгоритм розв'язування задачі масоперенесення солей*

Для знаходження числового розв'язку задачі перенесення солей (1), (3), (4) застосуємо монотонну різницєву схему [4]. Різницєва схема поставленої задачі має вигляд:

$$\begin{aligned} \frac{1}{h_1^2} \left( \mu_{i11}^2 + \mu_{i12}^k d_{1i+1}^k + \frac{|r_{1i}^k|}{h_1} + \frac{\sigma}{\tau} \right) C_{1i}^{k+1} &= \frac{1}{h_1} \left( \frac{\mu_{i11}^k d_{1i}^k}{h_1} - r_{1i-}^k \right) C_{1i-1}^{k+1} + \\ &+ \frac{1}{h_1} \left( \frac{\mu_{i12}^k d_{1i+1}^k}{h_1} - r_{1i+}^k \right) C_{1i+1}^{k+1} + \frac{\sigma}{\tau} C_{1i}^k + f_i^k, \end{aligned} \quad (15)$$

де  $d_{1i}^k = \frac{D(x_i, C_{1i}^k) + D(x_{i-1}, C_{1i-1}^k)}{2}$ ,  $d_{1i+1}^k = \frac{D(x_i, C_{1i+1}^k) + D(x_i, C_{1i}^k)}{2}$ ,  
 $r_{1i+}^k = 0,5(V(x_i, C_{1i}^k) + |V(x_i, C_{1i}^k)|) \geq 0$ ,  $r_{1i-}^k = 0,5(V(x_i, C_{1i}^k) - |V(x_i, C_{1i}^k)|) \leq 0$ .  
 $V(x_i, C_{1i}^k) = r_{1i+}^k + r_{1i-}^k$ ,  $f_1 = -\gamma_1(C_{1i}^{k+1} - C_1^*)$ ,  $C_i^0 = C_0(ih_1)$ ,  $C_i^{k+1} = C_1((k+1)\tau)$ ,  
 $C_n^{k+1} = C_2((k+1)\tau)$ ,  $\mu_{i11}^k = \frac{1}{1 + \frac{0,5h_1|r_{1i}^k|}{d_{1i}^k}}$ ,  $\mu_{i12}^k = \frac{1}{1 + \frac{0,5h_1|r_{1i}^k|}{d_{1i+1}^k}}$ .

Розв'язок задачі знайдемо за допомогою методу прогонки. Різницєва схема (15) у прогонічному вигляді є такою:

$$a_2 C_{1i-1}^{k+1} - c_2 C_{1i}^{k+1} + b_2 C_{1i+1}^{k+1} = -f_2, \quad (16)$$

де  $a_2 = \frac{1}{h_1} \left( \frac{d_{1i}^k \mu_{i11}^k}{h_1} - r_{1i-}^k \right)$ ,  $b_2 = \frac{1}{h_1} \left( \frac{d_{1i+1}^k \mu_{i12}^k}{h_1} + r_{1i+}^k \right)$ ,  
 $c_2 = \frac{1}{h_1^2} \left( d_{1i}^k \mu_{i11}^k + d_{1i+1}^k \mu_{i12}^k \right) + \frac{|r_{1i}^k|}{h_1} + \frac{\sigma_1}{\tau}$ ,  $f_2 = -\frac{\gamma_1}{\tau} C_{1i}^{k+1} - C_1^*$ .

Так, загальний розв'язок задачі масоперенесення за методом прогонки знаходимо у вигляді:

$$\begin{aligned} C_{1i}^{k+1} &= \alpha_{i+1}^2 C_{1i+1}^{k+1} + \beta_{1i+1}^2, \quad (17) \\ \alpha_{i+1}^2 &= \frac{b_1}{c_1 - \alpha_i^2 a_1}, \quad \beta_{1i+1}^2 = \frac{a_1 \beta_i - f_1}{c_1 - \alpha_i^2 a_1}, \\ \alpha_1 &= 0, \quad \beta_1 = \tilde{C}_1(t). \end{aligned}$$

Щодо хімічного осмосу  $v_1 = v_1(c)$  різницєва схема рівняння швидкості фільтрації (3), (8) має вигляд:

$$V_{1i}^{k+1} = -k_{1i}^k \left( \frac{H_{1i+1}^k - H_{1i-1}^k}{2h_2} + \frac{1}{h_2} \left( Z_{1i+1}^k \frac{C_{1i+1}^{k+1} - C_{1i}^{k+1}}{h_2} - Z_{1i}^k \frac{C_{1i}^{k+1} + C_{1i-1}^{k+1}}{h_2} \right) \right), \quad (18)$$

де  $Z_{1i+1}^k = 0,5(v_1(C_{1i+1}^k) + v_1(C_{1i}^k))$ ,  $Z_{1i}^k = 0,5(v_1(C_{1i}^k) + v_1(C_{1i-1}^k))$ .

Аналогічно до (12)–(18) здійснено побудову обчислювального алгоритму знаходження числового розв'язку задачі масоперенесення солей у II шарі аналізованої області.

Розв'язок задачі (11) на РГВ знайдемо за допомогою методу прогонки [4, 5]. Відтак, обчислювальний алгоритм розв'язування задачі масоперенесення солей при вологоперенесенні побудовано повністю.

**Програмна реалізація та результати числових експериментів**

Програмну реалізацію поставленої задачі виконано в середовищі візуально-подійного, об'єктно-орієнтованого програмування Microsoft Visual Studio 2012 на платформі .NET 4.0 за таких вхідних даних:

$$T = 360 \text{ діб}, l_1 = 10 \text{ м}, l_2 = 10 \text{ м}, \sigma_1 = 0,4, \sigma_2 = 0,2,$$

$$\gamma_1 = 0,0065, \gamma_2 = 0,00065, C^* = 350 \text{ г/літр}, \tilde{C}_1 = 10 \text{ г/літр},$$

$$\tilde{C}_2 = 10 \text{ г/літр}, H_1 = 4,4 \text{ м}, H_2 = 0,1 \text{ м},$$

$$k_1(c) = 0.0002 c^6 - 0.0088 c^5 + 0.162 c^4 - 1.3194 c^3 + 3.9229 c^2 +$$

$$+ 0.0223 c + 18.187,$$

$$k_2(c) = 5.9404 \cdot 10^{-2} c^5 - 0.16703 c^4 + 0.17051 c^3 - 0.074311 c^2 +$$

$$+ 0.010563 c + 1.0053 \cdot 10^{-3}, c \in [0, 1].$$

У результаті програмної реалізації обчислювального алгоритму (12)–(18) знайдено числові розв'язки задач вологоперенесення, вологоперенесення з масоперенесенням та задачі масоперенесення солей з урахуванням вологоперенесення в шарах I та II цієї області. Також знайдено розв'язки цих задач сумісно для обох шарів в області неповного насичення.

Унаслідок розв'язання задачі вологоперенесення в шаруватому ґрунті отримано такі графіки розподілу напорів води (рис. 2, 3):

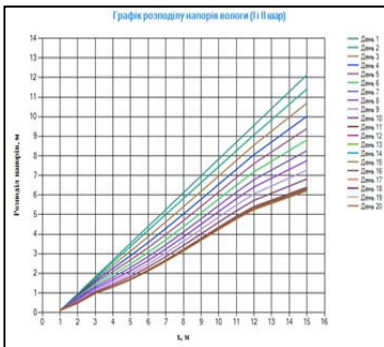


Рис. 2. Розподіл напорів води для задачі чистого вологоперенесення

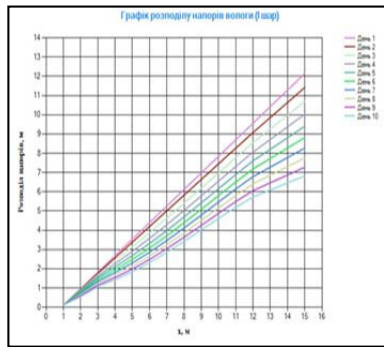


Рис. 3. Розподіл напорів води з урахуванням концентрації солей

Із використанням результатів задачі вологоперенесення розв'язано задачу масоперенесення сольових розчинів у шаруватому ґрунті, внаслідок чого отримано такі графіки розподілу концентрації солей (рис. 4, 5).



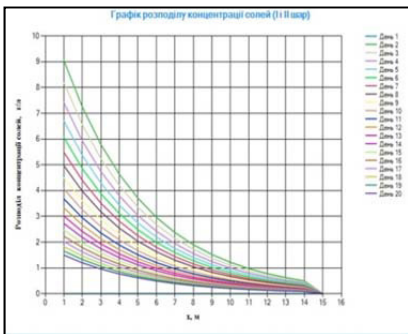


Рис. 4. Розподіл концентрації солей у шаруватому ґрунті

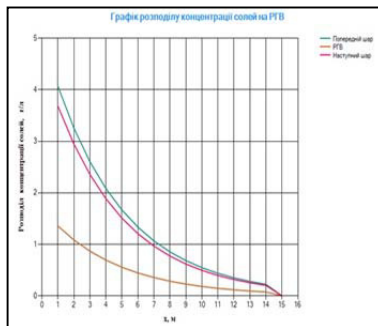


Рис. 5. Розподіл концентрації солей на РГВ

**Висновки.** На основі програмної реалізації задачі було проведено низку числових експериментів, унаслідок чого отримано такі результати: розподіл напорів води зменшується з часом та глибиною по всій області шаруватого ґрунту; розподіл поширення води у піщаному шарі ґрунту є рівномірним по всьому ґрунтовому масиві; поширення води у глинистому шарі відбувається у більш вузькому верхньому прошарку ґрунту через невелику пористість цього ґрунтового середовища; розподіл концентрації солей з урахуванням розподілу напорів води зменшується з часом та глибиною, що потрібно пояснювати вимиванням сольових розчинів із пористого середовища; встановлено, що осмотичні явища впливають на розподіл концентрації солей пропорційно до вибору коефіцієнта осмосу або осмотичної функції.

1. Веригин Н. Н. Диффузия и массообмен при фильтрации жидкостей в пористых средах / Н. Н. Веригин, Б. С. Шержуков // Развитие исследований по теории фильтрации в СССР (1917–1967). – М. : Наука, 1969. – С. 237–313. 2. Сергиенко И. В. Математическое моделирование и исследование процессов в неоднородных средах. / И. В. Сергиенко, В. В. Скопецкий, В. С. Дейнека. – К. : Наук. думка, 1991. – С. 27–31. 3. Самарский А. А. Теория разностных схем / А. А. Самарский. – [3-е изд.] – М. : Наука, 1989. – 616 с. 4. Власюк А. П. Монотонна різницева схема для нелінійного одновимірного рівняння параболічного типу, що містить першу похідну / А. П. Власюк, О. П. Остапчук, О. М. Степанченко // Вісник Київського університету. Серія фіз.-мат.н. Вип. №3. – 2005. – С. 188–197. 5. Цветкова Т. П. Математичне моделювання масопереносу з врахуванням осмосу на вологоперенос в одновимірному випадку / Т. П. Цветкова // Математичне та комп. моделювання. Серія : Техн. науки. Вип. 4 : зб. наук. праць. – К-П. НУ ім. І. Огієнка. – 2010. – С. 227–233. 6. Власюк А. П. Математичне моделювання солеперенесення при сумісній фільтрації та вологоперенесенні у насичено-ненасичених ґрунтах в лінійній постановці / А. П. Власюк, Т. П. Цветкова // Вісник Київського університету. Серія фіз.-мат.н. Вип. №3. – 2013. – С. 118–124.

Цветкова Т. П., ст. викл., Волянiк Г. В., студентка 5 курсу ННІ АКОТ (Національний університет водного господарства та природокористування, м. Рівне)

## **МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ СОЛЕПЕРЕНЕСЕННЯ ПРИ ВОЛОГОПЕРЕНЕСЕННІ У ШАРУВАТИХ ҐРУНТАХ У ЛІНІЙНІЙ ПОСТАНОВЦІ**

Проведено математичне моделювання солеперенесення при вологоперенесенні в шаруватому ґрунті області неповного насичення з урахуванням осмотичних явищ у лінійному випадку. Числові розв'язки крайової задачі знайдено за допомогою методу скінчених різниць. На основі програмної реалізації задачі виконано та проаналізовано спектр числових експериментів.

**Ключові слова:** математична модель, солеперенесення, вологоперенесення, неповне насичення, шаруватий ґрунт, концентрація, напір, осмос.

Проведено математическое моделирование солепереноса при вологопереносе в слоистом ґрунте области неполного насыщения с учетом осмотических явлений в линейном случае. Числовые решения краевой задачи найдено методом конечных разностей. На основе программной реализации задачи проведен ряд численных экспериментов и сделано их анализ.

**Ключевые слова:** математическая модель, солеперенос, вологоперенос, неполное насыщение, слоистый ґрунт, концентрация, напор, осмос.

The mathematical model of the mass transfer during the moisture transfer in partial saturation soils layers including osmotic phenomena is formulated in the linear formulation. Numerical solutions of respective boundary problems are obtained by means of finite differences method. Proper software is developed using which numerical experiments and their analysis are performed.

**Key words:** mathematical model, salt transfer, moisture transfer, region of incomplete saturation of soil, layered soils, concentration, distribution of water pressures, osmosis.

### **Вступ**

Значні зміни водного режиму, зумовлені і природно-кліматичними чинниками, і впливом несприятливих техногенних умов, можуть призвести до розвитку небезпечних гідрогеологічних процесів – підтоплення територій, забруднення, засолення родючих ґрунтів. Тому на посилену увагу заслуговує

проблема дослідження процесів солеперенесення, зокрема такі її аспекти, як: поширення забруднень, засолення ґрунтів, вологоперенесення – вивчення динаміки поширення вологи в ненасичених ґрунтах.

Різні аспекти означеної вище проблеми були предметом наукових пошуків вітчизняних та зарубіжних учених. Достатньо розроблено теорію масоперенесення у ненасичених ґрунтах: відомими є роботи М. О. Веригіна, І. В. Сергієнка, В. С. Дейнеки, В. В. Скопечького, І. І. Ляшко. Теоретичні основи вологоперенесення в ґрунтах закладено в працях С. Ф. Авер'янова, А. Дарсі, Річардса, О. Я. Олійника, В. Л. Полякова, М. М. Рекса, П. І. Ковальчука та ін. Однак у проаналізованих наукових студіях процес вологоперенесення висвітлено без урахування масоперенесення.

Мета запропонованої роботи – проведення дослідження процесу солеперенесення з урахуванням вологоперенесення в шаруватих ґрунтах [1, 2].

### Постановка задачі

Розглядається шаруватий ґрунт великої протяжності області неповного насичення ( $P < 0$ ) (рис. 1).

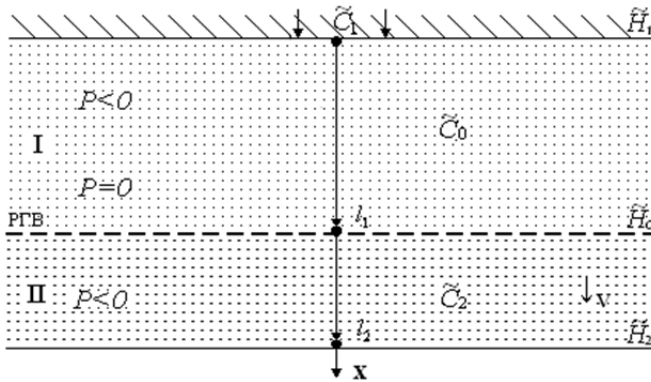


Рис. 1. Масоперенесення солей у шарах області неповного насичення

Досліджувана область вологоперенесення складається з двох різних за типами шарів ґрунту: I шар – піщаний, II шар – глинистий.

На поверхню ґрунту потрапляють атмосферні опади і поливи з концентрацією  $\tilde{C}_1$ . Розподіл концентрації в I шарі області неповного насичення –  $\tilde{C}_0$ , в II шарі –  $\tilde{C}_2$ . На деякій глибині  $l_1$  від поверхні землі утворюється вільна поверхня рівня ґрунтових вод (РГВ), яку надалі будемо вважати нерухомою. На глибині  $l_2$  знаходиться фронт промочування, який будемо вважати нерухомим.

Потрібно знайти поля розподілів напорів вологи та концентрації солей у шаруватому ґрунті обраної для вивчення області водонасичення; дослідити

вплив масоперенесення солей на вологоперенесення; провести числові експерименти та проаналізувати отримані результати.

### Математична модель задачі

Математична модель задачі солеперенесення при вологоперенесенні в I шарі в загальноприйнятих позначеннях має вигляд [1, 2, 6, 7]:

$$\frac{\partial \left( D_1 \frac{\partial c_1}{\partial x} \right)}{\partial x} - v_1 \frac{\partial c_1}{\partial x} - \gamma_1 (c_1 - C_1^*) = \sigma_1 \frac{\partial c_1}{\partial t}, \quad (1)$$

$$\mu(h_1) \frac{\partial h_1}{\partial t} = \frac{\partial}{\partial x} \left( k_1 \frac{\partial h_1}{\partial x} \right) - v_1 \frac{\partial^2 c_1}{\partial x^2}, \quad (2)$$

$$v_1 = -k_1 \frac{\partial h_1}{\partial x} + v_1 \frac{\partial c_1}{\partial x}, \quad (3)$$

$$c_1(x, 0) = \tilde{C}_0(x), \quad c_1(0, t) = \tilde{C}_1(t), \quad (4)$$

$$h_1(x, 0) = \tilde{H}_0(x), \quad h_1(0, t) = \tilde{H}_1(t). \quad (5)$$

Математична модель задачі в II шарі має вигляд:

$$\frac{\partial \left( D_2 \frac{\partial c_2}{\partial x} \right)}{\partial x} - v_2 \frac{\partial c_2}{\partial x} - \gamma_2 (c_2 - C_2^*) = \sigma_2 \frac{\partial c_2}{\partial t}, \quad (6)$$

$$\mu(h_2) \frac{\partial h_2}{\partial t} = \frac{\partial}{\partial x} \left( k_2 \frac{\partial h_2}{\partial x} \right) - v_2 \frac{\partial^2 c_2}{\partial x^2}, \quad (7)$$

$$v_2 = -k_2 \frac{\partial h_2}{\partial x} + v_2 \frac{\partial c_2}{\partial x}, \quad (8)$$

$$c_2(x, 0) = \tilde{C}_0(x), \quad c_2(l_2, t) = \tilde{C}_2(t), \quad (9)$$

$$h_2(x, 0) = \tilde{H}_0(x), \quad h_2(0, t) = \tilde{H}_2(t), \quad h_2(l_2, t) = \tilde{H}_2(x). \quad (10)$$

На межі шарів I та II задано умови спряження для напору та концентрації такого вигляду:

$$[h] |_{x=l_1} = [c] |_{x=l_1} = \left[ cv_x - D \frac{\partial c}{\partial x} \right] \Big|_{x=l_1} = 0. \quad (11)$$

У математичних моделях (1) – (11) використані такі позначення:  $D_1, D_2$  – коефіцієнти конвективної дифузії,  $\gamma_1, \gamma_2$  – коефіцієнти масообміну,  $C_1^*, C_2^*$  – концентрації граничного насичення сольових розчинів;  $\sigma_1, \sigma_2$  – пористості ґрунтів;  $k_1, k_2$  – коефіцієнти фільтрації;  $v_1, v_2$  – швидкості фільтрації сольових розчинів;  $h_1 = h_2$  – напори води;  $v_1, v_2$  – коефіцієнти осмосу;  $\mu(h)$  – коефіцієнт вологоємності,  $[h]_{l_1} = [c]_{l_1}$  –

стрибки функції щодо напорів та концентрації солей, задані на межі двох шарів області.

Потрібно розрахувати розподіли напорів вологи та концентрації солей у шарах області неповного насичення.

### Числовий метод розв'язування задачі

Для знаходження розв'язку поставлених задач (1)–(11) використаємо метод скінчених різниць [6]. Розглянемо алгоритм розв'язування задачі вологоперенесення (1)–(5), (11) та задачі солеперенесення (1), (3), (4), (11) в шарі I області неповного насичення.

Для знаходження розв'язку задачі вологоперенесення (2), (5) та (7), (10) використаємо неявну різницеву схему [4, 5, 6].

Неявна різницєва схема для задачі вологоперенесення (2), (5) має вигляд:

$$\mu_{li} \frac{H_{li}^{k+1} - H_{li}^k}{\tau} = k_1 \frac{H_{li-1}^{k+1} - 2H_{li}^{k+1} + H_{li+1}^{k+1}}{h_1^2} - V_1 \left( \frac{C_{li+1}^{k+1} - 2C_{li}^{k+1} + C_{li-1}^{k+1}}{h_1^2} \right), \quad (12)$$

де

$$\mu_{li} = a \rho g \left( 1 - \frac{2h_1}{H_{li+1}^k - H_{li-1}^k} \right).$$

Числовий розв'язок задачі знайдемо за допомогою методу прогонки. Різницєва схема (12) у прогночному вигляді буде мати такий вигляд:

$$a_1 H_{li-1}^{k+1} - c_1 H_{li}^{k+1} + b_1 H_{li+1}^{k+1} = -f_1, \quad (13)$$

де  $a_1 = b_1 = \frac{k_1}{h_1^2}$ ,  $c_1 = \frac{\mu_{li}}{\tau} + \frac{2k_1}{h_1^2}$ ,  $f_1 = \frac{\mu_{li}}{\tau} H_{li}^k - \frac{V_1}{h_1^2} (C_{li+1}^k - 2C_{li}^k + C_{li-1}^k)$ ,

$i = \overline{1, n}$ ,  $k = \overline{1, m}$ .

Таким чином, розв'язок задачі знайдемо у вигляді:

$$H_{li}^k = \alpha_{i+1} H_{li+1}^{k+1} + \beta_{i+1}, \quad (14)$$

де  $\alpha_{i+1} = \frac{b_1}{c_1 - a_1 \alpha_i}$ ,  $\beta_{i+1} = \frac{a_1 \beta_i + f_1}{c_1 - a_1 \alpha_i}$ ,  $i = \overline{1, n}$ ,  $\alpha_0 = 0$ ,  $\beta_0 = \tilde{H}_1$ .

Для знаходження числового розв'язку задачі солеперенесення (1), (3), (4), (11) застосуємо монотонну різницєву схему [3, 5, 6]. Різницєва схема поставленої задачі має вигляд:

$$D_1 \frac{C_{li+1}^{k+1} - 2C_{li}^{k+1} + C_{li-1}^{k+1}}{h_1^2} - V_1 \frac{C_{li+1}^{k+1} - C_{li}^k}{h_2} - \gamma_1 (C_{li}^{k+1} - C_{li}^*) = \sigma_1 \frac{C_{li}^{k+1} - C_{li}^k}{\tau} \quad (15)$$

або  $\eta_i^{k+1} \left( \frac{C_{li+1}^{k+1} - 2C_{li}^{k+1} + C_{li-1}^{k+1}}{h_1^2} \right) + r_{i+1}^{k+1} \frac{C_{li+1}^{k+1} - C_{li}^k}{h_1} + r_{-i}^{k+1} \frac{C_{li}^{k+1} - C_{li-1}^k}{h_1} - \gamma_1 (C_{li}^{k+1} - C_{li}^*) = \frac{\sigma_1}{\tau} (C_{li}^{k+1} - C_{li}^k)$ ,

де

$$\eta_i^k = \frac{1}{1 + \frac{h_1 |r_i^k|}{2}}, \quad r_i^k = r_{+i}^k + r_{-i}^k,$$

$$r_{+i}^k = \frac{-V_{1i}^{k+1} + |V_{1i}^{k+1}|}{2} \geq 0, \quad r_{-i}^k = \frac{-V_{1i}^{k+1} - |V_{1i}^{k+1}|}{2} \leq 0.$$

Розв'язок задачі знайдемо за методом прогонки. Різницева схема (15) в прогоночному вигляді є такою:

$$a_2 C_{1i-1}^{k+1} - c_2 C_{1i}^{k+1} + b_2 C_{1i+1}^{k+1} = -f_2, \quad (16)$$

де

$$a_2 = \frac{\tau}{\sigma_1} \left( \frac{2\eta_i^{k+1}}{h_1^2} - \frac{r_{-i}^{k+1}}{h_1} \right), \quad b_2 = \frac{\tau}{\sigma_1} \left( \frac{2\eta_i^{k+1}}{h_1^2} - \frac{r_{+i}^{k+1}}{h_1} \right),$$

$$c_2 = \frac{\tau}{\sigma_1} \left( \frac{2\eta_i^{k+1}}{h_1^2} + \frac{r_{+i}^{k+1} - r_{-i}^{k+1}}{h_1} + \gamma_1 \right), \quad f_2 = \gamma_1 C_1^* + \frac{\sigma_1}{\tau} C_1^k.$$

Так, загальний розв'язок задачі солеперенесення знайдемо у вигляді:

$$C_{1i}^k = \alpha_{i+1} C_{1i+1}^{k+1} + \beta_{i+1}, \quad (17)$$

$$\alpha_{i+1} = \frac{b_2}{c_2 - a_2 \alpha_i}, \quad \beta_{i+1} = \frac{a_2 \beta_i + f_2}{c_2 - a_2 \alpha_i}, \quad \alpha_2 = 0, \quad \beta_2 = \tilde{C}_1.$$

Різницева схема рівняння швидкості фільтрації (3), (8) має вигляд:

$$V_{1i}^{k+1} = -k_1 \left( \frac{H_{1i+1}^k - H_{1i-1}^k}{2h_1} \right) + v_1 \left( \frac{C_{1i+1}^k - 2C_{1i}^k + C_{1i-1}^k}{h_1^2} \right).$$

Аналогічно до (12)–(17) побудуємо обчислювальний алгоритм знаходження числового розв'язку задачі солеперенесення в шарі II досліджуваної області.

Розв'язок задачі на РГВ (11) знайдено за допомогою методу прогонки [5]. Відтак, обчислювальний алгоритм розв'язування задачі солеперенесення при вологоперенесенні побудовано повністю.

### Програмна реалізація та результати числових експериментів

Для реалізації обчислювального алгоритму поставленої задачі розроблено програму, яку виконано в системі об'єктно-орієнтованого, візуально-подвійного програмування Microsoft Visual Studio 2010 на мові C# за таких вхідних даних:

$$T = 360 \text{ діб}, \quad l_1 = 1 \text{ м}, \quad l_2 = 0,5 \text{ м}, \quad \sigma_1 = 0,4, \quad \sigma_2 = 0,2, \quad \gamma_1 = 0,0065,$$

$$\gamma_2 = 0,00065, \quad C_* = 350 \text{ г/літр}, \quad \tilde{C}_0 = 0 \text{ г/літр}, \quad \tilde{C}_1 = 0 \text{ г/літр},$$

$$\tilde{C}_2 = 5 \text{ г/літр}, \quad H_1^1 = 1,4 \text{ м}, \quad H_2^1 = 0,05 \text{ м}, \quad H_1^2 = 1,8 \text{ м}, \quad H_2^2 = 1,4 \text{ м},$$

$$v = 2,8 \cdot 10^{-5} \text{ м}^5 / \text{кг} \cdot \text{доба}, \quad k_1 = 3,5 \text{ м/добу}, \quad k_2 = 2,3 \text{ м/добу}.$$

У результаті програмної реалізації обчислювального алгоритму (12)–(17) знайдено числові розв’язки задач вологоперенесення з урахуванням та без урахування солеперенесення та задачі солеперенесення в шарах I та II аналізованої області. Також знайдено розв’язки таких задач сумісно в шаруватому ґрунті досліджуваної області водонасичення.

Шляхом розв’язання задачі вологоперенесення в шаруватому ґрунті отримуємо такі графіки розподілу напорів води (рис. 2, 3).

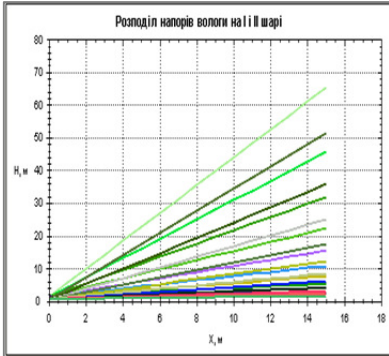


Рис. 2. Розподіл напорів води для випадку задачі чистого вологоперенесення

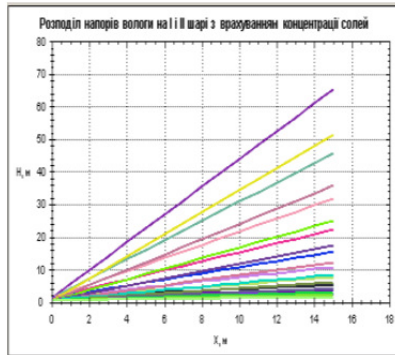


Рис. 3. Розподіл напорів води з урахуванням концентрації солей

Із використанням результатів задачі вологоперенесення розв’язано задачу солеперенесення в шаруватому ґрунті, внаслідок чого побудовано такі графіки розподілу концентрації солей (рис. 4, 5).

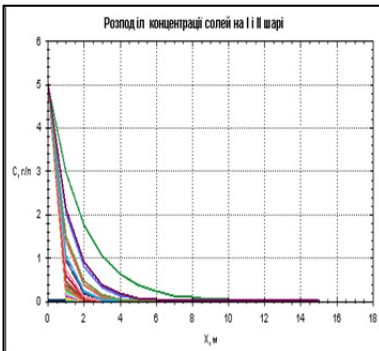


Рис. 4. Розподіл концентрації солей у шаруватому ґрунті

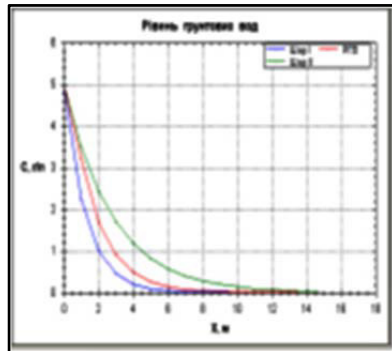


Рис. 5. Розподіл концентрації солей на РГВ

### Висновки

На основі програмної реалізації задачі було проведено числові експерименти, що дало можливість отримати такі результати: розподіл

напорів води для задачі чистого вологоперенесення зростає з глибиною та зменшується з часом. З урахуванням концентрації сольових розчинів розподіл напорів збільшується з глибиною пористого середовища.

Розподіл концентрації солей з урахуванням напорів води зменшується з часом та глибиною шаруватого ґрунту, що варто пояснювати вимиванням сольових розчинів з пористого середовища. Розподіл концентрації на вільній поверхні з часом та глибиною зменшується.

Установлено, що з урахуванням осмосу значення розподілу концентрації солей зростають: осмотичні явища впливають на розподіл концентрації солей пропорційно до вибору коефіцієнта осмосу.

У разі збільшення коефіцієнта фільтрації швидкість фільтрації зростає. Із збільшенням пористості ґрунту концентрація сольового розчину зменшується. Після збільшення коефіцієнта масообміну концентрація солей зростає.

Шляхом проведення числових експериментів визначено, що розподіл поширення води у піщаному шарі ґрунту відбувається рівномірно по всьому ґрунтовому масиві, а у глинистому шарі проходить у більш вузькому верхньому прошарку ґрунту через невелику пористість такого ґрунтового середовища.

1. Власюк А. П. Математичне моделювання масоперенесення при фільтрації та вологоперенесенні в ненасичених шаруватих ґрунтах з врахуванням інфільтрації та осмосу / А. П. Власюк, Т. П. Цветкова // Сучасні проблеми прикладної математики та інформатики : тез. XVIII Всеукр. наук. конф. (4–5 жовтня 2012 р., м. Львів), 2012. – С. 63.
2. Власюк А. П. Математичне моделювання перенесення солей при вологоперенесенні в шаруватих ґрунтах / А. П. Власюк, Т. П. Цветкова // Сучасні проблеми математичного моделювання та обчисл. методів : тез. I Всеукр. наук. конф. (22–23 лютого 2013 р., Рівне), 2013. – С. 46.
3. Власюк А. П. Монотонна різницєва схема для нелінійного одновимірного рівняння параболічного типу, що містить першу похідну / А. П. Власюк, О. П. Остапчук, О. М. Степанченко // Вісник Київського університету. Серія фізико-математичні науки. Вип. №3. – 2005. – С. 188–197.
4. Цветкова Т. П. Математичне моделювання масопереносу з врахуванням осмосу на вологоперенос в одновимірному випадку / Т. П. Цветкова // Математичне та комп. моделювання. Серія: Техн. науки. Випуск 4 : зб. наук. праць. – К-П. НУ ім. І. Огієнка. – 2010. – С. 227–233.
5. Власюк А. П. Математичне моделювання солеперенесення при сумісній фільтрації та вологоперенесенні у насичено-ненасичених ґрунтах в лінійній постановці / А. П. Власюк, Т. П. Цветкова // Вісник Київського університету. Серія фізико-математичні науки. Вип. №3. – 2013. – С. 118–124.
6. Самарский А. А. Теория разностных схем / А. А. Самарский. – [3-е изд.]. – М. : Наука, 1989. – 616 с.
7. Сергиенко И. В. Математическое моделирование и исследование процессов в неоднородных средах / И. В. Сергиенко, В. В. Скопецкий, В. С. Дейнека. – К. : Наукова думка, 1991. – С. 27–31.



**Федорчук Н. А., ст. викладач, Бормат Д. М., студент 5 курсу ННІ АКОТ**  
(Національний університет водного господарства та природокористування,  
м. Рівне)

### **МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ НАПРУЖЕНО-ДЕФОРМОВАНОГО СТАНУ БАГАТОШАРОВОГО ҐРУНТОВОГО МАСИВУ З РУХОМОЮ ВІЛЬНОЮ ПОВЕРХНЕЮ ПІД ВПЛИВОМ ТЕПЛО-МАСОПЕРЕНЕСЕННЯ**

**Розглянуто задачу напружено-деформованого стану (НДС) багатошарового ґрунтового масиву з рухомою вільною поверхнею під впливом тепло-масоперенесення. Знайдено числовий розв'язок відповідної крайової задачі за допомогою методу скінченних різниць. Проведено та проаналізовано серію числових експериментів.**

**Ключові слова:** напружено-деформований стан, рухома вільна поверхня, тепло-масоперенесення.

**Рассмотрена задача напряженно-деформированного состояния (НДС) многослойного грунтового массива с подвижной свободной поверхностью под воздействием тепло-массопереноса. Найдено численное решение соответствующей краевой задачи методом конечных разностей. Проведена серия численных экспериментов и выполнен их анализ.**

**Ключевые слова:** напряженно-деформированное состояние, подвижная свободная поверхность, тепло-массоперенос.

**The problem of stressed and deformed state (SDS) of a multilayer soil mass at presence of a moving free surface under heat and mass transferring was solved. There is obtained the numerical solution of respective boundary value problem by the method of finite differences. A series of numerical experiments and their analysis was conducted.**

**Key words:** stressed and deformed state, moving free surface, heat and mass transferring.

**Питання врахування деформаційних характеристик ґрунтових масивів набувають усе більш вагомого значення на сучасному етапі під час проектування, будівництва та експлуатації гідротехнічних, цивільних, промислових об'єктів і споруд, оскільки в ході експлуатації таких об'єктів НДС ґрунтових основ, на яких їх побудовано, може значно змінюватись.**

**Як відомо, на напружено-деформованому стані (НДС) ґрунтових масивів можуть досить відчутно позначатися фізико-механічні характеристики**

грунтів, гідрогеологічні умови ґрунтового середовища та тип навантажень, зокрема: фільтрація підземних вод, масоперенесення, теплоперенесення розчинених речовин фільтраційними потоками (розчинення та вимивання порід).

Важливий вплив на стійкість споруд має вільна поверхня, зміна положення якої може змінити НДС ґрунтових масивів і, зрештою, призвести до виникнення різноманітних негативних явищ та аварій.

Також у природних умовах ґрунтовий масив може складатись із багатьох шарів, кожному з яких притаманні певні характеристики, відмінні від характеристик інших шарів (пористості, коефіцієнти Ламе, конвективної дифузії, термодифузії, теплопровідності, коефіцієнти фільтрації та ін.).

Задачу НДС ґрунтів досліджено й описано в працях [1–10]. Однак у цих роботах недостатньо вивчено НДС багат шарового ґрунтового масиву під впливом тепло-масоперенесення та за наявності рухомої вільної поверхні.

Тому, враховуючи вищесказане, **актуальною задачею** вважаємо математичне моделювання та дослідження НДС багат шарового ґрунтового масиву з урахуванням тепло-масоперенесення та наявності рухомої вільної поверхні.

**Метою роботи** є постановка задачі та вдосконалення математичної моделі НДС багат шарового ґрунтового масиву з урахуванням тепло-масоперенесення та за наявності рухомої вільної поверхні; числове розв’язання поставленої крайової задачі; отримання результатів числових експериментів у вигляді графіків; аналіз зміни НДС багат шарового ґрунтового масиву за наявності рухомої вільної поверхні ґрунтових вод та з урахуванням тепло-масоперенесення.

**Розглянемо багат шаровий ґрунтовий масив** з рухомою вільною поверхнею, що займає область  $\Omega = \{(x, y) : 0 < x < l_1, 0 < y < l_2\}$  (див. рис. 1).

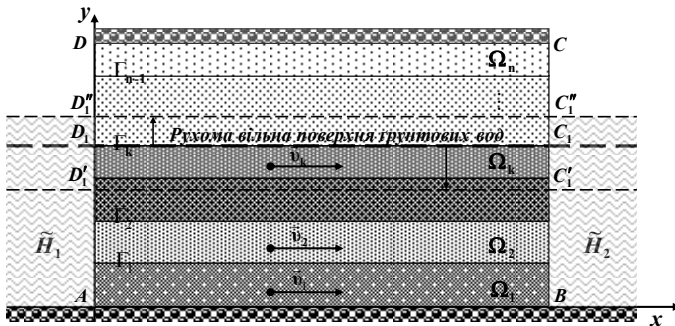


Рис. 1. Багат шаровий ґрунтовий масив за наявності в ньому рухомої вільної поверхні та фільтрації підземних вод під вільною поверхнею

Кількість шарів у ґрунтовому масиві дорівнює  $n$ . Шари пронумеровано знизу вверху. Вільна поверхня ґрунтових вод ( $C_1D_1$ ) знаходиться на рівні  $\Gamma_k$ . Відтак, шари ґрунтового масиву, розміщені вище  $\Gamma_k$ , знаходяться в природному стані, а нижче  $\Gamma_k$  – за повної вологоємності.

Багатошаровий ґрунтовий масив вважають пружно-деформівним у межах лінійної теорії пружності з різними пружними параметрами (коефіцієнтами Ламе)  $\lambda_i(c_i, T_i)$ ,  $\mu_i(c_i, T_i)$ ,  $i = \overline{1, n}$ , які залежать від концентрації сольового розчину та температури. На шари ґрунту діє сила тяжіння, а в разі водонасичених шарів ґрунтового масиву – архімедова та фільтраційна сили.

Задано значення п'єзометричних напорів  $\tilde{H}_1$ ,  $\tilde{H}_2$  лівому ( $AD_1$ ) та правому ( $BC_1$ ) водним басейнам, відповідно, причому  $\tilde{H}_1 > \tilde{H}_2$ . У лівому басейні задано значення концентрації солей  $\tilde{C}_1$ , а у правому – умову швидкого винесення солей. Значення температур  $\tilde{T}_1$  та  $\tilde{T}_2$  ( $\tilde{T}_1 > \tilde{T}_2$ ) задано на межах  $AD$  та  $BC$  відповідно. В результаті різниці напорів відбувається перенесення розчинених у воді речовин та тепла фільтраційним потоком. При цьому процес фільтрації розчинених у воді речовин та тепла підлягає законам Дарсі, Фіка та Фур'є.

Потрібно розрахувати значення зміщень, напружень та деформацій, а також розподіл поля напорів, швидкостей фільтрації, концентрації солей та температури ґрунтового масиву за умови, що немає зміщення нижньої ( $AB$ ) та верхньої ( $CD$ ) меж багатошарового ґрунтового масиву, а також  $AB$  та  $CD$  є теплоізольованими і непроникними.

**Математичну модель** двовимірної задачі НДС багатошарового ґрунтового масиву з рухомою вільною поверхнею під впливом тепло-масоперенесення в загальноприйнятих позначеннях можна описати такою крайовою задачею [1–10]:

$$\begin{aligned}
 & \mu_i(c_i, T_i)\Delta U_i + (\lambda_i(c_i, T_i) + \mu_i(c_i, T_i)) \frac{\partial}{\partial x} \left( \frac{\partial U_i}{\partial x} + \frac{\partial V_i}{\partial y} \right) + \left[ \frac{\partial \lambda_i(c_i, T_i)}{\partial x} \left( \frac{\partial U_i}{\partial x} + \frac{\partial V_i}{\partial y} \right) + \right. \\
 & \left. + 2 \frac{\partial \mu_i(c_i, T_i)}{\partial x} \frac{\partial U_i}{\partial x} + \frac{\partial \mu_i(c_i, T_i)}{\partial y} \left( \frac{\partial U_i}{\partial y} + \frac{\partial V_i}{\partial x} \right) \right] - (2\lambda_i(c_i, T_i) + 2\mu_i(c_i, T_i))\alpha_T^{(i)} \frac{\partial T_i}{\partial x} = X_i, \\
 & \mu_i(c_i, T_i)\Delta V_i + (\lambda_i(c_i, T_i) + \mu_i(c_i, T_i)) \frac{\partial}{\partial y} \left( \frac{\partial U_i}{\partial x} + \frac{\partial V_i}{\partial y} \right) + \left[ \frac{\partial \lambda_i(c_i, T_i)}{\partial y} \left( \frac{\partial U_i}{\partial x} + \frac{\partial V_i}{\partial y} \right) + \right. \\
 & \left. + 2 \frac{\partial \mu_i(c_i, T_i)}{\partial y} \frac{\partial V_i}{\partial y} + \frac{\partial \mu_i(c_i, T_i)}{\partial x} \left( \frac{\partial U_i}{\partial y} + \frac{\partial V_i}{\partial x} \right) \right] - (2\lambda_i(c_i, T_i) + 2\mu_i(c_i, T_i))\alpha_T^{(i)} \frac{\partial T_i}{\partial y} = Y_i, \\
 & i = \overline{1, n},
 \end{aligned} \tag{1}$$

$$\nabla \cdot (\mathbf{D}_i(c_i, T_i) \nabla c_i) - \mathbf{v}_i \nabla c_i - \gamma_i(c_i - C_m) + \nabla \cdot (\mathbf{D}_T^{(i)} \nabla T_i) = n_p^{(i)} \frac{\partial c_i}{\partial t}, \quad i = \overline{1, n_*(t)}, \quad (2)$$

$$\nabla \cdot (\boldsymbol{\lambda}_T^{(i)} \nabla T_i) - \rho_i c_p^{(i)} \bar{\mathbf{v}}_i \nabla T_i = c_T^{(i)} \frac{\partial T_i}{\partial t}, \quad i = \overline{1, n}, \quad (3)$$

$$\mathbf{v}_i = -\mathbf{K}_i(c_i, T_i) \nabla h_i + \mathbf{v}_c^{(i)}(c_i) \nabla c_i + \mathbf{v}_T^{(i)} \nabla T_i, \quad \operatorname{div} \bar{\mathbf{v}}_i + \frac{\partial n_p^{(i)}}{\partial t} = 0, \quad i = \overline{1, n_*(t)}, \quad (4)$$

$$\mathbf{q}_c^{(i)} = \mathbf{v}_i c_i - \mathbf{D}_i(c_i, T_i) \nabla c_i - \mathbf{D}_T^{(i)} \nabla T_i, \quad i = \overline{1, n_*(t)}, \quad \mathbf{q}_T^{(i)} = \rho c_p \mathbf{v}_i T_i - \boldsymbol{\lambda}_T^{(i)} \nabla T_i, \quad i = \overline{1, n}, \quad (5)$$

$$(\bar{\mathbf{v}}_i, \bar{\mathbf{n}}_i)|_{AB \cup CD} = 0, \quad h_i|_{AD_1} = \tilde{H}_1, \quad h_i|_{BC_1} = \tilde{H}_2, \quad i = \overline{1, n_*(t)}, \quad (6)$$

$$(\bar{\mathbf{q}}_c^{(i)}, \bar{\mathbf{n}}_i)|_{AB \cup CD} = 0, \quad c_i|_{AD_1} = \tilde{C}_1, \quad \left. \frac{\partial c_i}{\partial y} \right|_{BC_1} = 0, \quad i = \overline{1, n_*(t)}, \quad (7)$$

$$(\bar{\mathbf{q}}_T^{(i)}, \bar{\mathbf{n}}_i)|_{AB \cup CD} = 0, \quad T_i|_{AD_1} = \tilde{T}_1, \quad T_i|_{BC_1} = \tilde{T}_2, \quad T_i|_{D_1 D} = \tilde{T}_3, \quad T_i|_{C_1 C} = \tilde{T}_4, \quad i = \overline{1, n}, \quad (8)$$

$$[h_i]|_{\Gamma_i} = 0, \quad [c_i]|_{\Gamma_i} = 0, \quad [\mathbf{v}_i]|_{\Gamma_i} = 0, \quad [\mathbf{q}_c^{(i)}]|_{\Gamma_i} = 0, \quad i = \overline{1, n_*(t) - 1}, \quad (9)$$

$$[T_i]|_{\Gamma_i} = 0, \quad [\mathbf{q}_T^{(i)}]|_{\Gamma_i} = 0, \quad i = \overline{1, n - 1}, \quad (10)$$

$$\mathbf{u}^{(l)} = 0, \quad (x, y) \in AB, \quad \mathbf{u}^{(m)} = 0, \quad (x, y) \in CD, \quad (11)$$

$$\sigma_n^{(i)} = 0, \quad \tau_s^{(i)} = 0, \quad i = \overline{1, n}, \quad (x, y) \in AD_1 \cup D_1 D \cup BC_1 \cup C_1 C, \quad (12)$$

$$[u_n^{(i)}] = [u_s^{(i)}] = 0, \quad [\sigma_n^{(i)}] = [\tau_s^{(i)}] = 0, \quad (x, y) \in \Gamma_i, \quad i = \overline{1, n - 1}, \quad (13)$$

$$h_i(x, y, 0) = \tilde{H}_0^{(i)}(x, y), \quad c_i(x, y, 0) = \tilde{C}_0^{(i)}(x, y), \quad i = \overline{1, n_*(t)}, \quad (14)$$

$$T_i(x, y, 0) = \tilde{T}_0^{(i)}(x, y), \quad i = \overline{1, n}, \quad (15)$$

де  $(x, y) \in \Omega_i$ ,  $i = \overline{1, n}$ ,  $t > 0$ ,  $\mathbf{u}^{(l)} = (U_i, V_i)$ ;  $\tilde{H}_1$ ,  $\tilde{H}_2$ ,  $\tilde{C}_1$ ,  $\tilde{T}_1$ ,  $\tilde{T}_2$ ,  $\tilde{T}_3$ ,  $\tilde{T}_4$ ,  $\tilde{C}_0^{(i)}(x, y)$ ,  $\tilde{T}_0^{(i)}(x, y)$ ,  $\tilde{H}_0^{(i)}(x, y)$  – задані функції,  $n_*(t)$  – номер водонасиченого шару ґрунтового масиву, в якому наявна вільна поверхня;  $\bar{\mathbf{v}}_i = \mathbf{v}_i$  за  $i = \overline{1, n_*(t)}$  та  $\bar{\mathbf{v}}_i = 0$  за  $i = \overline{n_*(t), n}$ .

Тут: (1) – це система рівнянь НДС багатшарового ґрунтового масиву, що складається з областей  $\Omega_i$ ,  $i = \overline{1, n_*(t)}$  повного насичення (фільтрації), а також з областей ґрунтового масиву  $\Omega_i$ ,  $i = \overline{n_*(t), n}$ , що знаходяться в природному стані; (2), (3), (5) описують сольовий та тепловий режими ґрунтового масиву; (4) – це рівняння фільтрації сольових розчинів в неізотермічних умовах у водонасичених шарах ґрунту. Крайові умови та умови спряження ідеального контакту для зміщення, концентрації солей, температури, а також для напору мають вигляд (6)–(15) відповідно.

Для **числового розв'язання** поставленої крайової задачі (1)–(15) використано метод скінчених різниць [11].

Як **приклад, розглянуто двовимірну задачу НДС** ґрунтового масиву прямокутної форми довжиною  $l_1 = 40 \text{ м}$  та товщиною  $l_2 = 20 \text{ м}$ . Вільна поверхня розташована на рівнях  $l_2^1 = 8 \text{ м}$ ,  $l_2^2 = 10 \text{ м}$  та  $l_2^3 = 15 \text{ м}$ .

Час, протягом якого розглянуто процес НДС, становив 3 роки.

Проведено серію числових експериментів, результати яких наведено у вигляді графіків за  $t = 1080 \text{ діб}$ .

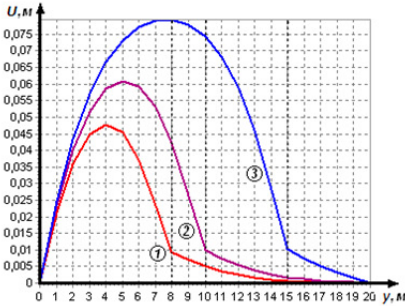


Рис. 2. Графік розподілу зміщень  $U$  вздовж осі  $Oy$  з урахуванням тепло-масоперенесення в площині перерізу  $x = 20 \text{ м}$  за умови розташування вільної поверхні на рівнях ① (8 м), ② (10 м) та ③ (15 м)

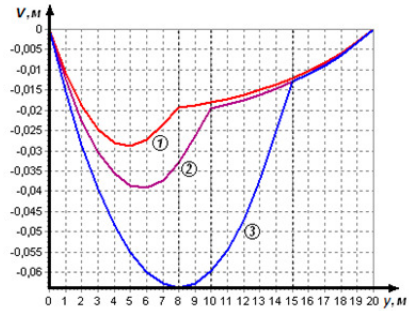


Рис. 3. Графік розподілу зміщень  $V$  вздовж осі  $Oy$  з урахуванням тепло-масоперенесення в площині перерізу  $x = 20 \text{ м}$  за умови розташування вільної поверхні на рівнях ① (8 м), ② (10 м) та ③ (15 м)

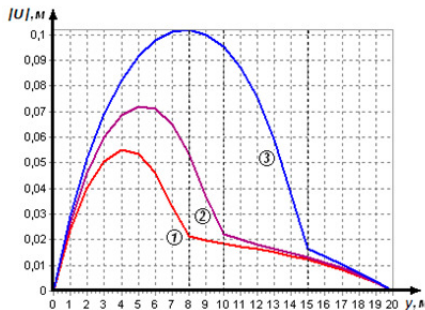


Рис. 4. Графік розподілу модуля вектора зміщень вздовж осі  $Oy$  з урахуванням тепло-масоперенесення в площині перерізу  $x = 20 \text{ м}$  за умови розташування вільної поверхні на рівнях ① (8 м), ② (10 м) та ③ (15 м)

Як видно із графіків, унаслідок збільшення глибини положення вільної поверхні розподіл поля зміщень по осі  $Ox$  та модуля вектора зміщень (рис. 2, рис. 4) значно збільшуються, а по осі  $Oy$  (рис. 3) – зменшуються.

**Висновки.** В роботі сформульовано постановку та проведено математичне моделювання задачі НДС багат шарового ґрунтового масиву з урахуванням тепло-масоперенесення та за наявності рухомої вільної поверхні. Отримано числовий розв'язок поставленої крайової задачі. Проведено та проаналізовано числові експерименти, результати яких наведено у вигляді графіків.

1. Сергиенко И. В. Математическое моделирование и исследование процессов в неоднородных средах [Текст] / И. В. Сергиенко, В. В. Скопецкий, В. С. Дейнека. – К. : Наукова думка, 1991. – 432 с.
2. Власюк А. П. Розв'язування одновимірної стаціонарної задачі НДС ґрунтових масивів в умовах масоперенесення [Текст] / А. П. Власюк, Є. І. Катерина [Текст] // Вісник Львів. ун-ту. Сер. Прикл. матем. та інформ. – 1999. – Вип.1. – С. 50–55.
3. Власюк А. П. Про деякі нові математичні моделі підземної гідромеханіки [Текст] / А. П. Власюк // Вісник РДТУ. – Рівне. – Вип. 3(5), Ч.2. – 2000. – С. 57–62.
4. Кузло М.Т. Про деякі математичні моделі напружено-деформованого стану ґрунтових масивів в процесі руху вільної поверхні ґрунтових вод [Текст] / М. Т. Кузло, І. А. Філатова // Вісник Нац. ун-ту водн. госп-ва та природокористув. – Рівне, 2005. – Вип. 2(30). – С. 282–287.
5. Кузло М. Т. Дослідження впливу концентрації сольових розчинів на деформаційні характеристики ґрунтів [Текст] / М. Т. Кузло, І. А. Філатова // Вісник Нац. ун-ту водн. госп-ва та природокористув. – Рівне, 2006. – Вип. 4(36), Ч.2. – С. 209–214.
6. Філатова І. А. Математичне моделювання НДС багат шарового ґрунтового масиву при наявності рівня ґрунтових вод [Текст] / І. А. Філатова // Вісник Київського ун-ту. Сер. фіз.-мат. науки. – Вип. 1. – 2008. – С. 161–165.
7. Федорчук Н. А. Математичне моделювання одновимірної задачі напружено-деформованого стану ґрунтового масиву при фільтрації сольових розчинів в неізотермічних умовах та наявності рівня ґрунтових вод [Текст] / Н. А. Федорчук // Математичне та комп'ютерне моделювання. Серія: Технічні науки : зб. наук. праць / Ін-т кібернетики ім. В. М. Глушкова Нац. акад. наук України, Кам.-Под. нац. ун-т ім. І. Огієнка. – Кам'янець-Подільський : Кам.-Под. нац. ун-т ім. І. Огієнка, 2010. – Вип. 4 – С. 204–215.
8. Філатова І. А. Математичне моделювання напружено-деформованого стану ґрунтових масивів в складних гідрогеологічних умовах [Текст] / І. А. Філатова, Н. А. Федорчук // Вісник Київського ун-ту. Сер. фіз.-мат. науки. – Вип. 3. – 2011. – С. 218–226.
9. Власюк А. П. Математичне моделювання одновимірної задачі напружено-деформованого стану ґрунтового масиву в умовах тепло-масоперенесення при фільтрації сольових розчинів [Текст] / А. П. Власюк, Н. А. Федорчук // Вісник Київського ун-ту. Сер. фіз.-мат. науки. – Вип. 1. – 2012. – С. 131–136.
10. Коваленко А. Д. Термоупругість [Текст] / А. Д. Коваленко. – К. : Вища школа, 1975. – 216 с.
11. Самарский А. А. Теория разностных схем [Текст] / А. А. Самарский. – М. : Наука, 1989. – 616 с.
12. Власюк А. П. Математичне моделювання консолидації ґрунтів при фільтрації сольових розчинів в неізотермічних умовах : [монографія] [Текст] / А. П. Власюк, П. М. Мартинюк. – Рівне : Нац. ун-т водн. госп-ва та природокористув., 2008. – 416 с.

Демчук О. С., ст. викл., Харів Н. О., ст. викл., Грицюк О. В., студент 5 курсу ННІ АКOT (Національний університет водного господарства та природокористування, м. Рівне)

## **ОЦІНКА ЕКОЛОГІЧНОГО СТАНУ БАСЕЙНУ РІЧКИ ЖОВТА ЗА ДОПОМОГОЮ ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ**

**Проведено аналіз екологічного стану басейну малої річки Жовта за допомогою вдосконаленої системної моделі, реалізованої в інформаційно-аналітичній системі оцінки стану басейнів малих річок України.**

**Проведен анализ экологического состояния бассейна малой реки Желтая с помощью усовершенствованной системной модели, реализованной в информационно-аналитической системе оценки состояния бассейнов малых рек Украины.**

**The analysis of the ecological state of the basin of small river Zhovta has been held with the implementation of the improved system model, realized in the information-analytical system assessment of the basins of small rivers in Ukraine.**

**Постановка задачі.** Системну логіко-математичну модель оцінки антропогенного навантаження і класифікації стану малих річок [1, 2] удосконалено шляхом розроблення моделей нових підсистем [5, 6], а також методів і критеріїв прийняття рішень в наявних підсистемах [3, 4]. Таку модель реалізовано на практиці у вигляді програмного комплексу “Інформаційно-аналітична система оцінки екологічного стану басейнів малих річок України” [7], що автоматизує процес розрахунку антропогенного навантаження і класифікації стану басейнів малих річок.

За допомогою інформаційно-аналітичної системи проведено оцінку екологічного стану басейну малої річки Жовта за показниками чотирьох основних підсистем: “Радіоактивне забруднення”, „Використання земель”, “Використання річкового стоку” та „Якість водних ресурсів”. Актуальність такого дослідження зумовлена зростаючим негативним впливом комплексу природних та антропогенних факторів, що спричиняє погіршення якості води в річках Жовта та Інгулець за низкою показників, серед яких – сумарна  $\beta$ -активність. Одним із найважливішим серед названих чинників є безконтрольне затоплення вироблених горизонтів шахти «Нова» в м. Жовті Води, з якої видобувають уранову, залізну та комплексні руди. У разі повного затоплення шахти стає реальною загроза радіохімічного забруднення р. Жовта, що, в свою чергу, може призвести до забруднення Карачунівського

водосховища – основного джерела водопостачання регіону, в тому числі м. Кривий Ріг.

Для прийняття завчасних управлінських рішень щодо експлуатації водозабору на р. Жовта та Карачунівському водосховищі, а також для поліпшення якісного стану вод необхідним є вдосконалення системи моніторингу і в руслі екосистемного підходу здійснення комплексної оцінки екологічного стану води, повітря та ґрунтів басейну малої річки. З використанням удосконаленої системної моделі [5], реалізованої в інформаційно-аналітичній системі, нами проведено оцінку екологічного стану басейну малої річки Жовта за даними спостережень за 2011 рік.

**Підсистема “Радіоактивне забруднення”.** Спочатку визначено стан радіоактивного забруднення земель басейну р. Жовта, який за трьома наявними показниками системної моделі (цезій, стронцій, плутоній) визнано “задовільним” (табл. 1).

Таблиця 1  
Оцінка стану підсистеми “Радіоактивне забруднення земель” у басейні р. Жовта (2011 р.)

Назва показника	Значення показника, (Кі/км <sup>2</sup> )	Оцінка	Категорія
Цезій	0,007	0	Задовільний
Стронцій	0,08	0	Задовільний
Плутоній	0	0	Задовільний

Таблиця 2  
Оцінка стану підсистеми “Використання земель” на основі нечіткої логічної моделі.

Природна сільськогосподарська провінція – Степова правобережна

Назва показника	Значення показника, (%)	Рівень використання	Міра, $x_k$	Вагові коефіцієнти, $\alpha_k$	$\mu_j^k$
Лісистість, %	2	значний	-4	0,2	-
Ступінь природного стану, %	<20	значний	-4	0,1	-
Сільськогосподарська освоєність, %	84	низький	1	0,1	0,29
	87	вище норми	-1		0,71
Розораність, %	72	вище норми	-1	0,1	-
	73				
Урбанізація, %	17	значний	-4	0,2	-
Еродованість, %	6	низький	1	0,3	-
	8				



**Підсистема “Використання земель”.** На другому етапі оцінено стан підсистеми “Використання земель” (табл. 2). Низку параметрів, як-от лісистість, ступінь природного стану та урбанізація, визначено на основі картографічного методу за допомогою ГІС MapInfo Professional 9.0 (рис. 1). Інші параметри (сільськогосподарська освоєність, розораність, еродованість) встановлено на основі статистичних даних. Деякі з них мають нечітке представлення. Наприклад, частина басейну малої річки ( $\approx 71\%$ ) має сільськогосподарську освоєність вище норми, а інша ( $\approx 29\%$ ) – низького рівня.

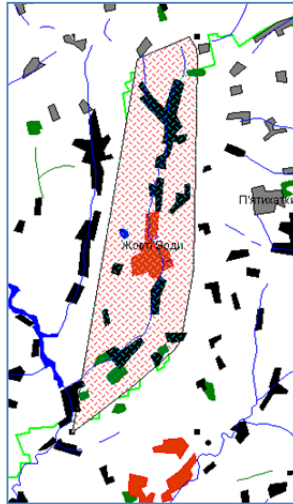


Рис. 1. Басейн річки Жовта, виконаний засобами ГІС MapInfo Professional 11.5, де  
■ – міста ■ – селища ■ – ліси

Із використанням узагальнення детермінованої моделі в разі нечіткого задання вхідних показників обчислюємо значення функції ризиків-корисностей та функції ризиків за формулами [4]:

$$H_i = \frac{\sum_{j=1}^5 \sum_{k=1}^n \mu_j^k \alpha_k \varphi_k(U_j)}{\sum_{j=1}^5 \sum_{k=1}^n \mu_j^k \alpha_k}, \quad (1)$$

$$H_i^{(-)} = \frac{\sum_{j=1}^5 \sum_{k=1}^{n_k} \mu_j^k \alpha_k \varphi_k^{(-)}(U_j)}{\sum_{j=1}^5 \sum_{k=1}^n \mu_j^k \alpha_k}, \quad (2)$$

де  $n$  – кількість наявних показників з позитивними і негативними значеннями формалізованих функцій;  $n_k$  – кількість показників з негативними значеннями

функцій ризиків;  $\alpha_k$  – ваговий коефіцієнт  $k$ -го показника, який відображає важливість останнього залежно від природно-сільськогосподарської зони і провінції;  $\mu_j^k$  – функція належності  $k$ -го показника інтервалу  $[\lambda_{j,i}^k; \lambda_{j,i+1}^k]$ ;  $i$  – номер річки.

За значеннями  $H_i = -1,76$  і  $H_i^{(-)} = -2,1$  стан підсистеми визнано “незадовільним”. Функція міри підсистеми:  $\varphi(L_4) = -4$ .

**Підсистема “Використання річкового стоку”.** Третій крок алгоритму – визначення стану підсистеми “Використання річкового стоку”. Останній є “добрим”, оскільки  $H_i = 3$  (табл. 3). Функція міри підсистеми:  $\varphi(W_1) = 3$ .

Таблиця 3

Оцінка стану підсистеми “Використання річкового стоку”

Назва показника	Значення показника (%)	Рівень використання	Міра, $U_k$	Вагові коефіцієнти, $\beta_k$
Повне використання річкового стоку, %	0,32	низький	3	0,1
Безповоротне водоспоживання, %	0,32	низький	3	0,2
Скид води, %	3,95	низький	3	0,3
Скид забруднених стічних вод, %	0,01	низький	3	0,4

**Підсистема “Якість води”.** Розглянуто підсистему “Якість води” за трьома блоками показників. За результатами оцінювання блоку сольового складу води річки визнано солонуватими  $\beta$ -мезогалинними (табл. 4), значення блокового індексу  $I_1 = 2$ .

Таблиця 4

Оцінка блоку сольового складу підсистеми “Якість води”

Назва показника	Значення показника, (мг/дм <sup>3</sup> )	Клас	Категорія
Сума іонів	1805	II	2
Хлориди	305	II	2
Сульфати	558	II	2
Мінералізація	1,8	Солонуваті	$\beta$ -мезогалинні

Наступний етап – стан блоку трофо-сапробіологічних показників (табл. 5). Значення блокового індексу  $I_2 = 4,7$ . Аналогічно в результаті оцінки блоку специфічних показників токсичної дії (табл. 6) отримали значення блокового індексу  $I_3 = 4,1$ .

Значення інтегрального екологічного індексу ( $I_e=3,6$ ) визначено за формулою [3]:

$$I_E = \frac{\{I_1 + I_2 + I_3\}}{3} \quad (3)$$

Таблиця 5

Оцінка блоку трофо-сапробіологічних показників підсистеми “Якість води”

Назва показника	Значення показника, (мг/дм <sup>3</sup> )	Клас	Категорія
Завислі речовини	20,55	II	3
РН	8,1	I	1
Азот амонійний	0,8	III	5
Азот нітритний	0,14	V	7
Азот нітратний	24	V	7
Фосфор фосфатів	1,08	V	7
Розчинений кисень	8,9	I	1
% насичення	30,5	V	7
БСК5	4,5	III	5

Таблиця 6

Оцінка блоку специфічних показників токсичної дії підсистеми “Якість води”

Назва показника	Значення показника (мкг/дм <sup>3</sup> )	Клас	Категорія
<b>Мідь</b>	18	III	5
<b>Цинк</b>	14	II	2
<b>Хром</b>	23	III	5
<b>Залізо</b>	130	III	4
<b>Нафтопродукти</b>	300	IV	6
<b>СПАР</b>	60	III	5
<b>Нікель</b>	7	II	3
<b>Кадмій</b>	5	IV	6
<b>Марганець</b>	15	II	2
<b>Свинець</b>	7	II	3

Стан підсистеми “Якість води” визначено як води «чисті», перехідні за якістю від «досить чистих» до «задовільних», «слабко забруднених» (субкатегорія якості води – 3–4). Функція міри цієї підсистеми:  $\varphi(Q_2) = I$ .

На підставі поточних значень мір кожної підсистеми визначено числову міру класу всієї системи басейну річки (індукційний коефіцієнт антропогенного навантаження) за формулою:

$$\varphi(K_n) = 0,3\varphi(L_n) + 0,2\varphi(W_n) + 0,5\varphi(Q_n), \quad (4)$$

де  $L_n$ ,  $W_n$ ,  $Q_n$  – поточні класи станів підсистем “Використання земель”, “Використання річкового стоку”, “Якість води”, “Оцінки забруднення повітря і ґрунтів” відповідно;  $K_n$  – клас усієї системи басейну річки.

**Висновки.** За отриманим на основі проведення екологічної оцінки значенням  $КАН=-0,1$  в 2011 році стан басейну річки Жовта визнано “задовільним”, утім, без урахування показників радіоактивного забруднення донних відкладень, які часто є вищими за ГДК.

1. Методика розрахунку антропогенного навантаження і класифікації екологічного стану басейнів малих річок України / А. В. Яцик, Л. Б. Бишовець, О. М. Петрук, А. П. Чернявська. – К., 2007. – 71 с. 2. Методика екологічної оцінки якості поверхневих вод за відповідними категоріями / В. Д. Романенко, В. М. Жукинський, А. В. Яцик та ін. – К. : СИМВОЛ-Т, 1998. – 28 с. 3. Ковальчук П. І. Система аналізу антропогенного навантаження і класифікації екологічного стану малих річок України / П. І. Ковальчук, О. С. Демчук, О. М. Стаднічук // Вісник НУВГП. – Рівне, 2005. – Вип. 3 (31). – С. 36–43. 4. Ковальчук П. І. Прийняття рішень в підсистемі “Використання земель” інформаційно-аналітичної системи оцінки екологічного стану малих річок в умовах невизначеності та ризику / П. І. Ковальчук, О. С. Демчук, О. В. Бобер // Вісник НУВГП. – Рівне, 2007. – Вип. 4(40) (присвячений 85-річчю НУВГП). Ч.1. – С. 265–270. 5. Ковальчук П. І. Методи оцінки ризиків в інформаційній системі аналізу оцінки екологічного стану басейну малої ріки / П. І. Ковальчук, О. С. Демчук // Математичне та комп’ютерне моделювання : зб. наукових праць Кам’янець-Подільського національного університету ім. Івана Огієнка. Серія: технічні науки. – Кам’янець-Подільський : Кам’янець-Подільський національний університет ім. Івана Огієнка, 2012. – Вип. 7. – С. 69–76. 6. Демчук О. С. Моделювання ризиків затоплення земель в басейні малої ріки / О. С. Демчук // Вісник НУВГП. – Рівне, 2009. – Вип. 3 (47), Ч.1. – С. 290–296. 7. Демчук О. С. Інформаційно-аналітична система оцінювання екологічного стану басейнів малих річок / О. С. Демчук // Меліорація і водне господарство. – К., 2012. – Вип. 99. – С. 257–266. 8. Бідоконь Г. С. Забруднення радіонуклідами р. Інгулець та її приток / Бідоконь Г. С., Лаврова Т. В., Дворецький А. І. – Дніпропетровськ, 2007 – С. 1–10. 9. Шапара А. Г. Екологічний атлас Дніпропетровської області / А. Г. Шапара. – Дніпропетровськ : ПП «Моноліт», 2009. – 66 с.

Остапчук О. П., к.т.н., ст. викл., Мартинчук А. І., студент 5 курсу  
ННІ АКОТ (Національний університет водного господарства та  
природокористування, м. Рівне)

## МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ПЕРЕНЕСЕННЯ СОЛЬОВИХ РОЗЧИНІВ ПРИ ФІЛЬТРАЦІЇ ПІДЗЕМНИХ ВОД ЧЕРЕЗ ЗЕМЛЯНУ ДАМБУ З ВІЛЬНОЮ ПОВЕРХНЕЮ

Побудовано математичну модель процесу перенесення сольових розчинів при фільтрації підземних вод через земляну дамбу з вільною поверхнею. Розроблено обчислювальний алгоритм розв'язання відповідної задачі. Наведено результати числових експериментів.

Построено математическую модель процесса переноса солевых растворов при фильтрации подземных вод через земляную дамбу со свободной поверхностью. Разработан вычислительный алгоритм решения соответствующей задачи. Приведены результаты численных экспериментов.

The mathematical model of process to transfer salt liquid during the filtration the earth's water through earthen dike with free surface was constructed. It was worked out the calculating algorithm of solving the certain problem. The results of numerical experiments are given.

Розв'язування багатьох практичних задач підземної гідромеханіки, аеромеханіки, тепло-масоперенесення передбачає використання математичних моделей, основу яких складають змішані крайові задачі для рівнянь параболічного типу в областях складної геометричної форми. З огляду на це видається актуальним розроблення обчислювального алгоритму розв'язання змішаних крайових задач для рівнянь параболічного типу в таких областях.

Розглянуто задачу перенесення сольових розчинів під час їхньої фільтрації на земляному відкосі (рис. 1) [4, 7]. Нижню межу ґрунту вважаємо непроникною. Відомі п'езометричні напори у лівому та правому басейнах відповідно  $H_1$  та  $H_2$  ( $H_2 > H_1$ ). Фільтрація сольового розчину відбувається за наявності вільної поверхні, на якій задано умови  $h|_{\Gamma_2} = y, \frac{\partial c}{\partial n}|_{\Gamma_2} = 0$ .

Крім того, відомий розподіл концентрації сольового розчину в початковий момент часу  $c(x, y, 0) = \tilde{c}_0(x, y)$ .

Необхідно розрахувати поле концентрації сольового розчину  $c(x, y, t)$  за відомого розподілу на межі області фільтрації й у початковий момент часу, а також дослідити розподіл концентрації сольових розчинів по області фільтрації з часом.

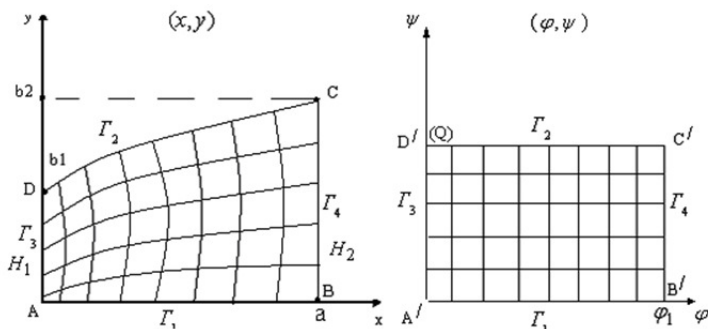


Рис. 1. Масоперенесення сольових розчинів на земляному відкосі у фізичній та фіктивній областях

Математичну модель поставленої задачі у фізичній області описано крайовою задачею:

$$D \left( \frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} \right) - V_x \frac{\partial c}{\partial x} - V_y \frac{\partial c}{\partial y} - \gamma(c - C_*) = \sigma \frac{\partial c}{\partial t}, \quad (1)$$

$$V_x = -k \frac{\partial h}{\partial x}, \quad V_y = -k \frac{\partial h}{\partial y}, \quad \text{div} \vec{V} = \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} = 0, \quad (2)$$

$$h|_{AD} = h(0, y) = H_2, \quad \frac{\partial h}{\partial n} \Big|_{AB} = \frac{dh(x, 0)}{dy} = 0, \quad (3)$$

$$h|_{BC} = h(a, y) = H_1, \quad h|_{CD} = y, \quad \frac{\partial h}{\partial n} \Big|_{CD} = 0, \quad (4)$$

$$c(x, y, 0) = \tilde{C}_0(x, y), \quad \frac{\partial c}{\partial n} \Big|_{AB} = \frac{\partial c(x, 0, t)}{\partial y} \Big|_{AB} = 0, \quad (5)$$

$$c|_{BC} = c(a, y, t) = \tilde{C}_1(y, t), \quad (6)$$

$$\frac{\partial c}{\partial n} \Big|_{CD} = \frac{\partial c(x, y, t)}{\partial n} \Big|_{CD} = 0, \quad \frac{\partial c}{\partial n} \Big|_{AD} = \frac{\partial c(0, y, t)}{\partial x} = 0. \quad (7)$$

Математична модель задачі (1)–(7) в області комплексного потенціалу  $(\varphi, \psi)$  має вигляд:

$$\frac{\partial^2 c}{\partial \varphi^2} + \frac{\partial^2 c}{\partial \psi^2} - \frac{1}{D} \cdot \frac{\partial c}{\partial \varphi} - \frac{\gamma}{D \cdot V^2} \cdot (c - C_*) = \frac{\sigma}{D \cdot V^2} \cdot \frac{\partial c}{\partial t}, \quad (8)$$

$$c(\varphi, \psi, 0) = \tilde{C}_0(\varphi, \psi), \quad \frac{\partial c}{\partial \psi} \Big|_{AB} = \frac{\partial c(\varphi, 0, t)}{\partial \psi} \Big|_{AB} = 0, \quad (9)$$

$$c|_{BC} = c(\varphi_1, \psi, t) = \tilde{C}_1(\psi, t), \quad (10)$$

$$\left. \frac{\partial c}{\partial \psi} \right|_{CD} = \left. \frac{\partial c(\psi, \varrho, t)}{\partial \psi} \right|_{CD} = \left. \frac{\partial c}{\partial \varphi} \right|_{AD} = \left. \frac{\partial c(0, \psi, t)}{\partial \varphi} \right|_{AD} = 0.$$

(11) Унаслідок розв'язання задачі фільтрації (2)–(4) знайдено компоненти та квадрат швидкості фільтрації за формулами:

$$V^2(\varphi, \psi) = V_x^2 + V_y^2, \quad (12)$$

$$V_x \approx \frac{\Delta \varphi}{\Delta x} = \frac{h_1}{\frac{1}{2} \cdot (x_{i+1,j} - x_{i-1,j})} = \frac{2h_1}{x_{i+1,j} - x_{i-1,j}}, \quad (13)$$

$$V_y \approx \frac{\Delta \varphi}{\Delta y} = \frac{h_2}{\frac{1}{2} \cdot (y_{i,j+1} - y_{i,j-1})} = \frac{2h_2}{y_{i,j+1} - y_{i,j-1}}. \quad (14)$$

Згідно з ЛОС Самарського задачу (8)–(11) потрібно розділити на дві одновимірні підзадачі. Крайова задача для поздовжньої прогонки має вигляд:

$$DV^2(\varphi, \psi) \frac{\partial^2 c}{\partial \varphi^2} - V^2(\varphi, \psi) \frac{\partial c}{\partial \varphi} - \frac{1}{2} \gamma(c - C_*) = \frac{\sigma}{2} \frac{\partial c}{\partial t}, \quad (15)$$

$$c(\varphi, \psi, 0) = \tilde{C}_0(\varphi, \psi), \quad c(\varphi_1, \psi_1, t) = \tilde{C}_1(\psi, t), \quad (16)$$

$$\frac{\partial c(0, \psi, t)}{\partial \varphi} = 0. \quad (17)$$

Крайову задачу для поперечної прогонки потрібно записати так:

$$DV^2(\varphi, \psi) \frac{\partial^2 c}{\partial \psi^2} - \frac{1}{2} \gamma(c - C_*) = \frac{\sigma}{2} \frac{\partial c}{\partial t}, \quad (18)$$

$$c(\varphi, \psi, 0) = \tilde{C}_0(\varphi, \psi), \quad \frac{\partial c(\varphi, 0, t)}{\partial \psi} = \frac{\partial c(\varphi, \varrho, t)}{\partial \psi} = 0. \quad (19)$$

Для розв'язання задачі (15)–(17) використали монотонну різницеву схему:

$$\begin{aligned} & \mu \frac{C_{i-1j}^{k+1/2} - 2C_{ij}^{k+1/2} + C_{i+1j}^{k+1/2}}{h^2} + r_+ \frac{C_{i+1j}^{k+1/2} - C_{ij}^{k+1/2}}{h} + \\ & + r_- \frac{C_{ij}^{k+1/2} - C_{i-1j}^{k+1/2}}{h} - \frac{0.5\gamma(C_{ij}^{k+1/2} - C_*)}{DV_{ij}^2} = \end{aligned} \quad (20)$$

$$= \frac{0.5\sigma}{DV_{ij}^2} \frac{C_{ij}^{k+1/2} - C_{ij}^k}{0.5\tau},$$

$$C_{ij}^0 = \tilde{C}_{0ij}, \quad C_{nj}^{k+1/2} = \tilde{C}_{nj}, \quad C_{0j}^{k+1/2} = C_{1j}^{k+1/2}.$$

Запишемо (20) у прогоночному вигляді:

$$\begin{cases} a_1^1 C_{i-1j}^{k+1/2} - c_1^1 C_{ij}^{k+1/2} + b_1^1 C_{i+1j}^{k+1/2} = -f_1^1, \\ C_{0j}^{k+1/2} = k_1^1 C_{1j}^{k+1/2} + \eta_1^1, \\ C_{nj}^{k+1/2} = k_2^1 C_{n-1j}^{k+1/2} + \eta_2^1, \end{cases} \quad (21)$$

де

$$a_1^1 = \frac{\mu}{h^2} - \frac{r_-}{h}, \quad b_1^1 = \frac{\mu}{h^2} + \frac{r_+}{h},$$

$$c_1^1 = \frac{2\mu}{h^2} + \frac{r_+}{h} - \frac{r_-}{h} + \frac{\gamma}{2DV^2}, \quad f_1^1 = \frac{\gamma C_*}{2DV^2} + \frac{\sigma C_{ij}^k}{D\tau V^2}.$$

Розв'язок (21) знаходимо за допомогою методу прогонки

$$C_{ij}^{k+1/2} = a_{i+1j}^1 C_{i+1j}^{k+1/2} + \beta_{i+1j}^1, \quad (22)$$

де

$$\alpha_{i+1j}^1 = \frac{b^1}{c_1^1 - \alpha_{ij}^1}, \quad \beta_{i+1j}^1 = \frac{a^1 \beta_{ij}^1 + f_1^1}{c_1^1 - \alpha_{ij}^1 a_1^1},$$

$$\alpha_{1j}^1 = k_1^1 = 1, \quad \beta_{1j}^1 = \eta_1^1 = 0.$$

Для розв'язання задачі (18)–(19) використано неявну різницеву схему:

$$\frac{C_{ij-1}^{k+1} - 2C_{ij}^{k+1} + C_{ij+1}^{k+1}}{h^2} - \frac{0.5\gamma(C_{ij}^{k+1} - C_*)}{DV_{ij}^2} =$$

$$= \frac{0.5\sigma C_{ij}^{k+1} - C_{ij}^{k+1/2}}{DV_{ij}^2 \cdot 0.5\tau}, \quad (23)$$

$$C_{ij}^0 = \tilde{C}_{0ij}, \quad C_{0j}^{k+1} = C_{1j}^{k+1}, \quad C_{im}^{k+1} = C_{im-1}^{k+1}.$$

Запишемо різницеву схему (23) у прогоночному вигляді:

$$\begin{cases} a_2^1 C_{1ij-1}^{k+1} - c_2^1 C_{1ij}^{k+1} + b_2^1 C_{1ij+1}^{k+1} = -f_2^1, \\ C_{i0}^{k+1} = k_3^1 C_{i1}^{k+1} + \eta_3^1, \\ C_{im}^{k+1} = k_4^1 C_{im-1}^{k+1} + \eta_4^1, \end{cases} \quad (24)$$

де

$$a_2^1 = \frac{1}{h^2}, \quad b_2^1 = \frac{1}{h^2}, \quad c_2^1 = \frac{2}{h^2} + \frac{\gamma}{2DV^2} + \frac{\sigma}{D\tau V^2},$$

$$f_2^1 = \frac{\gamma C_*}{2DV^2} + \frac{\sigma C_{ij}^{k+1/2}}{D\tau V^2}.$$



Розв'язок (24) знаходимо за допомогою методу прогонки:

$$C_{ij}^{k+1} = a_{ij+1}^2 C_{ij+1}^{k+1} + \beta_{ij+1}^2, \quad (25)$$

де

$$\alpha_{ij+1}^2 = \frac{b_2^1}{c_2^1 - \alpha_{ij}^2 a_2^1}, \quad \beta_{ij+1}^2 = \frac{a_2^1 \beta_{ij}^3 + f_2^1}{c_2^1 - \alpha_{ij}^2 a_2^1},$$

$$\alpha_{i1}^2 = k_3^1 = 1, \quad \beta_{i1}^2 = \eta_3^1 = 0, \quad C_{im}^{k+1} = \frac{\beta_{im}^2}{1 - \alpha_{im}^2}.$$

Наведені вище алгоритми програмно реалізовано з використанням сучасної програмної платформи .NET Framework та мови програмування С#. На основі такої програмної реалізації було проведено низку числових експериментів із такими вхідними даними:

$$n=30; m=15; a = 30 \text{ м.}; b_1 = 15 \text{ м.}; b_2 = 25 \text{ м.}; \tau=33 \text{ доби}; \sigma=0,2;$$

$$\gamma=0,0085 \text{ доба}^{-1}; Dm=0, 1 \text{ м}^2/\text{добу}; c_* = 2 \text{ кг/м}^3; \check{C}_1=30 \text{ кг/м}^3; \check{C}_2=0 \text{ кг/м}^3.$$

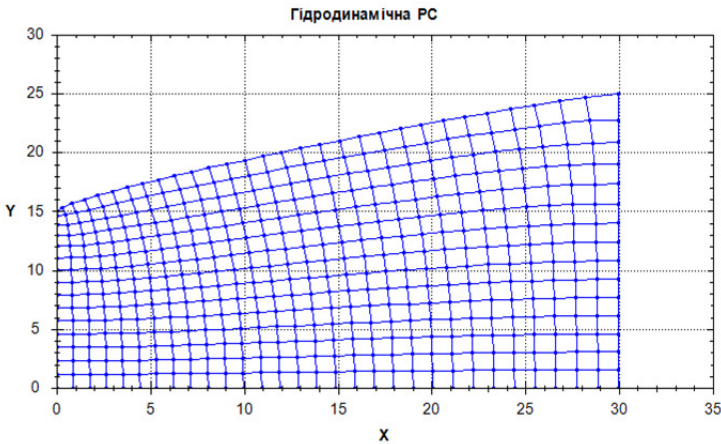


Рис. 2. Гідродинамічна різницева сітка фільтраційного потоку

Шляхом побудови числового конформного відображення [1, 3] отримано гідродинамічну різницеву сітку фільтраційного потоку (див. рис. 2), досліджено зміну поля концентрації сольового розчину (рис. 3, рис. 4). На основі проведених числових експериментів вивчено вплив параметрів масоперенесення (пористості, коефіцієнтів конвективної дифузії та масообміну) на розподіл концентрації солевих розчинів (рис. 5 – рис. 7). Також простежено динаміку зміни концентрації сольового розчину по області фільтрації з часом (рис. 8).

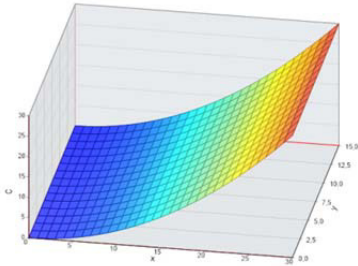


Рис. 3. Просторове зображення поля концентрації соляного розчину за  $t = 0$

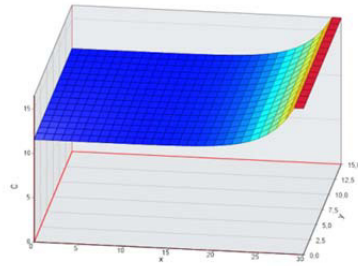


Рис. 4. Просторове зображення поля концентрації соляного розчину за  $t = 495$  діб

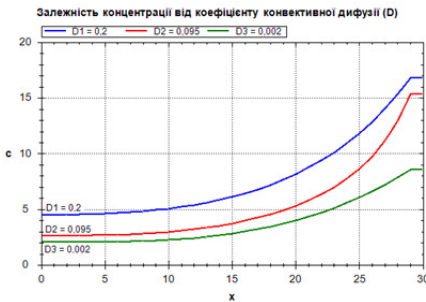


Рис. 5. Залежність концентрації соляного розчину від коефіцієнта конвективної дифузії

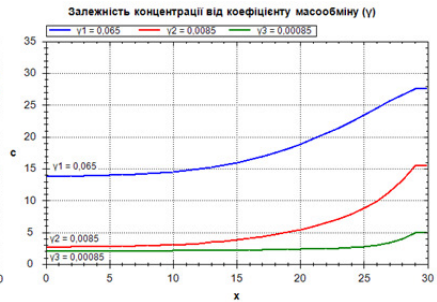


Рис. 6. Залежність концентрації соляного розчину від коефіцієнта масообміну

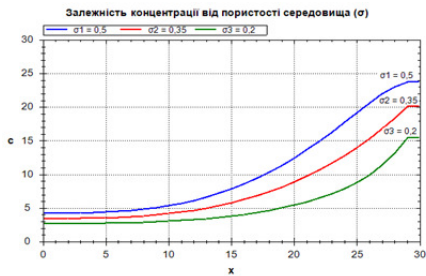


Рис. 7. Залежність концентрації соляного розчину від пористості середовища

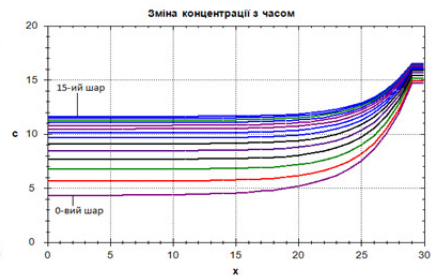


Рис. 8. Динаміка зміни концентрації соляного розчину з часом

У результаті проведеного математичного і комп'ютерного моделювання досліджено динаміку зміни концентрації соляних розчинів під час фільтрації

підземних вод через земляну дамбу з вільною поверхнею та залежність процесу перенесення сольових розчинів від різних параметрів масоперенесення, що дає підстави зробити такі висновки:

- концентрація сольового розчину перерозподіляється по області фільтрації з часом, а в разі виходу процесу на стаціонарний режим стабілізується на певній величині;
- унаслідок збільшення значень коефіцієнтів конвективної дифузії та масообміну концентрація сольових розчинів збільшується;
- чим більша пористість середовища, тим більшою є концентрація сольових розчинів у ньому.

Отже, під час математичного моделювання процесу масоперенесення речовин потрібно враховувати залежність параметрів фільтраційного потоку від його фізико-хімічних властивостей та властивостей пористого середовища.

1. Вабищевич П. Н. Численные методы решения задач со свободной границей / П. Н. Вабищевич. – М. : МГУ, 1987. – 164 с. 2. Власюк А. П. Застосування конформних і квазиконформних відображень до чисельної автоматичної побудови різницевих сіток / А. П. Власюк, В. Г. Михальчук // Вибрані питання математичного аналізу, алгебри і геометрії : зб. наук. праць. – Львів-Рівне, 1991. – С. 39–48. 3. Власюк А. П. Застосування чисельних конформних відображень до розв'язання нестационарної задачі фільтрації підземних вод / А. П. Власюк, О. П. Остапчук // Прогнозування та прийняття рішень в умовах невизначеності : тези доп. Міжн. конф. – Київ-Рівне, 2008. – С. 250–252. 4. Власюк А. П. Чисельне моделювання переносу сольових розчинів в основі ґрунтової греблі / А. П. Власюк, О. П. Остапчук // Сучасні проблеми прикладної математики та інформатики : тези доп. Всеукр. наук. конф. – Львів, 2004. – С. 33. 5. Власюк А. П. Чисельне моделювання переносу сольових розчинів в основах гідротехнічних об'єктів / А. П. Власюк, О. П. Остапчук // Прогнозування та прийняття рішень в умовах невизначеності : тези доп. Міжн. школи-семінару. – Тернопіль, 2004. – С. 217–219. 6. Власюк А. П. Чисельне розв'язання одного класу задач з вільними межами в криволінійних чотирикутниках / А. П. Власюк, В. Г. Михальчук // Нові підходи до розв'язання диф. р-нь: тези доп. Всеукраїнської наукової конф. – Дрогобич, 25–27 січня 1994 р. – К., 1994. – 102 с. 7. Власюк А. П. Числове моделювання масопереносу сольових розчинів під гідротехнічними спорудами / А. П. Власюк, О. П. Остапчук // Вісник НУВГП. – Рівне, 2006. – Вип. 4(36). Ч. 2. – С. 30–38.

Білецький А. І., к.ф.-м.н., доцент, Чуй Ю. В., студент 4 курсу ННІ АКОТ (Національний університет водного господарства та природокористування, м. Рівне)

### **ЗАСТОСУВАННЯ АЛГЕБР С. ЛІ ДЛЯ ДЕЯКИХ КЛАСІВ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ**

**Побудовано оператори ad та Ad для деяких класів диференціальних рівнянь. Наведено низку прикладів. Знайдено розв'язок диференціального рівняння за допомогою методу Лі.  
Ключові слова: алгебра Лі, метод Лі, солітонна теорія.**

**Построены операторы ad и Ad для некоторых классов дифференциальных уравнений. Приведен ряд примеров. Найдено решение дифференциального уравнения методом Ли.  
Ключевые слова: алгебра Ли, метод Ли, солитонная теория.**

**Constructed ad and Ad operators for some classes of differential equations. Shows some examples. Found the solution of differential equations by Lie.  
Key words: Lie algebra, Lie method, soliton theory.**

Елементарна теорія звичайних диференціальних рівнянь, за висловом Олвера [3], – це «багатовид спеціальних методів, призначених для розв'язування деяких часткових типів рівнянь». Застосування алгебр та груп Лі для розв'язування диференціальних рівнянь не всім відоме.

Упродовж останніх років алгебри та групи Лі набувають усе більш актуальності у середовищі науковців – і зарубіжних, і вітчизняних: в Україні цією проблемою займається школа В. І. Фушича [6]; у США передові позиції належать школі Спенсера [1], [3]; у Франції – школі колективу Бурбакі [2]. У НУВГП працює семінар «Алгебри Лі» [5].

Найважливіше досягнення теорії С. Лі полягає в тому, що для будь-яких векторних просторів виконуються закони:

1.  $[A, B] = AB - BA$ . (Дужка Лі)
2.  $[A, [B, C]] + [B, [C, A]] + [C, [A, B]] = 0$ .

На основі цих положень вибудовано вивчення у солітонній теорії нового класу диференціальних рівнянь виду  $\frac{\partial u}{\partial t} = [L, B]$  [4], які не охоплено класичною теорією.

Для цілого класу алгебр Лі введено універсальну операцію диференціювання:

$$D(x \cdot y) = Dx \cdot y + x \cdot Dy.$$

Центральна теорема. Множина  $Der(A)$  всіх диференціювань алгебри  $A$  є алгеброю Лі з операцією  $[D_1, D_2] = D_1D_2 - D_2D_1$  [2].

На основі цієї теореми маємо операцію  $ad$ :

$$ad x(y) = [x, y].$$

У монографії [3] показано, що відображення  $ad x$  є диференціюванням алгебри  $A$ .

На ґрунті алгебри Лі відомо, що клас диференціальних рівнянь має певну алгебру. Проілюструємо це положення прикладами.

*Приклад 1.* Дано  $y_x = ky$ . Зрозуміло, що  $y = ce^{kx}$ . З іншого боку, відомо [1], що векторний простір  $x, y, y_x$  має розмірність  $\dim R = 3$ . Цей вектор має математичну модель:

$$\bar{V} = \xi(x, y) \frac{\partial}{\partial x} + \varphi(x, y) \frac{\partial}{\partial y} + \varphi^x(x, y) \frac{\partial}{\partial y_x} \quad (I)$$

Для отримання розв'язку диференціального рівняння за допомогою методу Лі потрібно подіяти оператором (I) на це диференціальне рівняння. Остаточоно отримуємо:

$$V_1 = \frac{\partial}{\partial x};$$

$$V_2 = e^{kx} \frac{\partial}{\partial y};$$

$$V_3 = e^{kx} \frac{\partial}{\partial x} + kye^{kx} \frac{\partial}{\partial y};$$

$$[V_1, V_2] = kV_2;$$

$$[V_1, V_3] = kV_3;$$

$$[V_2, V_3] = 0.$$

Вектор  $V_1$  інформує, що, крім розв'язку  $y = ce^{kx}$ , функція  $\tilde{y} = ce^{k(x-\varepsilon)}$  є теж розв'язком.

*Приклад 2.* Дано  $y_{xx} = k^2y$ . Зрозуміло, що  $y = c_1e^{kx} + c_2e^{-kx}$ .

Вектор  $\bar{V}$  має модель:  $\bar{V} = \xi \frac{\partial}{\partial x} + \varphi \frac{\partial}{\partial y} + \varphi^{xx} \frac{\partial}{\partial y_{xx}}$  (II).

Для  $\varphi^{xx} = D_{xx}(\varphi - \xi y_x) + \xi y_{xxx}$ . [3]

Після нескладних обчислень отримуємо:

$$V_1 = \frac{\partial}{\partial x};$$

$$V_2 = e^{kx} \frac{\partial}{\partial y};$$

$$V_3 = kye^{-kx} \frac{\partial}{\partial y};$$

$$V_4 = e^{2kx} \frac{\partial}{\partial x} + kye^{2kx} \frac{\partial}{\partial y};$$

$$V_5 = e^{-2kx} \frac{\partial}{\partial x} - kye^{2kx} \frac{\partial}{\partial y}.$$

Обчислено 2 таблиці [3].

	$V_1$	$V_2$	$V_3$	$V_4$	$V_5$
$V_1$	0	$kV_2$	$-kV_3$	$2kV_4$	$-2kV_5$
$V_2$	$-kV_2$	0	0	0	$-2kV_3$

$V_3$	$kV_3$	0	0	$2kV_3$	0
$V_4$	$-kV_4$	0	$-2kV_3$	0	$-2kV_3$
$V_5$	$2kV_5$	$2kV_3$	0	$4kV_1$	0

Установлено таку залежність між групами та алгебрами Лі [3].

$$Ad(\exp(V))W_0 = W_0 - \varepsilon[V, W_0] + \frac{\varepsilon^2}{2}[V, [V, W_0]] - \dots$$

Таблиця  $Ad(V)$ :

	$V_1$	$V_2$	$V_3$	$V_4$	$V_5$
$V_1$	$V_1$	$e^{-k\varepsilon}V_2$	$e^{k\varepsilon}V_3$	$e^{-2\varepsilon}V_4$	$e^{2k\varepsilon}V_5$
$V_2$	$V_1 + \varepsilon kV_2$	$V_2$	$V_3$	$V_4$	$V_5 + 2k\varepsilon V_3$
$V_3$	$V_1 - k\varepsilon V_3$	$V_2$	$V_3$	$V_4 - 2k\varepsilon V_3$	$V_3$
$V_4$	$V_1 + k\varepsilon V_4$	$V_2$	$V_3$	$V_4$	$V_5 + 4k\varepsilon V_1$
$V_5$	$V_1 - 2k\varepsilon V_5$	$V_2 - 2k\varepsilon V_3$	$V_3$	$V_4 - 4k\varepsilon V_1$	$V_5$

### Висновки:

1. Можливим є новий розв'язок, який не передбачено класичною теорією диференціальних рівнянь. Новий розв'язок:

$$\tilde{y} = c_1 e^{k(x-\varepsilon)} + c_2 e^{-k(x-\varepsilon)}.$$

2. Обчислюємо інваріанти [3] вектора  $\bar{V}_4$ .

$$\frac{dx}{e^{2kx}} = \frac{dy}{e^{2kx}ky} \Rightarrow y_1 = ce^{kx}.$$

Розв'язок рівняння  $-y_x = ky$ .

3. Обчислюємо інваріанти  $\bar{V}_5$ .

$$-\frac{dx}{e^{-2kx}} = \frac{dy}{e^{-2kx}ky} \Rightarrow y_2 = ce^{-kx}.$$

Висновок:  $y_1$  та  $y_2$  – фундаментальні розв'язки рівняння  $y_{xx} = k^2y$ .

**У перспективі** – розгляд проблеми обчислення алгебр та груп Лі для диференціального рівняння  $y_{xx} + k^2y = 0$ .

1. Уорнер Ф. Основы теории гладких многообразий и групп Ли / Ф. Уорнер. – М. : Мир, 1987.
2. Серр Ж. П. Алгебры Ли и группы Ли / Ж. П. Серр. – М. : Мир, 1965.
3. Олвер П. Приложение групп Ли к дифференциальным уравнениям / П. Олвер. – М. : Мир, 1989.
4. Лэм Дж. Л. Введение в теорию солитонов / Дж. Л. Лэм. – М. : Мир, 1983.
5. Білецький А. І. Алгебра Лі рівняння Фоккера-Планка / А. І. Білецький // Сучасні проблеми математичного моделювання та обчислювальних методів : зб. наук. праць. – Рівне, 2013.
6. Фуцич В. И. Симметрия уравнений Максвелла / В. И. Фуцич, А. Г. Никитин. – К. : Наукова думка, 1983.

Прищепя О. В., ст. викл., Бабич Я. А., студент 5 курсу ННІ АКОТ  
(Національний університет водного господарства та природокористування,  
м. Рівне)

## **ГІСТЕРЕЗИСНІ СТРАТЕГІЇ КЕРУВАННЯ ДЛЯ ОДНОКАНАЛЬНИХ СИСТЕМ З ОДНІЄЮ СПРОБОЮ ПОВТОРУ**

**Досліджено стохастичну систему з одним приладом та однією спробою повтору, для якої розглянуто проблему оптимального керування інтенсивністю вхідного потоку та отримано рекурентні формули для стаціонарних імовірностей.**

**Исследовано стохастическую систему с одним прибором и одной попыткой повтора, для которой рассмотрено проблему оптимального управления интенсивностью входного потока и получено рекуррентные формулы для стационарных вероятностей.**

**An one-channel stochastic system with one repeat attempt is examined in the paper. An optimal control problem of input rate for this system is considered. The recurrent formulas for stationary probabilities are given.**

Серед стохастичних систем масового обслуговування особливо вирізняються системи з повторними викликами, які мають широку сферу застосування. Зокрема, це комп'ютерні мережі (локальні та глобальні), системи керування посадкою повітряних суден, системи мобільного зв'язку. Принцип роботи систем масового обслуговування з повторними викликами відображено у такій послідовності: ззовні до системи надходять вимоги для обслуговування; якщо на момент надходження вимог вільним є хоча б один прилад, то вимога відразу ж підлягає обслуговуванню, і після цього залишає систему; якщо всі прилади зайняті, то вимога стає джерелом повторних викликів, що, відповідно, означає повторення через деякий час спроби зайняти вільний прилад та отримання обслуговування. Результати досліджень таких систем подано у роботах [2], [4]. У працях [1], [3] проведено порівняльний аналіз класичних систем масового обслуговування та систем із повторними викликами. Під час дослідження систем із повторними викликами припускають, що може бути повторне звернення вимоги до системи до того часу, поки вона не отримає обслуговування. Останнє є лише приблизним відображенням реальних ситуацій, тому що кількість повторних спроб часто буває обмеженою. Зокрема, у наукових роботах [5], [8], [9] проаналізовано саме системи з обмеженою кількістю

повторних спроб отримати обслуговування, вивчення яких є досить актуальним на сьогодні, особливо з точки зору оптимізації їхньої роботи.

У роботі розглянуто систему типу  $M/M/1$  з однією спробою повтору (рис.1). Формально цю систему можна описати так (див., наприклад, [5], [9]). Ззовні до системи надходять вимоги, які за умови наявного вільного приладу обслуговуються. Якщо останнє не відбувається, проходить формування джерела повторних викликів, звідки повторно може бути опитано систему лише один раз. У разі, якщо під час повторного звернення вимоги, що утворила джерело повторних викликів, прилад є зайнятим – вона залишає систему. Параметри моделі:  $\mu$  – інтенсивність обслуговування,  $\nu$  – інтенсивність повторних опитувань,  $\lambda_j, j = 0,1,\dots$  – інтенсивність вхідного потоку, що залежить від кількості джерел повторних викликів. Ураховуючи змінний характер інтенсивності надходження вимог, у роботі представлено проблему оптимального керування інтенсивністю вхідного потоку в класі гістерезисних стратегій. Оптимізацію роботи такої системи в класі порогових стратегій було висвітлено в роботі [9].

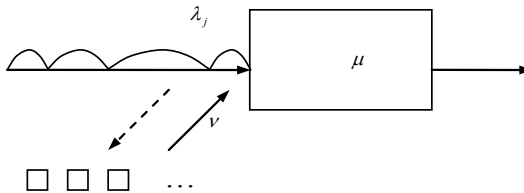


Рис. 1. Структура стохастичної системи з однією спробою повтору

Процес керування інтенсивністю вхідного потоку на основі гістерезисної стратегії можна описати так. Фіксують два невід’ємні цілі числа  $H_1$  та  $H_2$ , які мають назви пороги,  $H_1 \leq H_2$ . Якщо в деякий момент часу кількість повторних викликів у системі не перевищує  $H_1$ , то вона функціонує в першому режимі й інтенсивність вхідного потоку дорівнює  $h_1$ . Якщо кількість повторних викликів більша за  $H_2$ , то система функціонує у другому режимі з інтенсивністю вхідного потоку  $h_2$ . Якщо кількість повторних викликів лежить у проміжку  $(H_1, H_2]$ , то система зберігає той режим, в якому вона функціонувала в попередній момент часу.

Стан системи в будь-який момент часу  $t$  за фіксованої стратегії  $(H_1, H_2]$  може бути описано тривимірним процесом  $Q(t, H_1, H_2) = (Q_1(t, H_1, H_2), Q_2(t, H_1, H_2), R(t, H_1, H_2))$ , де



$Q_1(t, H_1, H_2)$  – кількість зайнятих приладів,  $Q_2(t, H_1, H_2)$  – кількість джерел повторних викликів,  $R(t, H_1, H_2)$  – режим роботи системи. Якщо  $R(t, H_1, H_2) = 1$ , то система працює в першому режимі з інтенсивністю вхідного потоку  $h_1$ . Якщо  $R(t, H_1, H_2) = 2$ , то система працює в другому режимі з інтенсивністю вхідного потоку  $h_2$ .

Процес  $Q(t, H_1, H_2)$  є ланцюгом Маркова з неперервним часом і множиною станів  $S = S^1 \cup S^2$ ,  $S^1 \cap S^2 = \emptyset$ ,

$$\text{де } S^1 = \{i = (i_1, i_2, 1) : i_1 = 0, 1; i_2 = 0, \dots, H_2\},$$

$$S^2 = \{i = (i_1, i_2, 2) : i_1 = 0, 1; i_2 = H_1 + 1, \dots\}.$$

Для заданих параметрів  $\mu, \nu, h_1, h_2 > 0$  інфінітезимальні характеристики  $b_{(i,j,r)(i',j',r')}$ ,  $(i, j, r), (i', j', r') \in S$  визначають так:

$$1) \text{ якщо } \left[ (i = 0) \wedge (j \in \{0, \dots, H_2\}) \wedge (r = 1) \right] \vee \left[ (i = 0) \wedge (j \in \{H_1 + 2, \dots\}) \wedge (r = 2) \right], \text{ то}$$

$$b_{(i,j,r)(i',j',r')} = \begin{cases} h_r, & \text{за } (i', j', r') = (i + 1, j, r); \\ j\nu, & \text{за } (i', j', r') = (i + 1, j - 1, r); \\ -(h_r + j\nu), & \text{за } (i', j', r') = (i, j, r); \\ 0, & \text{в іншому разі;} \end{cases}$$

$$2) \text{ якщо } \left[ (i = 1) \wedge (j \in \{0, \dots, H_2 - 1\}) \wedge (r = 1) \right] \vee \left[ (i = 1) \wedge (j \in \{H_1 + 2, \dots\}) \wedge (r = 2) \right], \text{ то}$$

$$b_{(i,j,r)(i',j',r')} = \begin{cases} h_r, & \text{за } (i', j', r') = (i, j + 1, r); \\ j\nu, & \text{за } (i', j', r') = (i, j - 1, r); \\ \mu, & \text{за } (i', j', r') = (i - 1, j, r); \\ -(h_r + j\nu + \mu), & \text{за } (i', j', r') = (i, j, r); \\ 0, & \text{в іншому разі;} \end{cases}$$

$$3) \text{ якщо } (i, j, r) = (1, H_2, 1), \text{ то}$$

$$b_{(i,j,r)(i',j',r')} = \begin{cases} h_1, & \text{за } (i', j', r') = (1, H_2 + 1, 2); \\ H_2 \nu, & \text{за } (i', j', r') = (1, H_2 - 1, 1); \\ \mu, & \text{за } (i', j', r') = (0, H_2, 1); \\ -(h_1 + H_2 \nu + \mu), & \text{за } (i', j', r') = (i, j, r); \\ 0, & \text{в іншому разі;} \end{cases}$$

4) якщо  $(i, j, r) = (0, H_1 + 1, 2)$ , то

$$b_{(i,j,r)(i',j',r')} = \begin{cases} h_2, & \text{за } (i', j', r') = (1, H_1 + 1, 2); \\ (H_1 + 1) \nu, & \text{за } (i', j', r') = (1, H_1, 1); \\ -(h_2 + (H_1 + 1) \nu), & \text{за } (i', j', r') = (i, j, r); \\ 0, & \text{в іншому разі;} \end{cases}$$

5) якщо  $(i, j, r) = (1, H_1 + 1, 2)$ , то

$$b_{(i,j,r)(i',j',r')} = \begin{cases} h_2, & \text{за } (i', j', r') = (1, H_1 + 2, 2); \\ (H_1 + 1) \nu, & \text{за } (i', j', r') = (1, H_1, 1); \\ \mu, & \text{за } (i', j', r') = (0, H_1 + 1, 2); \\ -(h_2 + (H_1 + 1) \nu + \mu), & \text{за } (i', j', r') = (i, j, r); \\ 0, & \text{в іншому разі.} \end{cases}$$

Позначимо  $\pi_{0j}^{(r)}(H_1, H_2)$ ,  $\pi_{1j}^{(r)}(H_1, H_2)$ ,  $r = 1, 2$ ,  $j = 0, 1, \dots$  стаціонарні ймовірності системи. У подальшому для скорочення викладок будемо використовувати позначення  $\pi_{0j}^{(r)}$ ,  $\pi_{1j}^{(r)}$ , опускаючи індекси  $(H_1, H_2)$ .

За теоремою Твіді [4], для будь-яких  $h_1, h_2, \mu, \nu > 0$  процес обслуговування  $Q(t, H_1, H_2)$  є регулярним, ергодичним і його граничний розподіл співпадає з єдиним стаціонарним розподілом.

Побудуємо розрахункові формули для стаціонарних ймовірностей  $\pi_{ij}^{(r)}$ ,  $(i, j, r) \in S$ .

**Теорема 1.** Якщо параметри моделі типу  $M/M/1$  з однією спробою повтору та керованою інтенсивністю вхідного потоку в класі гістерезисних стратегій є невироджені  $h_1, h_2, \mu, \nu > 0$ , то для процесу обслуговування  $Q(t, H_1, H_2)$ ,  $t \geq 0$  існує стаціонарний режим і стаціонарні ймовірності

можна подати у вигляді:

$$\pi_{00}^{(1)} = \frac{\mu}{h_1} \pi_{10}^{(1)}, \quad \pi_{0j}^{(1)} = \frac{\mu \alpha_j^{(1)} \pi_{10}^{(1)}}{h_1 + j\nu}, \quad \pi_{1j}^{(1)} = \alpha_j^{(1)} \pi_{10}^{(1)}, \quad j = 1, \dots, H_1,$$

$$\pi_{0j}^{(1)} = \frac{\mu}{h_1 + j\nu} \left[ \alpha_j^{(1)} - \frac{h_1 \alpha_{H_2}^{(1)} \beta_{j, H_1+1}^{(1)}}{\nu + h_1 \beta_{H_2, H_1+1}^{(1)}} \right] \pi_{10}^{(1)},$$

$$\pi_{1j}^{(1)} = \left[ \alpha_j^{(1)} - \frac{h_1 \alpha_{H_2}^{(1)} \beta_{j, H_1+1}^{(1)}}{\nu + h_1 \beta_{H_2, H_1+1}^{(1)}} \right] \pi_{10}^{(1)}, \quad j = H_1 + 1, \dots, H_2,$$

$$\pi_{0H_1+1}^{(2)} = \frac{h_1 \mu \alpha_{H_2}^{(1)}}{(H_1 + 1)(\mu + h_2 + (H_1 + 1)\nu)(\nu + h_1 \beta_{H_2, H_1+1}^{(1)})} \pi_{10}^{(1)},$$

$$\pi_{1H_1+1}^{(2)} = \frac{h_1 \alpha_{H_2}^{(1)} (h_2 + (H_1 + 1)\nu)}{(H_1 + 1)(\mu + h_2 + (H_1 + 1)\nu)(\nu + h_1 \beta_{H_2, H_1+1}^{(1)})} \pi_{10}^{(1)},$$

$$\pi_{0j}^{(2)} = \frac{\mu h_1 \alpha_{H_2}^{(1)} \beta_{j, H_1+1}^{(2)}}{(h_2 + j\nu)(\nu + h_1 \beta_{H_2, H_1+1}^{(1)})} \pi_{10}^{(1)},$$

$$\pi_{1j}^{(2)} = \frac{h_1 \alpha_{H_2}^{(1)} \beta_{j, H_1+1}^{(2)}}{\nu + h_1 \beta_{H_2, H_1+1}^{(1)}} \pi_{10}^{(1)}, \quad j = H_1 + 2, \dots, H_2 + 1,$$

$$\pi_{0j}^{(2)} = \frac{\mu \alpha_j^{(2)} h_1 \alpha_{H_2}^{(1)} \beta_{H_2+1, H_1+1}^{(2)}}{(h_2 + j\nu) \alpha_{H_2+1}^{(2)} (\nu + h_1 \beta_{H_2, H_1+1}^{(1)})} \pi_{10}^{(1)},$$

$$\pi_{1j}^{(2)} = \frac{\alpha_j^{(2)} h_1 \alpha_{H_2}^{(1)} \beta_{H_2+1, H_1+1}^{(2)}}{\alpha_{H_2+1}^{(2)} (\nu + h_1 \beta_{H_2, H_1+1}^{(1)})} \pi_{10}^{(1)}, \quad j = H_2 + 2, \dots,$$

де

$$\begin{aligned} \pi_{10}^{(1)} = & \left[ \frac{\mu + h_1}{h_1} + \sum_{j=1}^{H_1} \frac{h_1 + \mu + j\nu}{h_1 + j\nu} \alpha_j^{(1)} + \right. \\ & \left. + \sum_{j=H_1+1}^{H_2} \frac{h_1 + \mu + j\nu}{h_1 + j\nu} \left( \alpha_j^{(1)} - \frac{h_1 \alpha_{H_2}^{(1)} \beta_{j, H_1+1}^{(1)}}{\nu + h_1 \beta_{H_2, H_1+1}^{(1)}} \right) \right] + \end{aligned}$$

$$\begin{aligned}
& + \frac{h_1 \alpha_{H_2}^{(1)}}{\nu + h_1 \beta_{H_2, H_1+1}^{(1)}} \left( \frac{1}{H_1 + 1} + \sum_{j=H_1+2}^{H_2+1} \frac{\beta_{j, H_1+1}^{(2)} (h_2 + \mu + j\nu)}{h_2 + j\nu} + \right. \\
& \left. \sum_{j=H_2+2}^{\infty} \frac{(h_2 + \mu + j\nu) \alpha_j^{(2)} \beta_{H_2+1, H_1+1}^{(2)}}{\alpha_{H_2+1}^{(2)} (h_2 + j\nu)} \right)^{-1}, \\
& \alpha_j^{(r)} = \frac{h_r^j}{j! \nu^j} \prod_{k=0}^{j-1} \frac{h_r + (j-k)\nu}{h_r + \mu + (j-k)\nu}, \quad r = 1, 2, \\
& \beta_{jm}^{(r)} = \sum_{i=0}^{j-m} \frac{(j-i-1)! h_r^i}{j! \nu^i} \prod_{k=0}^i \frac{h_r + (j-k)\nu}{h_r + \mu + (j-k)\nu}, \quad r = 1, 2.
\end{aligned}$$

*Доведення.*

Для пошуку стаціонарних ймовірностей  $\pi_{ij}^{(r)}$ ,  $r = 1, 2$  використаємо теорему про рівність потоку ймовірностей через границю замкненої області в стаціонарному режимі ([10], стор. 49). Для кожного  $j = 0, 1, \dots, H_2$  побудуємо розбиття фазового простору  $S = S_{0j}^{(1)} \cup \bar{S}_{0j}^{(1)}$ , де  $S_{0j}^{(1)} = \{(0, j, 1)\}$ . Прирівнюючи потоки ймовірностей через границю області  $S_{0j}^{(1)}$ , знаходимо:

$$(h_1 + j\nu)\pi_{0j}^{(1)} = \mu\pi_{1j}^{(1)}, \quad j = 0, 1, \dots, H_2. \quad (1)$$

Для кожного  $j = H_1 + 1, \dots$  побудуємо розбиття фазового простору  $S = S_{0j}^{(2)} \cup \bar{S}_{0j}^{(2)}$ , де  $S_{0j}^{(2)} = \{(0, j, 2)\}$ . Прирівнюючи потоки ймовірностей через границю області  $S_{0j}^{(2)}$ , знаходимо:

$$(h_2 + j\nu)\pi_{0j}^{(2)} = \mu\pi_{1j}^{(2)}, \quad j = H_1 + 1, \dots. \quad (2)$$

Тепер для  $j = 0, 1, \dots, H_2$  побудуємо розбиття фазового простору  $S = S_j^{(1)} \cup \bar{S}_j^{(1)}$ ,  $S_j^{(1)} = \{(i, m, 1) : m \leq j\}$ . Прирівнюючи потоки ймовірностей через границю області  $S_j^{(1)}$ , отримаємо таку систему рівнянь:

$$j\nu\pi_{0j}^{(1)} + j\nu\pi_{1j}^{(1)} = h_1\pi_{1j-1}^{(1)}, \quad j = 1, \dots, H_1, \quad (3)$$

$$j\nu\pi_{0j}^{(1)} + j\nu\pi_{1j}^{(1)} + (H_1 + 1)\nu\pi_{0H_1+1}^{(2)} + (H_1 + 1)\nu\pi_{1H_1+1}^{(2)} = h_1\pi_{1j-1}^{(1)},$$

$$j = H_1 + 1, \dots, H_2, \quad (4)$$

$$(H_1 + 1)\nu\pi_{0H_1+1}^{(2)} + (H_1 + 1)\nu\pi_{1H_1+1}^{(2)} = h_1\pi_{1H_2}^{(1)}. \quad (5)$$

Для  $j = H_1 + 1, \dots$  побудуємо розбиття фазового простору  $S = S_j^{(2)} \cup \bar{S}_j^{(2)}$ , де  $S_j^{(2)} = \{(i, m, 2) : H_1 + 1 \leq m \leq j\}$ .

Прирівнюючи потоки ймовірностей через границю області  $S_j^{(2)}$ , отримаємо таку систему рівнянь:

$$j\nu\pi_{0j}^{(2)} + j\nu\pi_{1j}^{(2)} = h_2\pi_{1j-1}^{(2)} + (H_1 + 1)\nu\pi_{0H_1+1}^{(2)} + (H_1 + 1)\nu\pi_{1H_1+1}^{(2)},$$

$$j = H_1 + 2, \dots, H_2 + 1, \quad (6)$$

$$j\nu\pi_{0j}^{(2)} + j\nu\pi_{1j}^{(2)} + h_1\pi_{1H_2}^{(1)} = h_2\pi_{1j-1}^{(2)} + (H_1 + 1)\nu\pi_{0H_1+1}^{(2)} + (H_1 + 1)\nu\pi_{1H_1+1}^{(2)}, \quad j = H_2 + 2, \dots. \quad (7)$$

Використовуючи рівняння (1), подамо  $\pi_{0j}^{(1)}$  через  $\pi_{1j}^{(1)}$ :

$$\pi_{0j}^{(1)} = \frac{\mu}{h_1 + j\nu} \pi_{1j}^{(1)}, \quad j = 1, \dots, H_1.$$

Це подання підставляємо у (3). Отримаємо рекурентне співвідношення для  $\pi_{1j}^{(1)}$ , що визначають через  $\pi_{10}^{(1)}$ :

$$\pi_{1j}^{(1)} = \alpha_j^{(1)} \pi_{10}^{(1)}, \quad j = 1, \dots, H_1, \quad (8)$$

де  $\alpha_j^{(1)} = \frac{h_1^j}{j! \nu^j} \prod_{k=0}^{j-1} \frac{h_1 + (j-k)\nu}{h_1 + \mu + (j-k)\nu}$ . Враховуючи рівняння (1), (2),

підставляємо результат (8) у рівняння (4), отримуємо рекурентне співвідношення для  $\pi_{1j}^{(1)}$ ,  $j = H_1 + 1, \dots, H_2$ , яке виражається через  $\pi_{10}^{(1)}$  та  $\pi_{1H_1+1}^{(2)}$ ,

$$\pi_{1j}^{(1)} = \alpha_j^{(1)} \pi_{10}^{(1)} - \frac{(H_1 + 1)(h_2 + \mu + (H_1 + 1)\nu)}{h_2 + (H_1 + 1)\nu} \beta_{j, H_1+1}^{(1)} \pi_{1H_1+1}^{(2)},$$

$$j = H_1 + 1, \dots, H_2, \quad (9)$$

де  $\beta_{jm}^{(1)} = \sum_{i=0}^{j-m} \frac{(j-i-1)! h_1^i}{j! \nu^i} \prod_{k=0}^i \frac{h_1 + (j-k)\nu}{h_1 + \mu + (j-k)\nu}$ .

Щоб отримати  $\pi_{1H_2}^{(1)}$ , в рекурентному співвідношенні (9) покладемо  $j = H_2$ . Підставляючи в рівняння (5) значення  $\pi_{1H_2}^{(1)}$ , знайдемо представлення для  $\pi_{1H_1+1}^{(2)}$ :

$$\pi_{1H_1+1}^{(2)} = \frac{h_1 \alpha_{H_2}^{(1)} (h_2 + (H_1 + 1)v)}{(H_1 + 1)(\mu + h_2 + (H_1 + 1)v)(v + h_1 \beta_{H_2, H_1+1}^{(1)})} \pi_{10}^{(1)}. \quad (10)$$

Ураховуючи (9), (10), можемо подати в рекурентній формі ймовірності  $\pi_{1j}^{(1)}$ ,  $j = H_1 + 1, \dots, H_2$  через  $\pi_{10}^{(1)}$ :

$$\pi_{1j}^{(1)} = \left[ \alpha_j^{(1)} - \frac{h_1 \alpha_{H_2}^{(1)} \beta_{j, H_1+1}^{(1)}}{v + h_1 \beta_{H_2, H_1+1}^{(1)}} \right] \pi_{10}^{(1)}, \quad j = H_1 + 1, \dots, H_2. \quad (11)$$

Використовуючи рівняння (2), (6), (10), маємо також рекурентні співвідношення для  $\pi_{1j}^{(2)}$ ,  $j = H_1 + 2, \dots, H_2 + 1$ :

$$\pi_{1j}^{(2)} = \frac{h_1 \alpha_{H_2}^{(1)} \beta_{j, H_1+1}^{(2)}}{v + h_1 \beta_{H_2, H_1+1}^{(1)}} \pi_{10}^{(1)}, \quad (12)$$

$$\text{де } \beta_{jm}^{(2)} = \sum_{i=0}^{j-m} \frac{(j-i-1)! h_2^i}{j! v^i} \prod_{k=0}^i \frac{h_2 + (j-k)v}{h_2 + \mu + (j-k)v}.$$

У свою чергу, можна отримати рекурентне співвідношення для  $\pi_{1j}^{(2)}$ ,  $j = H_2 + 2, \dots$ , якщо скористатися рівнянням (7) та врахувати результат (12) за  $j = H_2 + 1$ :

$$\pi_{1j}^{(2)} = \frac{\alpha_j^{(2)} h_1 \alpha_{H_2}^{(1)} \beta_{H_2+1, H_1+1}^{(2)}}{\alpha_{H_2+1}^{(2)} (v + h_1 \beta_{H_2, H_1+1}^{(1)})} \pi_{10}^{(1)},$$

$$\text{де } \alpha_j^{(2)} = \frac{h_2^j}{j! v^j} \prod_{k=0}^{j-1} \frac{h_2 + (j-k)v}{h_2 + \mu + (j-k)v}.$$

Використовуючи рівняння (1), (2), можна отримати стаціонарні ймовірності  $\pi_{0j}^{(1)}$ ,  $j = 0, \dots, H_2$  та  $\pi_{0j}^{(2)}$ ,  $j = H_1 + 1, \dots$ .

Вираз для  $\pi_{10}^{(1)}$  знаходимо з умови нормування:

$$\sum_{j=0}^{H_2} (\pi_{0j}^{(1)} + \pi_{1j}^{(1)}) + \sum_{j=H_1+1}^{\infty} (\pi_{0j}^{(2)} + \pi_{1j}^{(2)}) = 1.$$

Таким чином,

$$\pi_{10}^{(1)} = \left[ \frac{\mu + h_1}{h_1} + \sum_{j=1}^{H_1} \frac{h_1 + \mu + jv}{h_1 + jv} \alpha_j^{(1)} + \right.$$

$$\begin{aligned}
& + \sum_{j=H_1+1}^{H_2} \frac{h_1 + \mu + j\nu}{h_1 + j\nu} \left( \alpha_j^{(1)} - \frac{h_1 \alpha_{H_2}^{(1)} \beta_{j,H_1+1}^{(1)}}{\nu + h_1 \beta_{H_2,H_1+1}^{(1)}} \right) + \\
& + \frac{h_1 \alpha_{H_2}^{(1)}}{\nu + h_1 \beta_{H_2,H_1+1}^{(1)}} \left( \frac{1}{H_1 + 1} + \sum_{j=H_1+2}^{H_2+1} \frac{\beta_{j,H_1+1}^{(2)} (h_2 + \mu + j\nu)}{h_2 + j\nu} + \right. \\
& \left. \sum_{j=H_2+2}^{\infty} \frac{(h_2 + \mu + j\nu) \alpha_j^{(2)} \beta_{H_2+1,H_1+1}^{(2)}}{\alpha_{H_2+1}^{(2)} (h_2 + j\nu)} \right)^{-1}.
\end{aligned}$$

*Теорему доведено.*

Змінний характер інтенсивності вхідного потоку в такій моделі дає можливість ставити і розв'язувати для неї оптимізаційні задачі.

У роботі розглядаємо оптимізаційну задачу

$$C_1 S_1(H_1, H_2) - C_2 S_2(H_1, H_2) - C_3 S_3(H_1, H_2) \rightarrow \max \quad (13)$$

$$H_1, H_2 \in \{0, 1, \dots\}, \quad H_1 \leq H_2,$$

де  $\lim_{t \rightarrow \infty} t^{-1} S_i(t, H_1, H_2) = S_i(H_1, H_2)$  ( $i = \overline{1, 3}$ );  $S_1(t, H_1, H_2)$  – кількість викликів, обслуговування яких завершено в системі за час  $t$  в першому та другому режимах;  $S_2(t, H_1, H_2)$  – кількість викликів, які отримали відмову в обслуговуванні і стали повторними викликами;  $S_3(t, H_1, H_2)$  – кількість перемикань інтенсивності вхідного потоку;  $C_1$  – прибуток, пов'язаний з обслуговуванням одного виклику в першому та другому режимах;  $C_2$  – штраф за відмову в обслуговуванні;  $C_3$  – штраф за перемикання інтенсивності вхідного потоку.

В умовах існування стаціонарного режиму функціонали  $S_i(H_1, H_2) = \lim_{t \rightarrow \infty} t^{-1} S_i(t, H_1, H_2)$ ,  $i = \overline{1, 3}$  існують і можуть бути виписані через стаціонарні ймовірності:

$$S_1(H_1, H_2) = \sum_{j=0}^{H_2} \mu \pi_{1j}^{(1)} + \sum_{j=H_1+1}^{\infty} \mu \pi_{1j}^{(2)},$$

$$S_2(H_1, H_2) = \sum_{j=0}^{H_2} h_1 \pi_{1j}^{(1)} + \sum_{j=H_1+1}^{\infty} h_2 \pi_{1j}^{(2)},$$

$$S_3(H_1, H_2) = h_1 \pi_{1H_2}^{(1)} + (H_1 + 1) \nu \pi_{0H_1+1}^{(2)} + (H_1 + 1) \nu \pi_{1H_1+1}^{(2)}.$$

Розв'язком задачі (13) є такі пороги  $H_1, H_2$ , які максимізують середній прибуток від роботи системи. Подібні оптимізаційні задачі для систем з повторними викликами було розглянуто в роботах [6], [7], [9].

В якості прикладу проаналізуємо задачу оптимізації системи масового обслуговування з такими характеристиками:  $h_1 = 8$ ,  $h_2 = 1$ ,  $\mu = 1$ ,  $\nu = 5$ . Вартісні коефіцієнти функціоналу якості:  $C_1 = 1000$ ,  $C_2 = 23$ ,  $C_3 = 3$ . Максимального прибутку 907,96 досягають за  $H_1 = 1$ ,  $H_2 = 7$ .

**Висновок.** З точки зору практичної реалізації особливе значення мають системи масового обслуговування з однією спробою повтору. Вивчення таких систем є досить актуальним на сьогодні, особливо в розрізі оптимізації їхньої роботи.

У запропонованій роботі досліджено систему масового обслуговування типу  $M/M/1$ , для якої розглянуто гістерезисну стратегію керування. Знайдено рекурентні формули для стаціонарних імовірностей.

**1.** Anisimov V. V. Approximation of multiserver retrial queues by means of generalized truncated models / Anisimov V. V., Artalejo J. R. // Top (2002). – vol. 10. – Num.1 (2002). – pp. 51–66. **2.** Artalejo J. R. Retrial Queueing Systems. A Computational Approach / Artalejo J. R., Gómez-Corral A. – Springer, 2008. – 318 p. **3.** Artalejo J. R. Standard and retrial queueing systems : a comparative analysis / Artalejo J. R., Falin G. I. // Revista matemática Complutense (2002). – vol. XV. – num. 1, pp. 101–129. **4.** Falin G. I. Retrial queues / Falin G. I., Templeton J. G. C. – London : Chapman & Hall, 1997. – 328 p. **5.** Shin Y.W. Retrial queues with limited number of retrials: numerical investigations / Shin Y. W., Moon D. H. // In: The seventh international symposium on operations research and its applications (ISORA'08), Lijiang, China; October 31–November 3, 2008. – pp. 237–247. **6.** Дудин А. Н. Оптимизация динамического управления входной нагрузкой в узле информационно-вычислительной сети / А. Н. Дудин, В. И. Клименок // Автоматика и вычислительная техника. – 1991. – № 2. – С. 25–31. **7.** Клименок В. И. Оптимизация динамического управления режимом работы информационно-вычислительных систем с повторными вызовами / В. И. Клименок // Автоматика и вычислительная техника. – 1990. – № 1. – С. 25–30. **8.** Лебедев Є. О. Стохастичні системи із повторними викликами та нетерплячими вимогами / Є. О. Лебедев, О. В. Прищепя // Вісник Київського університету. Серія : фіз.-мат. науки. – Вип. 2, 2007. – С. 169–173. **9.** Лебедев Є. О. Системи з повторними викликами, нетерплячими вимогами та керованим входним потоком / Є. О. Лебедев, О. В. Прищепя // Журнал Обчислюваної та прикладної математики. – Вип. 2(95). – 2007. – С. 59–64. **10.** Уолрэнд Дж. Введение в теорию сетей массового обслуживания / Дж. Уолрэнд. – М. : Мир, 1993. – 336 с.



Прищеп О. В., ст. викл., Бурмус М. В., студент 5 курсу ННІ АКОТ  
(Національний університет водного господарства та природокористування, м.  
Рівне)

## ОПТИМІЗАЦІЯ РОБОТИ БАГАТОКАНАЛЬНИХ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ З ОДНІЄЮ СПРОБОЮ ПОВТОРУ

У роботі розглянуто задачу оптимального керування інтенсивністю вхідного потоку для системи типу  $M/M/c/N$  з однією спробою повтору в класі порогових стратегій. Знайдено ефективний алгоритм підрахунку стаціонарних ймовірностей.

В работе рассматривается задача оптимального управления интенсивностью входного потока для системы типа  $M/M/c/N$  с одной попыткой повтора в классе пороговых стратегий. Найден эффективный алгоритм подсчета стационарных вероятностей.

In this paper, an optimal control problem of rate of the input flow for  $M/M/c/N$ -type retrial queue with one repeated attempt is considered. The threshold strategy for this system is used. The effective algorithm for stationary probabilities are found

**Основною особливістю стохастичних систем з повторними викликами є** те, що вимоги, які прийшли до системи в разі зайнятості всіх каналів обслуговування, можуть повертатися для обслуговування після певного періоду часу. Такі системи мають широку сферу застосування та розв'язують багато практичних задач у комп'ютерних та телекомунікаційних мережах. Для цих систем припускають, що повторні опитування тривають до того часу, поки вимогу не буде обслуговано (див., наприклад, [1]). Утім, слід звернути увагу на те, що досить часто виникають практичні задачі, в яких використовують системи з обмеженою кількістю повторів, зокрема з однією спробою повтору. Формально ці системи можна описати так. Ззовні до системи надходять вимоги, які за умови наявності хоча б одного вільного приладу обслуговуються, в іншому разі відбувається формування джерела повторних викликів, звідки повторно можуть опитати систему лише один раз. Якщо вимога, яка утворила джерело повторних викликів, в разі повторного звернення знайшла всі прилади зайнятими, вона залишає систему без обслуговування.

У пропонуваній роботі розглянуто систему типу  $M/M/c$  з однією спробою повтору. Загалом за довільної кількості джерел повторних викликів

побудувати явні формули для стаціонарних ймовірностей не можна, тому представимо урізану модель цієї системи  $M/M/c/N$ , в якій наявне скінченне число  $N$  місць для повторного виклику. Процес обслуговування в такому разі змодельовано двовимірним ланцюгом Маркова  $Q^N(t) = (Q_1^N(t), Q_2^N(t))$  з неперервним часом у фазовому просторі  $S^N = I \times J^N$ , де  $I = \{0, 1, \dots, c\}$ ,  $J^N = \{0, 1, \dots, N\}$ . Компонента  $Q_1^N(t)$  вказує на кількість зайнятих приладів на момент часу  $t$  в системі,  $Q_2^N(t)$  дорівнює кількості джерел повторних викликів. Інфінітезимальні характеристики  $a_{(i,j)(i',j')}(t)$  ланцюга  $Q^N(t)$  визначають так:

1) якщо  $i = 0, 1, \dots, c-1$ ,  $j = 0, 1, \dots, N$ , то

$$a_{(i,j)(i',j')}^N = \begin{cases} \lambda_j, & \text{за } (i', j') = (i+1, j), \\ i\mu, & \text{за } (i', j') = (i-1, j), \\ j\nu, & \text{за } (i', j') = (i+1, j-1), \\ -(\lambda_j + i\mu + j\nu), & \text{за } (i', j') = (i, j), \\ 0, & \text{в іншому разі;} \end{cases}$$

2) якщо  $i = c$ ,  $j = 0, 1, \dots, N-1$ , то

$$a_{(c,j)(i',j')}^N = \begin{cases} \lambda_j, & \text{за } (i', j') = (c, j+1), \\ c\mu, & \text{за } (i', j') = (c-1, j), \\ j\nu, & \text{за } (i', j') = (c, j-1), \\ -(\lambda_j + j\nu + c\mu), & \text{за } (i', j') = (c, j), \\ 0, & \text{в іншому разі;} \end{cases}$$

3) якщо  $i = c$ ,  $j = N$ , то

$$a_{(c,N)(i',j')}^N = \begin{cases} c\mu, & \text{за } (i', j') = (c-1, N), \\ N\nu, & \text{за } (i', j') = (c, N-1), \\ -(c\mu + N\nu), & \text{за } (i', j') = (c, N), \\ 0, & \text{в іншому разі.} \end{cases}$$

Для описаної системи визначено такі параметри:  $\mu$  – інтенсивність обслуговування,  $\nu$  – інтенсивність повторних опитувань,  $\lambda_j$  – інтенсивність вхідного потоку, що залежить від кількості джерел повторних викликів.

**Змінний характер інтенсивності** вхідного потоку в аналізованій моделі дає можливість ставити і розв'язувати для них оптимізаційні задачі. З огляду на це розглянемо порогову стратегію, яку задано порогом  $H$ . Якщо на момент часу  $t \geq 0$ , кількість джерел повторних викликів  $Q_2(t) \in [0, H]$ , то будемо казати, що система з повторними викликами функціонує в першому режимі й інтенсивність вхідного потоку на вході системи дорівнює  $h_1$ , якщо  $Q_2(t) \in (H, \infty)$ , то система з повторними викликами функціонує в другому режимі й інтенсивність вхідного потоку на вході системи дорівнює  $h_2$ . Інші параметри від режиму не залежать.

Вибір порогової стратегії  $H$  означає фіксацію залежності  $\lambda_j$  від кількості джерел повторних викликів:

$$\lambda_j = \begin{cases} h_1, & 0 \leq j \leq H, \\ h_2, & j > H. \end{cases}$$

Надалі індекс  $H$  буде вказувати на те, що відповідні характеристики системи розглянуто за фіксованої порогової стратегії.

Нехай  $S_1(t, H)$  – кількість викликів, обслуговування яких завершено в системі за час  $t$ ;  $S_2(t, H)$  – кількість викликів, які отримали відмову в обслуговуванні і стали повторними викликами;  $S_3(t, H)$  – кількість перемикачів інтенсивності вхідного потоку. Якщо існують границі  $\lim_{t \rightarrow \infty} t^{-1} S_i(t, H)$ , то будемо позначати їх через  $S_i(H)$ ,  $i = 1, 2, 3$ .

Розглянемо оптимізаційну задачу

$$C_1 S_1(H) - C_2 S_2(H) - C_3 S_3(H) \rightarrow \max, \quad (1)$$

$$H \in \{0, 1, \dots\},$$

де  $C_1$  – прибуток, пов'язаний з обслуговуванням одного виклику;  $C_2$  – штраф за відмову в обслуговуванні;  $C_3$  – штраф за перемикання інтенсивності вхідного потоку.

Розв'язком задачі (1) є такий поріг  $H$ , який максимізує середній прибуток від роботи системи.

В умовах існування стаціонарного режиму функціонали  $S_i(H)$ ,  $i = 1, 2, 3$  теж існують і їх може бути виписано через стаціонарні ймовірності:

$$S_1(H) = \sum_{j=0}^N \sum_{i=1}^c i \mu \pi_{ij}(H),$$

$$S_2(H) = h_1 \sum_{j=0}^H \pi_{cj}(H) + h_2 \sum_{j=H+1}^N \pi_{cj}(H),$$

$$S_3(H) = h_1 \pi_{cH} + (H+1) \nu \sum_{i=0}^c \pi_{iH+1}.$$

Таким чином, для розв'язання задачі (1) необхідні ефективні алгоритми розрахунку стаціонарного розподілу для системи, яку розглядають.

Процес  $Q^{(N)}(t)$  приймає значення в скінченній множині станів, є стаціонарним за будь-яких  $\lambda_j, \nu, \mu$ . Знайдемо стаціонарний розподіл  $\pi_{ij}^{(N)}$ ,  $(i, j) \in S^{(N)}$ .

Введемо такі позначення для матриць та векторів:

- 1)  $F(j)$ ,  $j = 0, 1, \dots, N-1$  – матриця розмірності  $((c+1) \times (c+1))$  вигляду

$$F(j) = \begin{pmatrix} f_0^{(0)} & f_0^{(+)} & 0 & 0 & \dots & 0 & 0 & f_0^{(*)} \\ f_1^{(-)} & f_1^{(0)} & f_1^{(+)} & 0 & \dots & 0 & 0 & f_1^{(*)} \\ 0 & f_2^{(-)} & f_2^{(0)} & f_2^{(+)} & \dots & 0 & 0 & f_2^{(*)} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & f_{c-1}^{(-)} & f_{c-1}^{(0)} & f_{c-1}^{(+)} + f_{c-1}^{(*)} \\ 0 & 0 & 0 & 0 & \dots & 0 & f_c^{(-)} & f_c^{(0)} \end{pmatrix},$$

- де  $f_i^{(0)} = \lambda_j + i\mu + j\nu$ ,  $i = 0, 1, \dots, c-1$ ;  $f_c^{(0)} = \lambda_j + c\mu$ ;  
 $f_i^{(+)} = -\lambda_j$ ,  $i = 0, 1, \dots, c-1$ ;  $f_i^{(-)} = -i\mu$ ,  $i = 1, 2, \dots, c$ ;  
 $f_i^{(*)} = -j\nu$ ,  $i = 0, 1, \dots, c-1$ ;

- 2)  $B$  – матриця розмірності  $((c+1) \times (c+1))$ , яку визначено так:

$$B = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix};$$

3)  $D(N)$  – трикутна матриця розмірності  $(c \times c)$  вигляду:

$$D(N) = \begin{pmatrix} \mu & 0 & \dots & 0 & 0 \\ -(N\nu + \lambda_N) & 2\mu & \dots & 0 & 0 \\ -N\nu & -(N\nu + \lambda_N) & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ -N\nu & -N\nu & \dots & -(N\nu + \lambda_N) & (c-1)\mu \end{pmatrix};$$

4)  $\bar{1}(k)$  –  $(k)$ -вимірний вектор, що складається з одиниць;

5)  $e_i(k)$  –  $(k)$ -вимірний вектор,  $i$ -та компонента якого дорівнює одиниці, а інші дорівнюють нулю.

**Теорема 1.** Якщо матриці  $F(j)$ ,  $j = 0, 1, \dots, N-1$  мають обернені, то стаціонарні ймовірності  $\pi_{ij}^{(N)}$ ,  $(i, j) \in \mathcal{S}^{(N)}$  можна подати у вигляді:

$$\begin{pmatrix} \pi_{1N}^N \\ \pi_{2N}^N \\ \dots \\ \pi_{cN}^N \end{pmatrix} = \pi_{0N}^N D^{-1}(N) (N\nu \bar{1}(c-1) + \lambda_N e_1(c-1)), \quad (2)$$

$$\pi_{cN}^N = \frac{\pi_{0N}^N}{c\mu} (r_{0N}^N, r_{1N}^N, \dots, r_{c-1N}^N) (N\nu \bar{1}(c) + \lambda_N e_c), \quad (3)$$

$$\begin{pmatrix} \pi_{0j}^N \\ \pi_{1j}^N \\ \dots \\ \pi_{cj}^N \end{pmatrix} = \frac{\pi_{0N}^N N! \nu^{N-j}}{j!} (r_{0N}^N, r_{1N}^N, \dots, r_{cN}^N) W(N-1) \times \dots \times W(j), \quad j=0,1,\dots,N-1, \quad (4)$$

$$\pi_{0N}^N = \left\{ N! \sum_{j=0}^{N-1} \frac{\nu^{N-j}}{j!} (r_{0N}^N, r_{1N}^N, \dots, r_{cN}^N) W(N-1) \times \dots \times W(j) \bar{1}(c+1) + \right. \\ \left. + (r_{0N}^N, r_{1N}^N, \dots, r_{c-1N}^N) \bar{1}(c) + \frac{1}{c\mu} (r_{0N}^N, r_{1N}^N, \dots, r_{c-1N}^N) (N\nu \bar{1}(c) + \lambda_N e_c) \right\}^{-1}, \quad (5)$$

$$\text{де } \begin{pmatrix} r_{0N}^N \\ r_{1N}^N \\ \dots \\ r_{c-1N}^N \end{pmatrix} = \begin{pmatrix} 1 \\ D^{-1}(N) (N\nu \bar{1}(c-1) + \lambda_N e_1 (c-1)) \end{pmatrix},$$

$$r_{cN}^N = \frac{1}{c\mu} (r_{0N}^N, r_{1N}^N, \dots, r_{c-1N}^N) (N\nu \bar{1}(c) + \lambda_N e_c(c)),$$

$$W(j) = BF^{-1}(j), \quad j = 0,1,\dots,N-1.$$

### Доведення.

Під час доведення врахуємо той факт, що за умовою Адамара [2] за  $\lambda_j, \mu, \nu > 0$  матриці  $F(j)$ ,  $j = 0,1,\dots,N-1$  мають обернені.

Для кожного  $k = 0,1,\dots,c-1$  розіб'ємо  $S^{(N)}$  на дві підмножини  $E_k = \{(0, N), (1, N), \dots, (k, N)\}$  та  $\bar{E}_k = S^{(N)} \setminus E_k$ . З огляду на рівність потоків ймовірностей через замкнений контур у стаціонарному режимі ([6]) будемо мати

$$N\nu\pi_{0N}^N + N\nu\pi_{1N}^N + \dots + N\nu\pi_{k-1N}^N + (N\nu + \lambda_j)\pi_{kN}^N = (k+1)\mu\pi_{k+1N}^N, \quad (6) \\ k = 0,1,\dots,c-1.$$

Введемо нову змінну  $r_{ij}^N = \frac{\pi_{ij}^N}{\pi_{0N}^N}$ ,  $(i, j) \in S^{(N)}$ . Тоді перші  $(c-1)$

рівняння системи (6) матимуть вигляд:

$$\left\{ \begin{array}{l} \mu r_{1N}^N = N\nu + \lambda_N \\ -(N\nu + \lambda_N)r_{1N}^N + 2\mu r_{2N}^N = N\nu \\ -N\nu r_{1N}^N - (N\nu + \lambda_N)r_{2N}^N + 3\mu r_{3N}^N = N\nu \\ \dots \\ -N\nu r_{1N}^N - \dots - N\nu r_{c-3N}^N - (N\nu + \lambda_N)r_{c-2N}^N + (c-1)\mu r_{c-1N}^N = N\nu \end{array} \right. \quad (7)$$

Стосовно  $r_{1N}, \dots, r_{c-1N}$  розв'язком (7) буде

$$\begin{pmatrix} r_{1N}^N \\ r_{2N}^N \\ \dots \\ r_{c-1N}^N \end{pmatrix} = D^{-1}(N) \left( N\nu \bar{1}(c-1) + \lambda_N e_1(c-1) \right) \quad (8)$$

З (6) за  $k = c-1$  знаходимо

$$r_{cN} = \frac{1}{c\mu} \left( r_{0N}^N, r_{1N}^N, \dots, r_{c-1N}^N \right) \left( N\nu \bar{1}(c) + \lambda_N e_c(c) \right). \quad (9)$$

Для кожного  $j = 0, 1, \dots, N-1$  будемо розбиття фазового простору  $S^N = S_j^N \cup \overline{S_j^N}$ , де  $S_j^{(N)} = \{(k, m) \in S^{(N)} : m \leq j\}$ . Використовуючи тепер теорему про рівність потоку ймовірностей через границю області  $S_j^N$  у стаціонарному режимі, маємо:

$$\lambda_j \pi_{cj}^N = (j+1)\nu \pi_{0j+1}^N + \dots + (j+1)\nu \pi_{cj+1}^N.$$

Для введеної нової змінної попереднє рівняння матиме вигляд:

$$\lambda_j r_{cj}^N = (j+1)\nu r_{0j+1}^N + \dots + (j+1)\nu r_{cj+1}^N. \quad (10)$$

Розглянемо тепер  $(c+1) \times N$  замкнених контурів, які містять одну точку  $(i, j)$  з області  $\tilde{S}^{(N)} = \{0, 1, \dots, c-1\} \times \{0, 1, \dots, N-1\}$ . Відповідні рівняння для  $r_{ij}^N$ ,  $(i, j) \in \tilde{S}^{(N)}$  мають вигляд:

$$(\lambda_j + j\nu)r_{0j}^N = \mu r_{1j}^N, \quad i = 0, \quad (11)$$

$$(\lambda_j + i\mu + j\nu)r_{ij}^N = (j+1)\nu r_{i-1j+1}^N + \lambda_j r_{i-1j}^N + (i+1)\mu r_{i+1j}^N, \quad (12)$$

$$i = 1, 2, \dots, c-2,$$

$$(\lambda_j + (c-1)\mu + j\nu)r_{c-1j}^N = (j+1)\nu r_{c-2j+1}^N + \lambda_j r_{c-2j}^N + c\mu r_{cj}^N, \quad (13)$$

$$i = c-1,$$

$$(\lambda_j + c\mu + j\nu)r_{cj}^N = \lambda_j r_{c-1j}^N + \lambda_{j-1} r_{cj-1}^N + (j+1)\nu r_{c-1j+1}^N + (j+1)\nu r_{cj+1}^N, \quad i = c. \quad (14)$$

Припустимо для рівняння (10)  $j = j-1$ , тоді перепишемо рівняння (14) у такому вигляді:

$$(\lambda + c\mu)r_{cj}^N = (\lambda_j + j\nu)r_{c-1j}^N + \lambda_{j-1} r_{cj-1}^N + (j+1)\nu r_{c-1j+1}^N + (j+1)\nu r_{cj+1}^N + j\nu r_{0j}^N + j\nu r_{1j}^N + \dots + j\nu r_{c-2j}^N, \quad i = c.$$

Тоді досить легко систему (11)–(14) подати у векторно-матричному вигляді:

$$(r_{0j}^N, r_{1j}^N, \dots, r_{cj}^N)F(j) = (j+1)\nu(r_{0j+1}^N, r_{1j+1}^N, \dots, r_{cj+1}^N)B,$$

або

$$(r_{0j}^N, r_{1j}^N, \dots, r_{cj}^N) = (j+1)\nu(r_{0j+1}^N, r_{1j+1}^N, \dots, r_{cj+1}^N)BF^{-1}(j) = (j+1)\nu(r_{0j+1}^N, r_{1j+1}^N, \dots, r_{cj+1}^N)W(j), \quad j = 0, 1, \dots, N-1. \quad (15)$$



При цьому рекурентний розв'язок (15) можна представити як:

$$\left(r_{0j}^N, r_{1j}^N, \dots, r_{cj}^N\right) = \frac{N!v^{N-j}}{j!} \left(r_{0N}^N, r_{1N}^N, \dots, r_{cN}^N\right) W(N-1) \times \dots \times W(j), \quad (16)$$

$$j = 0, 1, \dots, N-1.$$

Умову нормування для стаціонарних імовірностей  $\pi_{ij}^{(N)}, (i, j) \in S^{(N)}$ , яка

виглядає як  $\sum_{i=0}^c \sum_{j=0}^N \pi_{ij}^N = 1$ , може бути переписано так:

$$\sum_{i=0}^{c-1} r_{iN}^N + r_{cN}^N + \sum_{i=0}^c \sum_{j=0}^{N-1} r_{ij}^N = \left(\pi_{0N}^N\right)^{-1}.$$

Підставляючи сюди вирази з формул (11), (18), отримуємо:

$$\pi_{0N}^N = \left\{ N! \sum_{j=0}^{N-1} \frac{v^{N-j}}{j!} \left(r_{0N}^N, r_{1N}^N, \dots, r_{cN}^N\right) W(N-1) \times \dots \times W(j) \bar{1}(c+1) + \right. \\ \left. + \left(r_{0N}^N, r_{1N}^N, \dots, r_{c-1N}^N\right) \bar{1}(c) + \frac{1}{c\mu} \left(r_{0N}^N, r_{1N}^N, \dots, r_{c-1N}^N\right) (Nv\bar{1}(c) + \lambda_N e_c(c)) \right\}^{-1} \quad (17)$$

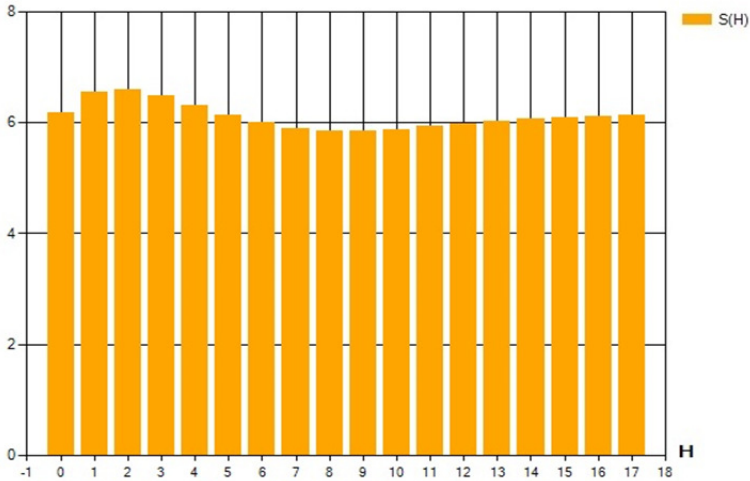
Співвідношення (2)–(5) є безпосереднім наслідком (8), (9), (16), (17). Теорему доведено. Формули (2)–(5) є ефективною рекурентною процедурою для обчислення стаціонарного розподілу для системи  $M/M/c/N$ .

**За допомогою програмної реалізації** було проведено спектр чисельних експериментів для розв'язання оптимізаційної задачі.

Для прикладу, розглянемо випадок керування системою  $M/M/2/18$ . Параметри системи дорівнюють  $h_1 = 2.5$ ,  $h_2 = 0.5$ ,  $\mu = 1$ ,  $v = 0.2$ , а коефіцієнти вартості:  $C_1 = 11$ ;  $C_2 = 7$ ;  $C_3 = 3$ .

Провівши обрахунки, маємо такі результати:

$$\begin{aligned} S(0) &= 6.1682; & S(1) &= 6.5463; & S(2) &= 6.5928; & S(3) &= 6.4800; \\ S(4) &= 6.3071; & S(5) &= 6.1329; & S(6) &= 5.9907; & S(7) &= 5.8958; \\ S(8) &= 5.8507; & S(9) &= 5.8488; & S(10) &= 5.8783; & S(11) &= 5.9256; \\ S(12) &= 5.9782; & S(13) &= 6.0272; & S(14) &= 6.0673; & S(15) &= 6.0971; \\ S(16) &= 6.1174; & S(17) &= 6.1292. \end{aligned}$$



Розв'язок цієї задачі – такий поріг  $H = 2$ , за якого середній прибуток від роботи системи є максимальним.

**Висновки.** В аспекті практичного використання особливо важливими є системи масового обслуговування з обмеженою кількістю повторних спроб. Вивчення таких систем визнано досить актуальним на сьогодні, особливо в площині оптимізації їхньої роботи.

У роботі розглянуто задачу оптимального керування інтенсивністю вхідного потоку для системи типу  $M/M/c/N$  з однією спробою повтору в класі порогових стратегій. Знайдено ефективний алгоритм підрахунку стаціонарних ймовірностей.

Подібні оптимізаційні задачі для систем з повторними викликами опрацьовано в роботах [3], [4], [5].

1. Falin G. I. Retrial queues / Falin G.I., Templeton J.G.C. – London Chapman & Hall, 1997. – 331 p. 2. Гантмахер Ф. Р. Теория матриц / Ф. Р. Гантмахер. – М. : Наука, 1966. – 576 с. 3. Дудин А. Н. Оптимизация динамического управления входной нагрузкой в узле информационно-вычислительной сети / А. Н. Дудин, В. И. Клименок // Автоматика и вычислительная техника. – 1991. – № 2. – С. 25–31. 4. Клименок В. И. Оптимизация динамического управления режимом работы информационно-вычислительных систем с повторными вызовами / В. И. Клименок // Автоматика и вычислительная техника. – 1990. – № 1. – С. 25–30. 5. Лебедев С. О. Системы с повторными вызовами, нетерпеливыми требованиями та керуванням вхідним потоком / С. О. Лебедев, О. В. Прищеп // Журнал Обчислювальної та прикладної математики. – Вип. 2 (95). – 2007. – С. 59–64. 6. Уолрэнд Дж. Введение в теорию сетей массового обслуживания / Дж. Уолрэнд. – М. : Мир, 1993. – 336 с.

**Бачишина Л. Д., ст. викладач, Мукоєд О. В., студент 5 курсу ННІ АКОТ** (Національний університет водного господарства та природокористування, м. Рівне)

## **РОЗРОБКА WEB-РЕСУРСУ „ФОТОСТУДІЯ-ШКОЛА”**

**За допомогою технології ASP .NET MVC3 розроблено web-ресурс „Фотостудія-школа”. Під час розробки було використано такі засоби: Microsoft SQL Server 2008 та Visual Studio 2010.**

**Ключові слова:** ASP (Active Server Pages), MVC (Model – view-controller), SQL (Structured Query Language), БД (база даних), СУБД (система управління базами даних).

**With the technology of ASP. NET MVC 3 developed web-resource «Photo Studio-School». During the development of these tools have been used: Microsoft SQL Server 2008 and Visual Studio 2010.**

**Key words:** ASP (Active Server Pages), MVC (Model – view-controller), SQL (Structured Query Language), DB (database), DBMS (database management system).

Із кожним днем Інтернет усе більше стає складовою щоденного життя кожної людини. З огляду на це зростають вимоги до ресурсів всесвітньої мережі: інтернет-сайти еволюціонують шляхом набуття більшої динамічності, інтерактивності та персоналізованості.

Наприкінці 90-х років Web-сайтом називали логічно зв'язану гіпертекстову інформацію, оформлену у вигляді окремих сторінок і доступну в мережі Інтернет. Для створення веб-сайту було потрібне лише знання мови гіпертекстової розмітки – HTML. Якщо сторінка надавала певні програмні засоби, то це були винятково засоби, що міг надати сервер, на якому розташований веб-сайт. Про зручність і красу тогочасних веб-сайтів взагалі особливо не доводилося говорити. З розвитком мов програмування, розширенням каналів передачі інформації Інтернет набуває статусу самодостатньої галузі економіки, а веб-сайти – статусу повноправних представництв фірм в Інтернеті. Сьогодні мільйони людей вранці встають і йдуть на роботу в Інтернет: їхніми офісами є веб-сайти.

В сучасному баченні веб-сайт – це сукупність програмних, інформаційних, медійних засобів, що логічно пов'язані між собою. Загалом веб-сайт варто визнати відображенням успішності фірми, її „обличчям”.

Основні завдання функціонування веб-сайту:

- реклама продукції, послуг, ідей;
- продаж товарів, послуг, інформації, ідей;

- безкоштовне надання інформації або послуг;
- підтримка клієнтів.

Рекламні веб-сайти може бути створено винятково з рекламно-промоутерською метою. Такі сайти безпосередньо не займаються продажем, їхнє завдання полягає в донесенні до цільової аудиторії рекламної інформації, а відтак їх створюють з орієнтуванням на певне коло товарів або послуг. Зазвичай такі сайти виконують з використанням великої кількості графіки, flash-анімації. Для залучення клієнтів на сайт послугуються ігровими й розважальними методами.

Для веб-сайтів-продавців характерна наявність описового рекламного матеріалу, товарів або послуг, інформація про фірму-продавця та контактна інформація. Додаткові сервіси, такі, як зручність замовлення через сайт у поєднанні із грамотною політикою популяризації, можуть зробити веб-сайт привабливим для сторонніх рекламодавців.

Інформаційні веб-сайти, або сайти, які надають деякі безкоштовні сервіси, теж потрібно обслуговувати, розвивати, а отже, вкладати в них кошти. Але проекти, які не приносять прибутку, довго не „живуть”, тому для цих веб-сайтів притаманне „заробляння грошей” або на рекламі, або на збиранні статистичних даних. На таких сайтах дуже часто пропонують зареєструватися, щоб отримати незначний додатковий сервіс. Як наслідок – подібні веб-сайти дізнаються про користувача практично все: від браузера й роздільної здатності екрана до місця роботи й рахунку в банку. Якщо ж користувач не реєструється на сайті, то стає відомою така інформація, як смак, уподобання, психологічний портрет, приблизний вік. Крім того, користувачу можна показати багато реклами.

Останній тип веб-сайтів – це підтримка клієнтів. Здебільшого на таких сайтах розміщують оновлення для програмних продуктів, новини; якщо йдеться про сайт банку, це може бути система управління засобами клієнта. Ці інтернет-ресурси можна вважати рекламою фірми, товару та ін.

Розроблений нами програмний продукт можна зарахувати до інформаційних веб-ресурсів. Він є сукупністю електронних документів, що знаходяться у веб-просторі мережі, та базою даних фотостудії-школи.

У роботі було використано ASP.NET – технологію створення веб-додатків і веб-сервісів від компанії Microsoft. Вона є складовою платформи Microsoft NET і розвитком технології Microsoft ASP. На сучасному етапі остання версія цієї технології – це ASP.NET 4.5.

ASP.NET зовні багато в чому зберігає схожість із попередньою технологією ASP, що дає змогу розробникам досить легко перейти на ASP.NET. Утім, внутрішня будова ASP.NET істотно відрізняється від ASP, оскільки вона ґрунтується на платформі .NET і, отже, використовує всі нові можливості цієї платформи.

**ASP.NET** входить до .NET Framework, яка містить ретельно відібрану колекцію функціональних, що вражає загальною кількістю складових, – понад 7000 типів. Стандартні мови розробки ASP.NET – це Visual C # і Visual Basic.

Програми ASP.NET завжди компілюються. Насправді щоразу проходять два етапи компіляції. На першому етапі написаний код C# компілюється в код проміжної мови за назвою Microsoft Intermediate Language (MSIL), або просто IL. Цей перший крок є фундаментальною причиною взаємозалежності .NET від мов (ураховуючи C #, Visual Basic і багато інших). Загалом, усі мови .NET (ураховуючи C#, Visual Basic і багато інших) компілюються у, фактично, ідентичний код IL. Цей перший етап компіляції може відбутися автоматично в разі першого запиту сторінки, або ж його можна виконати заздалегідь (цей процес відомий як „попередня компіляція”).

Другий етап компіляції проходить безпосередньо перед фактичним виконанням сторінки. На цьому етапі код IL компілюється в низькорівневий власний машинний код. Цей етап відомий як оперативна компіляція „точно до потрібного моменту” (Just-In-Time – JIT) і проходить однаково для всіх додатків .NET.

Усе середовище .NET Framework, тобто всі простори імен, додатки і класи, називають керованим. Це означає, що відбуваються такі процеси:

- автоматичне управління пам'яттю і збирання сміття;
- безпека типів (під час компіляції програми .NET додає інформацію про доступні класи, їхні члени, типи даних і т.д.);
- розширені метадані (інформація про класи й елементи є лише одним із типів метаданих, збережених .NET у компільованій структурі);
- структурована обробка помилок;
- багатопотоковість (.NET Framework надає пул потоків, які можуть бути використані різними класами. Наприклад, можна асинхронно викликати методи, читати файли або взаємодіяти з Web-службами без необхідності явного створення нових потоків).

ASP.NET є об'єктно-орієнтованим середовищем. Код користувача не тільки має повний доступ до всіх об'єктів в .NET Framework, він також може використовувати всі умовні позначення об'єктно-орієнтованого програмування (ООП).

Найбільш вдалий приклад об'єктно-орієнтованого мислення в ASP.NET можна знайти в серверних елементах управління. Серверні елементи управління є, фактично, інкапсуляцією в мініатюрі. Розробники мають можливість програмно маніпулювати об'єктами управління, використовуючи код для налаштування їхнього зовнішнього вигляду, надання даних для відображення і навіть реакції на події.

Одним із найскладніших випробувань для майбутньої Web-розробки є необхідність підтримання великої кількості браузерів. Різні браузери, версії і конфігурації по-різному підтримують HTML. Web-розробники повинні

вибирати: слід формувати вміст відповідно до стандарту HTML 3.2, HTML 4.0 чи до чого-небудь іншого, як-от XHTML 1.0 або навіть WML для мобільних пристроїв.

ASP.NET вирішив цю проблему у, на диво, розумний спосіб. Незважаючи на можливість витягувати інформацію про браузер клієнта і його властивості всередині сторінки ASP.NET, ASP.NET фактично спонукає розробників ігнорувати ці міркування і використовувати великий набір елементів управління Web-сервера.

Ще одна складна проблема, що постає перед розробниками, – це розгортання готового додатку на виробничому сервері. Кожна інсталяція .NET Framework надає однакові базові класи. Тому розгортання програми ASP.NET визначається як просте: здебільшого потрібно скопіювати всі файли у віртуальний каталог на сервері.

Конфігурування є ще одним завданням, пов'язаним із розгортанням програми, особливо в разі необхідності передачі інформації про безпеку, як-от: обліковий запис і привілеї користувача. ASP.NET спрощує процес розгортання, зводячи до мінімуму залежність від налаштувань IIS (Internet Information Services – інформаційні служби Internet). Замість цього більшість установок ASP.NET зберігається в спеціальному файлі web.config.

Файл web.config вміщено в той же каталог, що і Web-сторінки. У ньому наявні ієрархічно згруповані установки програми, збережені у зрозумілому для людини форматі XML, який можна редагувати з використанням простого текстового редактора, подібного до Notepad. В разі зміни налаштування програми ASP.NET помічає це і перезапускає додаток у новому домені додатків (підтримуючи цей домен програми аж до завершення обробки якихось невиконаних запитів). Файл web.config ніколи не блокується, тому його можна оновлювати в будь-який час.

**У якості середовища** для створення бази даних було обрано SQL Server, а для доступу до даних – Entity Framework (EF) – ORM-платформи .NET. Платформа ORM дає змогу працювати з таблицями, стовпцями і рядками реляційної бази даних, застосовуючи звичайні об'єкти C#. У розділі 4 було згадано, що LINQ може працювати з різними джерелами даних, і один з них – Entity Framework. Надалі у роботі буде пояснено, наскільки це спрощує завдання.

Платформа Entity Framework дозволяє розробникам створювати додатки для доступу до даних, що працюють з концептуальною моделлю програми, а не безпосередньо з реляційною схемою зберігання. Мета полягає в зменшенні обсягу коду і зниженні витрат на супровід додатків, орієнтованих на обробку даних.

Entity Framework є набором технологій ADO.NET, які забезпечують розробку додатків, пов'язаних з обробкою даних. Архітекторам і розробникам додатків, орієнтованих на обробку даних, доводиться враховувати необхідність досягнення двох зовсім різних цілей. Вони повинні

моделювати сутності, зв'язок і логіку розв'язуваних бізнес-завдань, а також працювати з ядрами СУБД, які використовують для збереження і отримання даних. Дані можуть бути розподілені за кількома системами зберігання даних, у кожній з яких застосовують свої протоколи. Втім, навіть у додатках, що працюють з однією системою зберігання даних, необхідно підтримувати баланс між вимогами системи зберігання даних і вимогами написання ефективного і зручного для обслуговування коду програми.

Entity Framework дозволяє працювати з даними у формі специфічних для домену об'єктів і властивостей, як-от: клієнти та їхні адреси, без необхідності звертатися до базових таблиць і стовпчиків бази даних, де збережено ці дані. Entity Framework дає розробникам можливість працювати з даними на більш високому рівні абстракції; створювати і супроводжувати додатки, орієнтовані на дані, використовуючи менше коду, ніж у традиційних додатках. Оскільки Entity Framework є компонентом .NET Framework, додатки Entity Framework можуть працювати на будь-якому комп'ютері, де встановлено платформу .NET Framework, починаючи з версії 3.5 з пакетом оновлень 1 (SP1).

Платформа Entity Framework дозволяє визначити концептуальну модель, виконувати запити, вставляти, оновлювати та видаляти дані, послуговуючись Common Language Runtime (CLR), відображає сутності і відношення, які визначені у моделі бази даних. Entity Framework також надає засоби для виконання таких дій: матеріалізувати дані, відстежувати зміни щодо об'єктів; обробляти паралельні транзакції; виконувати прив'язку об'єктів до елементів управління.

**Першочерговим завданням** під час проектування будь-якої бази даних є розробка моделі даних. *Модель даних* – фіксована система понять і правил для представлення даних структури, стану і динаміки предметної області в базі даних. Нижче наведено структуру таблиць (рис. 1), використаних у базі даних Web-ресурсу „Фотостудії-школи”, та схему зв'язків між ними (рис. 2).

<p><b><u>Клієнти</u></b></p> <p>Код клієнта -pk Код користувача</p>	<p><b><u>Замовлення</u></b></p> <p>Код замовлення - pk Дата Сума Код працівника Код клієнта</p>	<p><b><u>Послуги</u></b></p> <p>Код послуги - pk Назва Ціна Категорія Описання Зображення</p>
<p><b><u>Користувачі</u></b></p> <p>Код користувача - pk Логін Емейл Пароль Коментарі Дата створення Активність Остання дата входу</p>	<p><b><u>Працівники</u></b></p> <p>Код працівника - pk Ім'я Прізвище Телефон Посада</p>	<p><b><u>Заявка</u></b></p> <p>Код замовлення Код послуги</p> <p><b><u>Послуги школи</u></b></p> <p>Код послуги - pk Назва Ціна Термін навчання</p>

Рис. 1. Структура таблиць

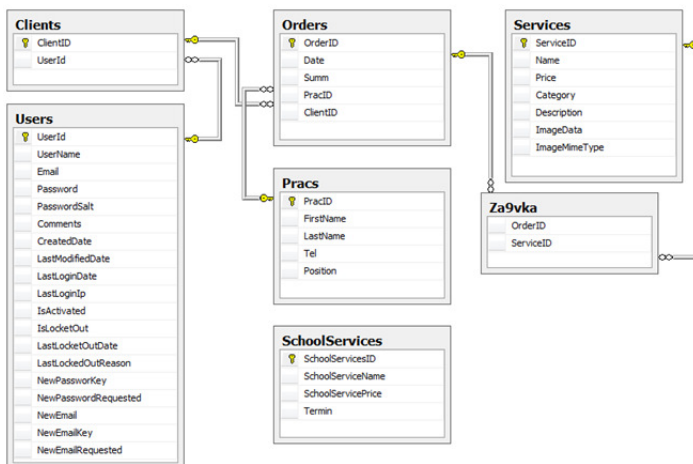


Рис. 2. Зв'язки між таблицями

**Розроблений web-ресурс** дає змогу користувачам переглядати та замовляти послуги, що надає фотостудія, а також записатися на курси фотошколи.

На головній сторінці сайту розміщено загальні відомості про ресурс, категорії послуг, можливість переходу від фотостудії до фотошколи, реалізовано можливість входу (реєстрації) для користувачів (рис. 3).





Рис. 3. Головна сторінка

Після реєстрації користувач отримує повідомлення про активацію на електронну пошту. Для того, щоб активувати обліковий запис, потрібно натиснути на посилання в листі. Якщо користувач зареєструвався й активував свій обліковий запис, він може увійти на сайт. У разі вдалого входу його логін буде відображено зверху на головній сторінці (рис. 4).



Рис. 4. Відображення логіна

Усі послуги фотостудії розділено за категоріями (друк, дизайн, рекламні послуги, фотопослуги). Для прикладу розглянемо категорію „Друк” (рис. 5). Ця категорія охоплює такі послуги: фотографії, фото на кружках, фото на футболках, послугу „весілля” та їхні ціни.

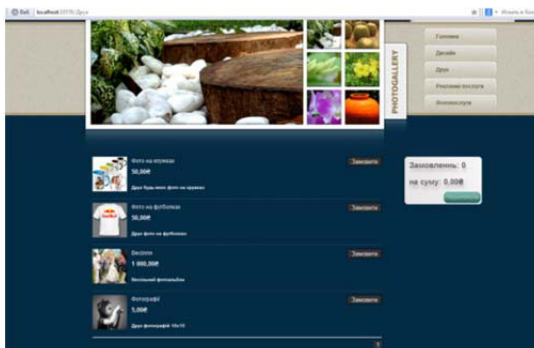


Рис. 5. Послуги друку

На сторінці фотошколи розміщено загальні відомості про неї: галерея „Роботи студентів”, курси (Основи фотографії, Photoshop для фотографа та ін.), форма для подання заявки на навчання.

**Унаслідок аналізу** проведеної роботи можна зробити такий висновок: представлений ресурс надає комплексні послуги фотостудії та фотошколи, що може привернути увагу більшої кількості користувачів; можливість електронної реєстрації забезпечить швидке та якісне обслуговування клієнтів.

У роботі було використано технологію ASP.NET, технологію створення веб-додатків і веб-сервісів від компанії Microsoft, яка є складовою платформи Microsoft .NET і подальшим розвитком попередньої на часі технології Microsoft ASP.

У ході виконання роботи основну увагу зосереджено на необхідності врахування таких критеріїв:

- швидкість роботи (база повинна швидко завантажуватись і так само швидко працювати);
- надійність (вагому роль відіграє надійність бази, тобто стійкість до збоїв, які можуть бути зумовлені невдалими діями користувача або електричним збоєм);
- зручний інтерфейс (зрозумілий, зручний інтерфейс особливо важливий для успішного застосування і швидкого розповсюдження проекту, оскільки користувач передусім звертає увагу на зовнішній вигляд ресурсу.

1. Сандерсон С. ASP.NET MVC3 Framework с примерами на C# для профессионалов / С. Сандерсон. – [3-е изд.]. – М., 2012. – 672 с. 2. Academy Softline – Реализация баз данных Microsoft SQL Server 2008, 2010. 3. Шумаков П. В. Руководство разработчика баз данных / П. В. Шумаков, В. В. Фаронов. – М., 2001. – 640 с. 4. Веб-сайт <http://msdn.microsoft.com/> 5. Райордан Р. М. Основы реляционных баз данных / Р. М. Райордан. – М., 2001. – 384 с. 6. Ржеуцкая С. Ю. Базы данных. Язык SQL / С. Ю. Ржеуцкая. – Вологда, 2010. – 159 с.

Остапчук О. П., к.т.н., ст. викл., Граб Р. І., студент 5 курсу ННІ АКOT (Національний університет водного господарства та природокористування, м. Рівне)

### МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ МІГРАЦІЇ РАДІОНУКЛІДІВ ДО ДРЕН-ВЛОВЛЮВАЧІВ У РАЗІ ПЛОСКО-ВЕРТИКАЛЬНОЇ НАПІРНОЇ ФІЛЬТРАЦІЇ

Сформульовано задачу міграції радіонуклідів до дрен-вловлювачів у разі плоско-вертикальної напірної фільтрації, розроблено відповідну математичну модель та обчислювальний алгоритм. Проведено дослідження концентрації радіонуклідів по області фільтрації з часом.

Сформулирована задача миграции радионуклидов в дрены-уловители при плоско-вертикальной напорной фильтрации, разработана соответствующая математическая модель и вычислительный алгоритм. Проведено исследование концентрации радионуклидов по области фильтрации со временем.

The migration problem of radionuclides to drain-traps was formulated at the float-vertical pressure filtration. The concentration of radionuclides changing in the filtration area during the time are performed. The exploration of changing the radionuclide concentration in the field of filtration under the influence of time was performed.

В сучасних умовах на особливу увагу заслуговують проблеми масоперенесення забруднювальних речовин фільтраційними потоками підземних вод. Забруднювальними речовинами вважають розчини солей, зокрема радіонукліди, які є наслідком радіоактивного забруднення. З огляду на це набувають актуальності питання розроблення, дослідження та впровадження методів математичного моделювання для розв'язання задач, що моделюють процеси адсорбції радіонуклідів та інших хімічних забрудників у ґрунтових середовищах для прийняття оптимальних технічних рішень.

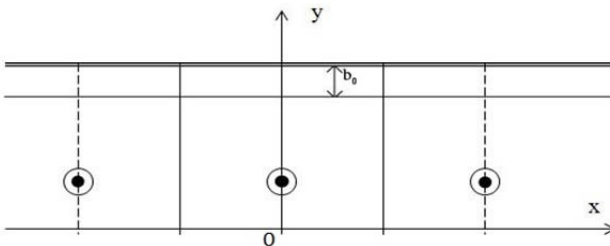


Рис. 1. Систематичний дренаж, наповнений фільтрами-вловлювачами

Для очищення радіоактивно забруднених родючих ґрунтів досить ефективним є застосування фільтрів та дрен-вловлювачів, наповнених деяким сорбуючим матеріалом, наприклад, вермикулітом [1, 2, 3, 4, 5]. Зважаючи на це, видається актуальним розв'язування задачі очищення шару родючого ґрунту від радіонуклідів шляхом перенесення їх фільтраційним потоком до системи горизонтальних дрен із подальшим їхнім уловлюванням (див. рис. 1).

У пропонуваній роботі розглянуто задачу очищення шару родючого ґрунту від радіонуклідів шляхом перенесення їх фільтраційним потоком до горизонтальної дрени з подальшим уловлюванням їх дреною. Нехай маємо переріз ґрунту, в якому на глибині  $b_1$  розташовано дрени-вловлювачі радіусом  $r_l$ , відстань між центрами сусідніх дрен дорівнює  $l$ , відстань від центру дрени до водонепроникного шару ґрунту становить  $b_2$ , товщина верхнього родючого шару ґрунту –  $b_0$ .

Потрібно провести числове моделювання означеного процесу, надалі – побудувати гідродинамічну сітку фільтраційного потоку з візуалізацією її на комп'ютері, розрахувати поля швидкостей фільтрації та концентрації радіонуклідів, спрогнозувати придатність родючих земель до їхнього подальшого використання, тобто визначити час, за який концентрація радіонуклідів у межах родючого шару товщиною  $b_0$  стане меншою за гранично допустиму  $C_r^*$ .

Через симетрію картини течії будемо розглядати лише частину рисунка – область  $AB_1B_2BCD$  (рис. 2). Вона є областю з чотирма зазначеними точками  $A, B, C, D$ , які в разі конформного відображення перейдуть у вершини параметричного прямокутника  $G_w$  (рис. 3).

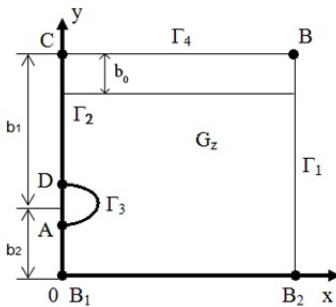


Рис. 2. Фрагмент фізичної області перенесення радіонуклідів

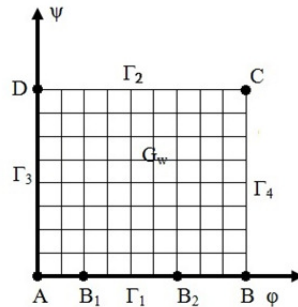


Рис. 3. Параметричний прямокутник в області комплексного потенціалу

Математична модель поставленої задачі у фізичній області матиме вигляд:

$$D \left( \frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} \right) - V_x \frac{\partial c}{\partial x} - V_y \frac{\partial c}{\partial y} - \gamma(c - C_*) = \sigma \frac{\partial c}{\partial t},$$

$$(x, y) \in G_z, t > 0, \quad (1) V_x = \frac{\partial \varphi}{\partial x}, \quad V_y = \frac{\partial \varphi}{\partial y}, \quad \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} = 0, \quad (x, y) \in G_z, \quad (2) \varphi|_{\Gamma_3} = \varphi_1, \quad \varphi|_{\Gamma_4} = \varphi_2, \quad \frac{\partial \varphi}{\partial n} \Big|_{\Gamma_1 \cup \Gamma_2} = 0,$$

$$(3) c(x, y, 0) = \widetilde{C}_0(x, y), \quad (x, y) \in \overline{G_z}, \quad (4) \frac{\partial c}{\partial n} \Big|_{\Gamma_1} = 0, \quad \frac{\partial c}{\partial n} \Big|_{\Gamma_2} = 0, \quad (5) c|_{\Gamma_3} = \widetilde{C}_1(x, y, t), \quad (x, y) \in \Gamma_3, t >$$

бо

$$\frac{\partial c}{\partial n} \Big|_{\Gamma_3} = 0, \quad (6)$$

$$c|_{\Gamma_4} = \widetilde{C}_2(x, y, t), \quad (x, y) \in \Gamma_4, t > 0 \text{ або}$$

$$\frac{\partial c}{\partial n} \Big|_{\Gamma_4} = 0.$$

(7) Перейдемо у математичній

моделі (1)–(7) до нових незалежних змінних  $\varphi, \psi$ , вона матиме вигляд:

$$\frac{\partial x}{\partial \varphi} = \frac{\partial y}{\partial \psi}, \quad \frac{\partial x}{\partial \psi} = -\frac{\partial y}{\partial \varphi}, \quad (\varphi, \psi) \in G_w, \quad (8)$$

$$\Gamma_1: \begin{cases} AB_1: x(\varphi, 0) = 0, \quad \varphi \in (0, \varphi_1), \\ B_1B_2: y(\varphi, 0) = 0, \quad \varphi \in (\varphi_1, \varphi_2), \\ B_2B: x(\varphi, 0) = l/2, \quad \varphi \in (\varphi_2, \varphi_0), \end{cases} \quad (9)$$

$$\Gamma_2: CD: x(\varphi, Q/2) = 0, \quad \varphi \in (0, \varphi_0), \quad (10)$$

$$\Gamma_3: DA: x^2(0, \psi) + (y(0, \psi) - b_1)^2 = r_1^2, \quad (11)$$

$$\Gamma_4: CB: y(\varphi_0, \psi) = b. \quad (12)$$

Рівняння конвективної дифузії матиме такий вигляд:

$$V^2(\varphi, \psi) \left[ \frac{\partial}{\partial \varphi} \left[ D_1 \frac{\partial c}{\partial \varphi} \right] + \frac{\partial}{\partial \psi} \left[ D_2 \frac{\partial c}{\partial \psi} \right] - \frac{\partial c}{\partial \varphi} \right] - \gamma(c - C_m) = \frac{\partial c}{\partial t},$$

$$(\varphi, \psi) \in G_w, t > 0. \quad (13)$$

Крайові умови перетворюються так:

$$\frac{\partial c}{\partial n} \Big|_{\Gamma_1} = \frac{\partial c(\varphi, 0, t)}{\partial \psi} = 0, \quad \frac{\partial c}{\partial n} \Big|_{\Gamma_2} = \frac{\partial c(\varphi, Q/2, t)}{\partial \psi} = 0,$$

$$\varphi \in (0, \varphi_0), t > 0, \quad (14) c|_{\Gamma_3} = \widetilde{C}_1(\psi, t) \text{ або}$$

$$\left. \frac{\partial c}{\partial \varphi} \right|_{\Gamma_3} = \frac{\partial c(0, \psi, t)}{\partial \varphi} = 0, \quad 0 < \psi < \frac{Q}{2}, t > 0, \quad (15)$$

$$c|_{\Gamma_4} = \widetilde{C}_2(\psi, t) \text{ або}$$

$$\left. \frac{\partial c}{\partial \varphi} \right|_{\Gamma_4} = \frac{\partial c(\varphi_0, \psi, t)}{\partial \varphi} = 0, \quad 0 < \psi < \frac{Q}{2}, t > 0, \quad (16)$$

$$c(x(\varphi, \psi), y(\varphi, \psi), 0) = \widetilde{C}_0(\varphi, \psi), (\varphi, \psi) \in \overline{G_w}. \quad (17)$$

Таким чином, (8)–(17) є записом математичної моделі задачі в нових незалежних змінних  $(\varphi, \psi)$ . Задачу потрібно розв'язати в області  $\Omega = G_w \times t$ , де  $G_w$  – параметричний прямокутник,  $t > 0$ .

Спочатку розв'яжемо задачу фільтрації та знайдемо компоненти і квадрат швидкості фільтрації.

Фільтраційна задача має вигляд:

$$V_x \frac{\partial \varphi}{\partial x}, V_y = \frac{\partial \varphi}{\partial y}, V^2(\varphi, \psi) = V_x^2 + V_y^2.$$

Проведемо дискретизацію

$$V_x \approx \frac{\Delta \varphi}{\Delta x} = \frac{h_1}{\frac{1}{2} \cdot (x_{i+1,j} - x_{i-1,j})} = \frac{2h_1}{x_{i+1,j} - x_{i-1,j}},$$

$$V_x \approx \frac{\Delta \varphi}{\Delta y} = \frac{h_2}{\frac{1}{2} \cdot (y_{i,j+1} - y_{i,j-1})} = \frac{2h_2}{y_{i,j+1} - y_{i,j-1}}, V_{ij}^2 = 4h_1^2 \cdot \left( \frac{1}{(x_{i+1,j} - x_{i-1,j})^2} + \frac{1}{(y_{i,j+1} - y_{i,j-1})^2} \right)$$

З точки зору математичної фізики рівняння (13) є двовимірним рівнянням параболічного типу. Тому для розв'язання крайової задачі (13)–(17) використаємо економні методи розв'язання крайових задач математичної фізики.

Запишемо рівняння (13) так:

$$\frac{\partial^2 c}{\partial \varphi^2} + \frac{\partial^2 c}{\partial \psi^2} - \frac{1}{D} \cdot \frac{\partial c}{\partial \varphi} - \frac{\gamma}{D \cdot V^2} \cdot (c - C^*) = \frac{\sigma}{D \cdot V^2} \cdot \frac{\partial c}{\partial t}. \quad (18)$$

Згідно з локальним однорідним методом Самарського [8] крайову задачу (13)–(17) розпишемо на дві підзадачі:

а) для поздовжньої схеми:

$$\begin{cases} \frac{\partial^2 c}{\partial \varphi^2} - \frac{1}{D} \frac{\partial c}{\partial \varphi} - \frac{1}{2} \frac{\gamma}{D \cdot V^2} \cdot (c - C^*) = \frac{1}{2} \frac{\sigma}{D \cdot V^2} \frac{\partial c}{\partial t} \\ c(\varphi, \psi, 0) = C_0(\varphi, \psi), \\ c|_{AD} = C(0, \psi, t) = C_1, \\ c|_{BC} = C(a, \psi, t) = C_2, \end{cases} \quad (19)$$

б) для поперечної схеми:

$$\begin{cases} \frac{\partial^2 c}{\partial \psi^2} - \frac{1}{2} \cdot \frac{\gamma}{D \cdot V^2} \cdot (c - C^*) = \frac{1}{2} \cdot \frac{\sigma}{D \cdot V^2} \cdot \frac{\partial c}{\partial t}, \\ c|_{AB} = C(\varphi, 0, t) = C_3, \\ c|_{DC} = C(\varphi, b, t) = C_4. \end{cases} \quad (20)$$

Оскільки диференціальне рівняння у крайовій задачі (19) містить конвективний член, то для її розв'язання використаємо монотонну різницеву схему, а для крайової задачі (20) – звичайну неявну схему.

Монотонна різницєва схема для крайової задачі (19) має вигляд:

$$\begin{aligned} \mu \cdot \frac{C_{i-1j}^{k+\frac{1}{2}} - 2C_{ij}^{k+\frac{1}{2}} + C_{i+1j}^{k+\frac{1}{2}}}{h_1^2} + r_+ \cdot \frac{C_{i+1j}^{k+\frac{1}{2}} - C_{ij}^{k+\frac{1}{2}}}{h_1} + \\ + r_- \cdot \frac{C_{ij}^{k+\frac{1}{2}} - C_{i-1j}^{k+\frac{1}{2}}}{h_1} - \frac{1}{2} \cdot \frac{\gamma}{D \cdot V^2} \cdot \left( C_{ij}^{k+\frac{1}{2}} - C^* \right) = \\ = \frac{\sigma}{2D \cdot V_{ij}^2} \cdot \frac{C_{ij}^{k+\frac{1}{2}} - C_{ij}^k}{\tau/2}, \end{aligned} \quad (21)$$

де  $C_{0j}^{k+1/2} = C_1$ ,  $C_{ij}^0 = C_{0ij}$ ,  $C_{nj}^{k+1/2} = C_2$ ,

$$\mu = \frac{1}{1+0.5h_1 \cdot |r|} = \frac{1}{1+\frac{h_1}{2D}},$$

де  $r = r_+ + r_-$ ,  $r_+ \equiv 0$ ,  $r_- = r < 0$ ,  $r = -\frac{1}{D}$ . У разі

дискретизації умови  $\frac{\partial c(a, \psi, t)}{\partial \varphi} = 0$ , маємо

$$C_{nj}^{k+1/2} = C_{n-1,j}^{k+1/2}. \text{ Запишемо (21) у прогнотичному вигляді:}$$

$$a_1 C_{i-1,j}^{k+\frac{1}{2}} - c_1 C_{ij}^{k+\frac{1}{2}} + b_1 C_{i+1,j}^{k+\frac{1}{2}} = -f_1, \quad (22)$$

$$C_{0j}^{k+\frac{1}{2}} = C_1, \quad C_{nj}^{k+\frac{1}{2}} = C_2, \quad (23)$$

де

$$a_1 = \left( \frac{\mu}{h_1^2} - \frac{r_-}{h_1} \right) > 0, \quad b_1 = \left( \frac{\mu}{h_1^2} - \frac{r_+}{h_1} \right) > 0,$$

$$c_1 = \frac{2\mu}{h_1^2} + \frac{r_+}{h_1} - \frac{r_-}{h_1} + \frac{1}{2} \cdot \frac{\gamma}{D \cdot V_{ij}^2} + \frac{\sigma}{D \cdot V_{ij}^2 \cdot \tau},$$

$$f_1 = \frac{\sigma}{D \cdot V_{ij}^2 \cdot \tau} \cdot C_{ij}^k + \frac{1}{2} \cdot \frac{\gamma}{D \cdot V_{ij}^2} \cdot C^*.$$

Розв'язок (22)–(23) знаходимо за допомогою методу прогонки, в результаті чого маємо:

$$C_{ij}^{k+1/2} = \alpha_{i+1j}^1 \cdot C_{i+1j}^{k+1/2} + \beta_{i+1j}^1, \quad (24)$$

де

$$\alpha_{i+1j}^1 = \frac{b_1}{c_1 - \alpha_{ij}^1 \cdot a_1}, \beta_{i+1j}^1 = \frac{a_1 \beta_{ij}^1 + f_1}{c_1 - \alpha_{ij}^1 \cdot a_1}, \quad (25)$$

$$\alpha_{1j}^1 = 0, \beta_{1j}^1 = C_1. \quad (26)$$

Розв'яжемо крайову задачу (20), використовуючи неявну різницеву схему:

$$\frac{C_{ij-1}^{k+1} - 2C_{ij}^{k+1} + C_{ij+1}^{k+1}}{h_2^2} - \frac{1}{2} \cdot \frac{\gamma}{D \cdot V_{ij}^2} \cdot (C_{ij}^{k+1} - C^*) = \frac{\sigma}{D \cdot V_{ij}^2} \cdot \frac{C_{ij}^{k+1} - C_{ij}^{k+1/2}}{\tau}, \quad (27)$$

де

$$C_{ij}^0 = C_{0ij}, C_{i0}^{k+1} = C_{i1}^{k+1}, C_{im}^{k+1} = C_{i m-1}^{k+1}. \quad (28)$$

Запишемо (27)–(28) у прогоночному вигляді:

$$a_2 C_{ij-1}^{k+1} - c_2 C_{ij}^{k+1} + b_2 C_{ij+1}^{k+1} = -f_2, \quad (29)$$

де

$$a_2 = \frac{1}{h_2^2}, b_2 = \frac{1}{h_2^2}, c_2 = \frac{2}{h_2^2} + \frac{\gamma}{2D \cdot V_{ij}^2} + \frac{\sigma}{D \cdot V_{ij}^2 \cdot \tau},$$

$$f_2 = \frac{\sigma}{D \cdot V_{ij}^2 \cdot \tau} \cdot C_{ij}^{k+1/2} + \frac{1}{2} \cdot \frac{\gamma}{D \cdot V_{ij}^2} \cdot C^*.$$

Розв'язок (29) знаходимо за допомогою методу прогонки:

$$C_{ij}^{k+1} = \alpha_{i+1j}^2 \cdot C_{i+1j}^{k+1} + \beta_{i+1j}^2,$$

де

$$\alpha_{ij+1}^2 = \frac{b_2}{c_2 - \alpha_{ij}^2 \cdot a_2}, \beta_{ij+1}^2 = \frac{a_2 \beta_{ij}^2 + f_2}{c_2 - \alpha_{ij}^2 \cdot a_2}, \alpha_{i1}^2 = 0, \beta_{i1}^2 = 0. \text{ На основі}$$

вищенаведених алгоритмів було створено програмне забезпечення за допомогою технології .NET Framework та мови програмування С#, що дає змогу чисельно та графічно переглянути отриману гідродинамічну сітку (рис. 4), вектор швидкості фільтрації (рис. 5) та розподіл концентрації радіонуклідів упродовж 30 років (рис. 6–8).

На основі програмної реалізації було проведено низку числових експериментів із такими вхідними даними:  $\tau=30$  діб,  $\lambda=1$ м,  $\sigma=0,2$ ;  $\gamma=0,0065$   $\gamma=0,0065$   $\text{добу}^{-1}$ ;  $D_m=0,000001 \text{ м}^2/\text{добу}$ ;  $C_*=2$   $\text{кг/м}^3$ ;  $\bar{C}_0=5$   $\text{кг/м}^3$ ;  $\bar{C}_1=0$   $\text{кг/м}^3$ ;  $\bar{C}_2=5 \text{ кг/м}^3$ .



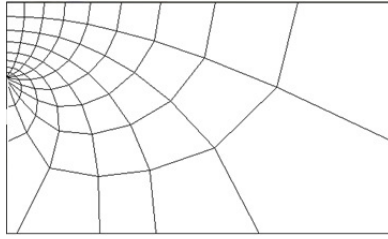


Рис. 4. Вигляд гідродинамічної сітки фільтраційного потоку

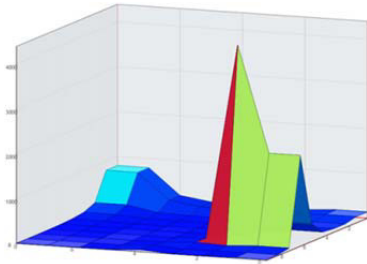


Рис. 5. Просторове зображення вектора швидкості фільтрації

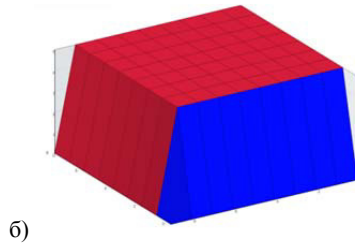
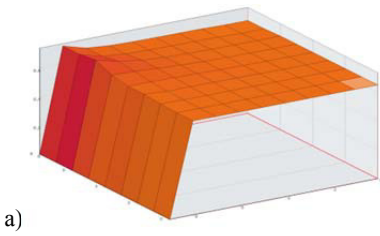


Рис. 6. Розподіл концентрації радіонуклідів за:  
а) t=0 років б) t=30 років

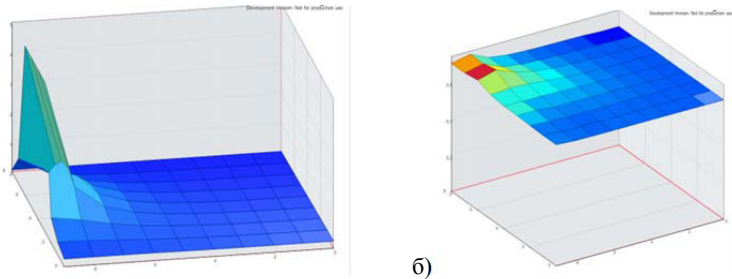


Рис. 7. Розподіл концентрації радіонуклідів у разі задання початкового забруднення за законом  $C_0=C_0 \cdot e^{-1,5y}$ , із використанням граничної умови другого роду на поверхні області ( $\Gamma_4$ ) за:

а) t=0 років б) t=30 років

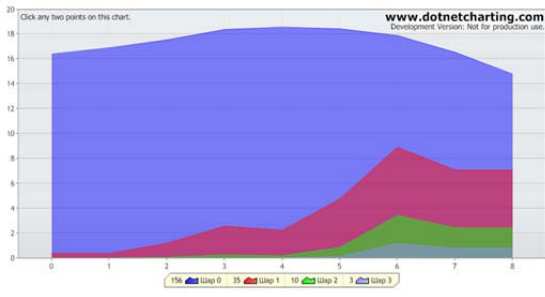


Рис. 8. Динаміка фронту концентрації в разі задання на поверхні області умови другого роду та  $C_0=C_0 \cdot e^{-1,5y}$  із часом

**Унаслідок аналізу результатів** числових експериментів можна стверджувати, що:

- час очищення родючого шар ґрунту, що складає 0,4 м і приблизно відповідає верхній межі фізичної області та першій верхній лінії побудованої різницевої сітки, до гранично допустимих значень, становить близько 8-10 років у разі задання розрахункових даних;
- коефіцієнт масообміну  $\gamma$  має суттєвий вплив на розподіл концентрації забрудника. У разі ускладнення задачі граничною умовою другого роду та задання початкової концентрації за експоненціальним законом вплив коефіцієнта масообміну стає менш помітним;
- під час аналізу отриманих результатів числового розв'язання задачі для різних початкових даних та граничних умов на межі  $\Gamma_4$  виявлено їхню суттєву відмінність.

З огляду на вищесказане можна зробити висновок, що час очищення земель до гранично допустимих значень залежить від задання початкових та граничних умов і фізичних властивостей ґрунту.

**1.** Власюк А. П. Математичне моделювання задачі про утилізацію радіонуклідів в шарі ґрунту / А. П. Власюк, О. П. Остапчук // Тези доп. XI Міжн. конф. ім. академіка М. Кравчука. – К., 2006. – С. 57. **2.** Власюк А. П. Числове моделювання локалізації радіонуклідів за допомогою дренажних вловлювачів при плоско-вертикальній напірній фільтрації / А. П. Власюк, О. П. Остапчук // Сучасні проблеми математичного моделювання та обчислювальних методів : тези доп. Всеукр. наук. конф. – Рівне, 2013. – С. 43. **3.** Власюк А. П. Числове моделювання міграції радіонуклідів при горизонтальній фільтрації підземних вод / А. П. Власюк, О. П. Остапчук // Прогнозування та прийняття рішень в умовах невизначеності : тези доп. Міжн. школи-семінару. – Східниця, 2006. – С. 180–181. **4.** Власюк А. П. Числове моделювання міграції радіонуклідів при плановій фільтрації підземних вод / А. П. Власюк, О. П. Остапчук // Прогнозування та прийняття рішень в умовах невизначеності : тези доп. Міжн. конф. – Чернівці, 2007. – С. 72–74. **5.** Власюк А. П. Числове моделювання однієї задачі про утилізацію радіонуклідів / А. П. Власюк, О. П. Остапчук // Прогнозування та прийняття рішень в умовах невизначеності : тези доп. Міжн. конф. – Бердянськ, 2005. – С. 111–113. **6.** Косолапова Е. В. До питання моніторингу ґрунтів, забруднених радіонуклідами / Е. В. Косолапова, С. І. Веремеєнко // Екологія. Людина. Суспільство : тези доповідей учасників II Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених (13-14 травня) – К. : НТУУ “КПІ”, – 1999. – С. 21–22. **7.** Прокопов Г. П. О расчете разностных сеток, близких к ортогональным, в областях с криволинейными границами / Г. П. Прокопов. – М., 1974. – 36 с. – (Препр. / АН СССР. Ин-т прикл. математики; 74.17). **8.** Самарский А. А. Теория разностных схем / А. А. Самарский. – [3-е изд]. – М. : Наука, 1989. – 616 с.

Громадченко Т. В., асп., Мартинюк П. М., доцент, к.ф.-м.н.,  
Ткачук С. А., студент 5 курсу ННІ АКОТ (Національний університет  
водного господарства та природокористування, м. Рівне)

## **ВИЗНАЧЕННЯ КРИТИЧНОЇ ПОВЕРХНІ КОВЗАННЯ В ГРУНТОВОМУ СХИЛІ ЗА ДОПОМОГОЮ МЕТОДІВ ОПТИМІЗАЦІЇ**

**Розглянуто задачу про визначення критичної поверхні ковзання з найменшим коефіцієнтом стійкості в ґрунтовому схилі. Числові розв'язки відповідної оптимізаційної задачі знайдено за допомогою методу покоординатного спуску. На основі програмної реалізації запропонованих алгоритмів проведено та проаналізовано низку числових експериментів.**

**Ключові слова:** коефіцієнт стійкості, поверхня ковзання, методи спуску.

**Consider the problem of determining the critical slip surface with the lowest factor of stability in the soil slope. Numerical solutions of the corresponding optimization problem found by pokoordynatnoho descent. Based on the software implementation of the proposed algorithms, a number of numerical experiments and made their analysis.**

**Key words:** coefficient of resistance, sliding surface, descent methods.

**Рассмотрено задачу об определении критической поверхности скольжения с наименьшим коэффициентом устойчивости в ґрунтовом склоне. Численные решения соответствующей оптимизационной задачи найдено методом покоординатного спуска. На основе программной реализации предложенных алгоритмов проведен ряд численных экспериментов и сделано их анализ.**

**Ключевые слова:** коэффициент устойчивости, поверхность скольжения, методы спуска.

**Вступ.** Динаміка розвитку сучасного будівництва є не тільки кількісною, але і якісною. На сьогодні значного поширення набуло будівництво виняткових за своїми масштабами споруд. Надійність останніх значною мірою залежить від стійкості ґрунтових основ, на яких їх зводять.

З огляду на вищевикладене проблеми оцінювання стійкості укосів і схилів, прогнозування зсувів і їхнє зміцнення продовжують залишатися актуальними. Увіразнює актуальність названих проблем зростаючий дефіцит вільних земельних площ, освоєння зсувонебезпечних територій під будівництво, а також активізація наявних і поява нових зсувів, зумовлених втручанням людини в геологічне середовище.

У контексті забезпечення адекватного вирішення означених вище проблем потребують подальшого вивчення методи обрахунку стійкості укосів і схилів, зокрема завдання автоматизації таких обрахунків та створення відповідного програмного забезпечення.

Вирішенням цієї проблеми займалось багато вчених [1, 2, 4, 6]. Відомими є роботи О. Л. Можевітінова, Р. Р. Чугайова, С. Г. Золотарьова, Я. Х. Хуан, В. Г. Федоровського. Теоретичні основи для визначення критичної поверхні ковзання з найменшим коефіцієнтом стійкості в ґрунтових схилах закладено в працях П. Л. Іванова, К. Л. Гінзбурга, В. В. Соколовського та ін. Метою пропонованої роботи є алгоритмізація та програмна реалізація відомих інженерних методів для відшукування критичної поверхні ковзання як оптимізаційних задач.

### Постановка задачі

Розглянемо задачу відшукування критичної поверхні ковзання ґрунтового укосу – область АОBC (рис. 1), яка відповідає мінімальному значенню коефіцієнта стійкості. Вважаємо, що фізико-механічні характеристики є заданими, а також відомою вологість ґрунту.

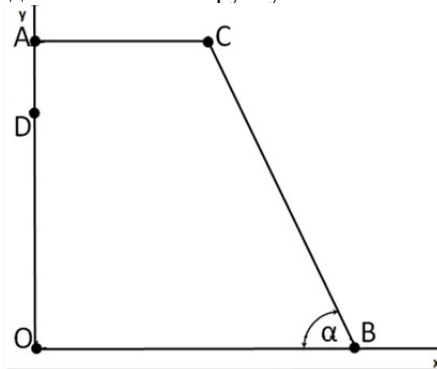


Рис. 1. Ґрунтовий схил

Аналізована область складається з однорідного та ізотропного за своїми характеристиками ґрунту. Ґрунтовий схил є нерівномірно зволуженим. У точці D можливий витік рідини (у найпростішому варіанті – води).

Потрібно побудувати критичну поверхню ковзання, що відповідає мінімальному значенню коефіцієнта стійкості.

### Інженерні методи розрахунку стійкості укосів і схилів

*Метод Федоровського, Курило [2].* Розглянемо в плоскій постановці задачу про стійкість укосу, контур якого в координатах  $\{x, y\}$  ( $x$  – по горизонталі;  $y$  – по вертикалі вгору) запишемо рівнянням  $y = \bar{y}(x)$  (рис. 2). Ґрунт вважають кусково-однорідним;  $i$ -тий шар ґрунту (нумерація шарів  $i=1, \dots, m$  відбувається зверху донизу) визначається нерівностями  $\tilde{y}_i(x) \geq y > \tilde{y}_{i+1}(x)$  (для останнього шару  $y \leq \tilde{y}_m(x)$ ). В будь-якому

вертикальному перерізі розглянутого шару рівень підземних вод може бути нижчим за його межу, вищим (якщо вищерозміщений шар також водоносний) або на рівні межі. У всіх випадках (у припущенні квазігоризонтальності фільтраційного потоку) поровий тиск води у водоносному  $i$ -тому шарі характеризується гідростатичним напором у кожному вертикальному перерізі  $h_i(x)$ , який в разі безнапірного режиму співпадає з рівнем підземних вод. Поровий тиск у точці  $(x, y)$   $i$ -го шару дорівнює:

$$u(x, y) = \begin{cases} \gamma_w(h_i(x) - y), & \text{якщо } y \leq h_i(x) \\ 0, & \text{якщо } y > h_i(x), \end{cases} \quad (1)$$

де  $\gamma_w$  – питома вага води.

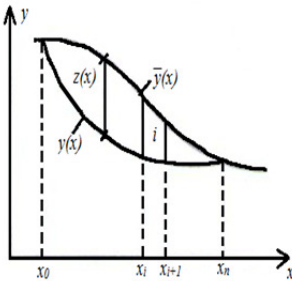


Рис. 2. Схема розрахунку коефіцієнта стійкості укосу

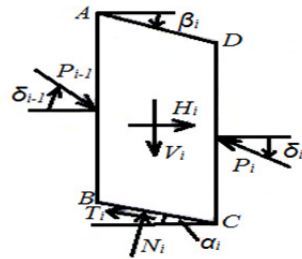


Рис. 3. Сили, що діють на відсік

Враховуємо очевидне з фізичних міркувань обмеження (рис. 2)

$$y(x) < \bar{y}(x) \text{ при } x_0 < x < x_n. \quad (2)$$

У переважній більшості методів розрахунку стійкості укосів рухомий масив розбивають на відсіки вертикальними лініями. Цей же прийом використаємо і тут.

Із припущення, що рухомий масив знаходиться в стані граничної рівноваги (основного припущення для пошуку коефіцієнта запасу стійкості), можна зробити висновок, що

$$T_i = \frac{(N_i \operatorname{tg} \varphi_i + c_i l_i)}{f}, \quad (3)$$

де  $\operatorname{tg} \varphi_i$ ,  $c_i$  – міцнісні характеристики ґрунтового відсіку  $BC$ ,  $l_i$  – довжина цього відсіку.

Згідно з цим методом отримано такі вирази для  $N_i$  і  $P_i$  із урахуванням (3):

$$N_i = \left[ -\overline{H}_i \sin \delta_i + \overline{V}_i \cos \delta_i + P_{i-1} \sin(\delta_{i-1} - \delta_i) - \frac{c_i l_i \sin(\alpha_i - \delta_i)}{f} \right] / D_i,$$

$$P_i = \left\{ \overline{H}_i \left( \cos \alpha_i + \frac{tg \varphi_i \sin \alpha_i}{f} \right) + \overline{V}_i \left( \frac{\sin \alpha_i - tg \varphi_i \cos \alpha_i}{f} \right) - \right. \\ \left. - \frac{c_i l_i}{f} + P_{i-1} \left[ \cos(\alpha_i - \delta_{i-1}) + \frac{tg \varphi_i \sin(\alpha_i - \delta_{i-1})}{f} \right] \right\} / D_i, \quad (6)$$

де  $D_i = \cos(\alpha_i - \delta_i) + \frac{tg \varphi_i \sin(\alpha_i - \delta_i)}{f}$ ;  $\overline{H}_i = H_i + \Delta H_i$ ;  $\overline{V}_i = V_i - \Delta V_i$ .

Очевидно, що  $P_0 = 0$  (оскільки довжина відповідного відрізка дорівнює 0). Оскільки довжина останнього вертикального відрізка дорівнює нулю, повинно бути

$$P_n(f) = 0. \quad (7)$$

Вищенаведене рівняння є нелінійним відносно  $f$ , розв'язавши яке можна знайти коефіцієнт запасу для цієї лінії ковзання, точніше для такої її апроксимації, яка визначається  $(n+2)$  величинами  $x_0, x_n$  і  $z_i = \bar{y}(x_i) - y_i (i = 0, 1, 2, \dots, n - 1)$ .

*Метод Можевітінова [4].* Розглянемо розбитий відрізок обрушення з недеформівними елементами – стовпчиками (рис. 5) – за наявності гранично напруженого стану на поверхні ковзання. Систему сил, які діють на один елемент, прийнято таку, як на рис. 4, але для зручності подальшого викладу частину сил зведено до рівнодіючих.

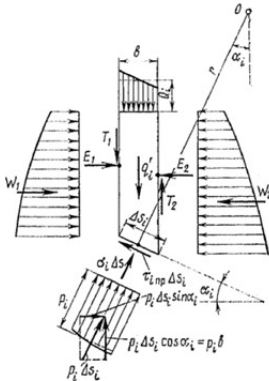


Рис. 4. Сили, що діють на відріск

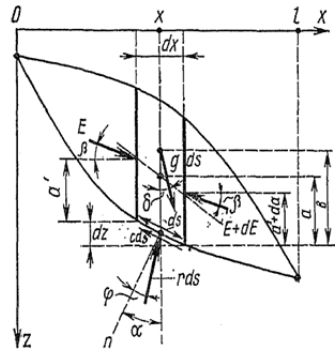


Рис. 5. Призма обрушення й елементарний затвердлий вертикальний відріск

Рівнянь рівноваги елемента призми обрушення (рис. 4) в цьому методі недостатньо, щоб оцінити  $k_3$ . Тому необхідним є введення додаткових

фізичних уявлень, або гіпотез. Як і в інших способах, запропоновано прийняти кут  $\beta$  нахилу сил взаємодії  $E$  постійним для всіх відсіків

$$E_z = E_x \operatorname{tg} \beta, \quad (8)$$

де  $\beta = \text{const}$ .

О. Л. Можевітиновим було запропоновано приблизно прийняти

$$\beta = [\sum(a_i + \delta_i)\Delta x] / l. \quad (9)$$

Згідно з таким методом отримано:

$$E(x) = \int_0^x \frac{g \sin(a+\delta-\varphi) dx - c \cos \varphi ds}{\cos(\alpha-\beta-\varphi)}. \quad (10)$$

Так, переходячи у рівнянні (10) від інтегрування до елементарного сумування по всій довжині призми  $l$  і відокремивши члени, які містять  $\operatorname{tg} \varphi$  і  $c$ , О. Л. Можевітинов запропонував вираз для коефіцієнта запасу стійкості у вигляді:

$$k_3 = \frac{R}{A} = \left[ \sum G_i \cos(\beta + \delta_i) \operatorname{tg}(\varphi_i + \beta - a_i) + \sum \frac{c_i \cos \varphi_i \Delta s_i}{\cos(\varphi_i + \beta - \alpha_i)} \right] / \sum G_i \sin(\beta + \delta_i), \quad (11)$$

де  $G$  – рівнодіюча сил, які діють на відсік,  $\delta$  – кут нахилу по вертикалі рівнодіючих сил,  $\delta = 0$ ,  $\alpha$  – кут нахилу по горизонталі поверхні ковзання,  $\Delta S$  – довжина поверхні ковзання,  $G_i$  – розкладена на нормальну  $P_{Ni}$  і дотичну  $P_{Qi}$ ,  $P_{Qi} = P_i \cos \alpha_i$ ,  $P_{Ni} = P_i \sin \alpha_i$ .

#### **Чисельний метод розв'язування задачі**

Для знаходження мінімального значення коефіцієнта стійкості використовують методи оптимізації функції багатьох змінних. Розглянемо метод покоординатного спуску.

*Метод покоординатного спуску [7].* Цей метод використовують для пошуку мінімуму функції  $n$  змінних. Розглянемо функцію  $f(x): R^n \rightarrow R$ .

#### Алгоритм 1.

1. Ініціалізація деяких початкових значень  $x_i^0 \in R^n, i = \overline{1, n}$ .
2. Виконуємо у циклі для  $i = \overline{1, n}$ :
  - фіксуємо значення всіх змінних функції, окрім  $x_i$ , і тим самим отримуємо функцію однієї змінної  $f(x_i)$ ;
  - проводимо одновимірну оптимізацію по змінній  $x_i$ , вибравши будь-який метод одновимірної оптимізації;
    - якщо виконується критерій зупинки, то фіксуємо значення  $x = (x_1, x_2, \dots, x_n)$ .



## Алгоритм 2.

1. Ініціалізація деяких початкових значень  $x_i^0 \in R^n, i = \overline{1, n}$  і кроку зміни  $h$ .
2. Фіксуємо значення всіх змінних, окрім  $x_i$ , і знаходимо значення функції  $f(x_i)$ .
3. Якщо  $f(x_i) < f(x_i + h)$ , то  $x_i := x_i - h$ , тут “:=” – символ присвоєння, то зменшуємо значення  $x_i$  на крок  $h$  доти, поки  $f(x_i - h) < f(x_i)$ .
4. Якщо  $f(x_i) > f(x_i + h)$ , то збільшуємо значення  $x_i$  на крок  $h$  доти, поки  $f(x_i - h) > f(x_i)$ .
5. Повторюємо пункти 2, 3, 4 для всіх  $i = 1..n$ .

Схема розрахунку коефіцієнта стійкості і побудови критичної поверхні ковзання за методом Федоровського, Курило [2].

1. Задаємо фізико-механічні характеристики ґрунту, а саме розміри ( $H, L, \alpha$ ), густину  $\rho$ , вологість  $W$ , і обчислюємо міцнісні характеристики  $c$  і  $\varphi$ .
2. Ґрунтовий схил розбиваємо опорними вертикалями  $x_k, k = 1..n$  і для кожної з цих вертикалей задаємо значення  $\overline{y}_k(x_k), y_k(x_k)$ , де  $\overline{y}_k(x_k)$  – апроксимація контурів схилу, а  $y_k(x_k)$  – апроксимація початкової поверхні ковзання. Задаємо кути  $\delta_i$ , але, як дають підстави стверджувати пробні розрахунки, мінімальне значення коефіцієнта стійкості отримуємо за  $\delta_i = 0$ .
3. Підраховуємо величини  $H_i, V_i, \tan \varphi_i, c_i$  для кожного відсіка. Якщо сейсмічних сил немає, то  $H_i = 0$ .  $V_i$  визначають як інтеграл сил ваги:

$$V_i = \int_{x_{i-1}}^{x_i} dx \cdot \int_{y_i}^{\overline{y}_i} \gamma dy = \int_{x_{i-1}}^{x_i} w(x) dx. \quad (12)$$

Функцію  $w(x)$  апроксимують ламаною й інтеграл (12) підраховуємо за методом трапецій.

4. Обчислюємо

$$\cos \alpha_i = \frac{x_{i+1} - x_i}{\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}}, \quad \sin \alpha_i = \sqrt{1 - \cos^2 \alpha_i} \text{ і довжину}$$

початкової кривої ковзання в кожному відсіку

$$l_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}.$$

5. Знаходимо

$$T_i = \frac{(N_i \operatorname{tg} \varphi_i + c_i l_i)}{f},$$
$$N_i = \left[ \frac{V_i - c_i l_i \sin(\alpha_i - \delta_i)}{f} \right] / D_i,$$
$$P_i = \left\{ V_i \left( \frac{\sin \alpha_i - \operatorname{tg} \varphi_i \cos \alpha_i}{f} \right) - \frac{c_i l_i}{f} + \left[ \frac{\cos(\alpha_i) + \operatorname{tg} \varphi_i \sin(\alpha_i)}{f} \right] \right\} / D_i,$$

де  $D_i = \frac{\cos(\alpha_i) + \operatorname{tg} \varphi_i \sin(\alpha_i)}{f}$ .

6. Розв'язуємо рівняння (7) за допомогою методу ділення відрізка навпіл. Отримаємо початкове значення коефіцієнта стійкості.

7. Пошук мінімального коефіцієнта стійкості виконуємо згідно з алгоритмом 1, де цільовою функцією є  $P_i(y_i)$ , а роль змінних будуть виконувати координати поверхні ковзання  $U(x)$  по  $y$ , оскільки  $x$  фіксовані.

Схема розрахунку коефіцієнта стійкості і побудови критичної поверхні ковзання за методом Можевітінова [7].

1. Задаємо фізико-механічні характеристики ґрунту, а саме розміри ( $H$ ,  $L$ ,  $\alpha$ ), густину  $\rho$ , вологість  $W$ , і обчислюємо міцнісні характеристики  $c$  і  $\varphi$ .

2. Ґрунтовий схил розбиваємо опорними вертикалями  $x_k, k = 1..n$  і задаємо початкові координати кривої ковзання  $y_k(x_k)$ .

3. Знаходимо вагу кожного відріка  $P_i = \gamma \cdot S_i$ , де  $S_i$  – площа кожного відріка,  $\gamma = \rho \cdot g$  – питома вага ґрунту, довжина кривої ковзання в межах кожного відріка  $\Delta S_i$ .

4. Розраховуємо кути нахилу поверхні ковзання по горизонталі  $\alpha_i = \arccos\left(\frac{x_{i+1}-x_i}{\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}}\right)$  і знаходимо значення параметра  $\beta$  за формулою  $\beta = [\sum(a_i)\Delta x] / L$ , де  $\Delta x$  – різниця між двома опорними вертикалями,  $L$  – довжина схилу.

5. Обраховуємо коефіцієнт стійкості за формулою:

$$k_3 = \left[ \sum P_i \cos(\beta) \operatorname{tg}(\varphi_i + \beta - \alpha_i) + \sum \frac{c_i \cos \varphi_i \Delta S_i}{\cos(\varphi_i + \beta - \alpha_i)} \right] / \sum P_i \sin(\beta).$$

Мінімізація коефіцієнта стійкості відбувається згідно з алгоритмом 2, де в ролі цільової функції виступає  $k_3(y_k)$ , а змінні – це координати поверхні ковзання по  $y$ , оскільки  $x$  – фіксовані значення.

### Програмна реалізація та результати числових експериментів

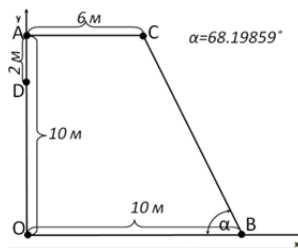


Рис. 6. Початкові дані

У результаті програмної реалізації було побудовано критичні поверхні ковзання для ґрунтового схилу на рис. 6.

Для сухого ґрунту отримано результати, які відображено на рис. 7 та 8. При цьому  $k_{min} = 3.20525$  для методу Федоровського, Курило та  $k_{min} = 5.23874$  для методу Можевітінова.

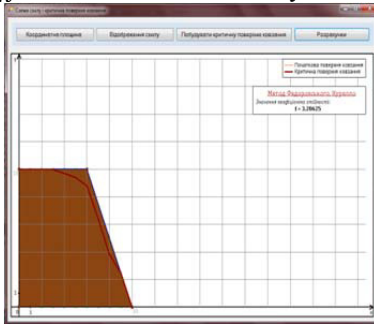


Рис. 7. Критична поверхня ковзання (метод Федоровського, Курило)

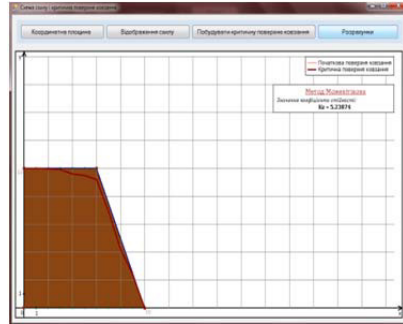


Рис. 8. Критична поверхня ковзання (метод Можевітінова)

Розглянуто випадок, коли в точці D відбувається витік чистої води. Задачу про розподіл вологи в ґрунті розв'язано в роботі [8]. Результати поверхонь ковзання представлено на рис. 9 та 10. При цьому  $k_{min} = 0.506796$  для методу Федоровського, Курило та  $k_{min} = 0.41246$  для методу Можевітінова.

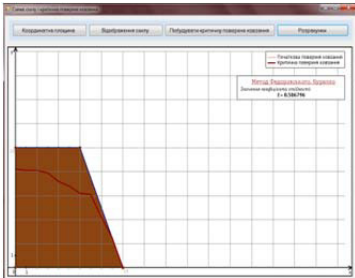


Рис. 9. Критична поверхня ковзання (метод Федоровського, Курило)

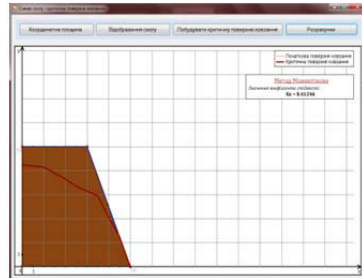


Рис. 10. Критична поверхня ковзання (метод Можевітінова)

## Висновки

Визначальне завдання цієї роботи полягало у розрахунку стійкості ґрунтових масивів на основі інженерних методів з урахуванням основних фізико-механічних характеристик порід. Було складено алгоритми для побудови критичної поверхні ковзання на основі інженерних методів і

розроблено програму для побудови цієї поверхні. Отримані дані дають підстави зробити такі висновки:

1. Аналіз наявних методів розрахунку стійкості ґрунтових масивів дав змогу встановити, що найвагомішими недоліками більшості інженерних методів є відсутність строгого аналізу напруженого стану приукісних областей та нерозробленість науково обґрунтованої методики побудови найбільш вірогідної поверхні ковзання. З огляду на це залишається актуальним подальший розвиток і розширення оптимізаційних методів у механіці ґрунтів.

2. Проаналізовано вплив зміни вологості на величину коефіцієнта стійкості та розміри поверхні ковзання.

3. Збільшення величини вологості по масиві ґрунту сприяє зменшенню стійкості укосу.

1. Гинзбург Л. К. Рекомендации по выбору методов расчета коэффициента устойчивости склона и оползневого давления / Л. К. Гинзбург. – М. : ЦБНТИ Минмонтаж-спецстроя СССР, 1986. – 124 с.
2. Федоровский В. Г. Метод расчета устойчивости откосов и склонов / В. Г. Федоровский, С. В. Курило // Геоэкология. – 1997. – №6. – С. 95–106.
3. Иванов П. Л. Ґрунти и основания гидротехнических сооружений : учеб. для гидротехн. спец. вузов / П. Л. Иванов. – М. : Высш. шк., 1985. – 352 с.
4. Можевитинов А. Л. Общий метод расчета устойчивости откосов земляных сооружений / А. Л. Можевитинов // Известия ВНИИГ, т. 92. – 1970. – С. 11–22.
5. Чугаев Р. Р. Земляные гидротехнические сооружения. Теоретические основы расчета / Р. Р. Чугаев. – Ленинград : Энергия, 1967. – 460 с.
6. Хуан. Я. Х. Устойчивость земляных откосов / Я. Х. Хуан. – М. : Стройиздат, 1988. – 206 с.
7. Калиткин Н. Н. Численные методы / Н. Н. Калиткин. – М. : Наука, 1978. – 512 с.
8. Громадченко Т. В. Математичне моделювання промочування сухого ґрунту в результаті аварії безнапірного водопроводу / Т. В. Громадченко, П. М. Мартинюк, А. В. Пащенко // Вісник ОДАБА. – 2012. – Вип. 48. Ч. 1. – С. 103–111.

**Брушковський О. Л., к.т.н., доцент, Дубчак І. В., асистент** (Національний університет водного господарства та природокористування, м. Рівне)

**ДО ВИЗНАЧЕННЯ ПОЛІВ НАПРУЖЕНЬ І ДЕФОРМАЦІЙ У ЕЛЕМЕНТАХ СИЛОВИХ КОНСТРУКЦІЙ НА ОСНОВІ ТЕОРЕТИКО-ЕКСПЕРИМЕНТАЛЬНОГО МЕТОДУ**

**Розглянуто задачу визначення полів деформацій і напружень в елементах конструкцій, що знаходяться у плоскому напруженому стані, за експериментально знайденими значеннями цих полів в окремих точках.**

**The problem of determining the tension and deformation fields in structural elements which are in the plane tense state has been examined with experimentally found values of these fields in the particular points.**

**Рассмотрена задача определения полей напряжений и деформаций в элементах конструкций, находящихся в плоском напряжённом состоянии, по экспериментально найденным значениям этих полей в отдельных точках.**

Загальним недоліком сучасних силових конструкцій є те, що вони пасивно сприймають навантаження, без пристосування до їхньої зміни. Пристосуванням конструкції будемо вважати її властивість переносити змінні зовнішні навантаження без перевищення заданого запасу міцності, якщо це можливо. Активною силовою конструкцією називають силову конструкцію, яка може бути пристосованою до змінних зовнішніх навантажень. Розробка таких конструкцій привертає все більшу увагу дослідників [1].

Для надійної роботи активних силових конструкцій за змінних умов експлуатації необхідно володіти інформацією про стан міцності її елементів у довільний момент часу. Але і в разі відомої програми навантаження недосконалість теоретичних та експериментальних методів визначення напружено-деформованого стану (НДС) силових елементів конструкцій не завжди дає змогу це зробити. Розширення можливостей обох методів досягають шляхом їхнього вдосконалення. Метою пропонованого дослідження є виявлення додаткових резервів для визначення полів напружень і деформацій в елементах силових конструкцій під час їхньої експлуатації, що можуть бути знайдені шляхом раціонального поєднання теоретичних та експериментальних методів.

Під час експериментальних досліджень напружено-деформованого стану елементів силової конструкції (літаків, машин, інженерних споруд) широко застосовуваною є тензOMETрія. Однак інформації, одержаної у результаті первинної обробки, недостатньо для висновку про міцність об'єкту з яким-небудь концентратором з огляду на малу ймовірність встановлення тензодатчиків у найбільш небезпечних місцях. У пропонованій роботі наведено методику визначення полів деформацій і напружень в елементах

силових конструкцій за їхніми відомими значеннями в окремих точках, знайденими на основі експериментальних методів, наприклад, за показами тензодатчиків. Питання первинної обробки показів не було розглянуто. Перехід від деформацій до напружень здійснено за допомогою відомих формул фізичного закону.

Розглянемо плоску задачу теорії пружності. Визначення полів напружень і деформацій за їхніми відомими значеннями в окремих точках може бути проведено без урахування і з частковим або повним урахуванням розрахункових схем досліджуваних об'єктів.

### **Визначення полів деформацій і напружень без урахування розрахункових схем досліджуваних об'єктів**

Для визначення функції деформації  $\varepsilon_r(s)$  на поверхні об'єкта може бути застосовано відомі алгоритми наближення функцій однієї або декількох змінних [2].

Поверхню апроксимації може складатись із частин площин, кожна з яких проведено за своєю сукупністю датчиків. Наприклад, рівняння кожної з них знаходимо за показами трьох тензодатчиків, що дає змогу визначити поле у внутрішніх точках цього базового трикутника й у деякому околі навколо нього. Використано рівняння площини, що проходить через три задані точки:

$$M_1(x_1; y_1; \varepsilon_1), M_2(x_2; y_2; \varepsilon_2), M_3(x_3; y_3; \varepsilon_3):$$

$$\begin{vmatrix} x - x_1 & y - y_1 & \varepsilon - \varepsilon_1 \\ x_2 - x_1 & y_2 - y_1 & \varepsilon_2 - \varepsilon_1 \\ x_3 - x_1 & y_3 - y_1 & \varepsilon_3 - \varepsilon_1 \end{vmatrix} = 0. \quad (1)$$

Краще проводити апроксимацію неперервною поверхнею, для чого область D, у якій буде проведено апроксимацію, розбиваємо на трикутники, у вершинах яких відомі експериментальні значення досліджуваної функції, а потім у кожному трикутнику проводимо лінійну апроксимацію.

Загальним недоліком вказаних методів є їхня погана чутливість до змін поля у точках, відмінних від експериментальних, через що реальну зміну поля не буде помічено, якщо її не відчули датчики, або згладжено, якщо її екстремальні значення знаходяться не у вузлах інтерполяції.

### **Визначення полів деформацій і напружень із частковим урахуванням розрахункових схем досліджуваних об'єктів**

Розглянемо більш точні методи, у яких враховано тією чи іншою мірою фізичний бік досліджуваного явища.

Під час часткового врахування розрахункової схеми можна використати ту властивість, що функції напружень і деформацій є бігармонічними. Тоді для

плоского напруженого стану деформації  $\varepsilon_r$  може бути представлено таким чином:

$$\varepsilon_r = \sum_{i=0}^n P_i(x, y), \quad (2)$$

де:

$$P_0(x, y) = a_{00}; P_1(x, y) = a_{10}x + a_{11}y;$$

$$P_2(x, y) = a_{20}x^2 + a_{21}xy + a_{22}y^2; \dots;$$

$$P_i(x, y) = a_{i0}x^i + a_{i1}x^{i-1}y + a_{i2}x^{i-2}y^2 + \dots + a_{ii}y^i.$$

Функції  $P_i(x, y)$  повинні задовольняти рівняння

$$\frac{\partial^4 P_i(x, y)}{\partial x^4} + 2 \frac{\partial^4 P_i(x, y)}{\partial x^2 \partial y^2} + \frac{\partial^4 P_i(x, y)}{\partial y^4} = 0. \quad (3)$$

Кожну з компонент тензорів напружень, або деформацій, також може бути представлено у вигляді степеневого ряду. Зв'язок між коефіцієнтами цих рядів знаходимо за допомогою рівнянь рівноваги або рівнянь нерозривності.

Рівняння рівноваги може бути враховано за допомогою функції напружень  $F(x, y)$ , представленої у вигляді ряду типу (2). Напруження за допомогою цієї функції визначають так:

$$\sigma_x = \frac{\partial^2 F}{\partial y^2}; \quad \sigma_y = \frac{\partial^2 F}{\partial x^2}; \quad \tau_{xy} = -\frac{\partial^2 F}{\partial x \partial y}. \quad (4)$$

Розглянемо, чи можна в разі врахування тільки рівнянь нерозривності досягти скорочення числа невідомих параметрів.

Виразимо, наприклад, деформації  $\varepsilon_x; \varepsilon_y; \gamma_{xy}$  через деяку функцію  $\psi(x, y)$  таким чином:

$$\varepsilon_x = \frac{\partial^2 \psi}{\partial y^2}; \quad \varepsilon_y = \frac{\partial^2 \psi}{\partial x^2}; \quad \gamma_{xy} = -2 \frac{\partial^2 \psi}{\partial x \partial y}. \quad (5)$$

Тоді рівняння нерозривності

$$\frac{\partial^2 \varepsilon_x}{\partial y^2} + \frac{\partial^2 \varepsilon_y}{\partial x^2} = \frac{\partial^2 \gamma_{xy}}{\partial x \partial y} \quad (6)$$

буде бігармонічним:

$$\frac{\partial^4 \psi}{\partial x^4} + 2 \frac{\partial^4 \psi}{\partial x^2 \partial y^2} + \frac{\partial^4 \psi}{\partial y^4} = 0. \quad (7)$$

Однак необхідно визначити, які обмеження на функцію  $\psi(x, y)$  накладають рівняння рівноваги. У нашому випадку:

$$\begin{cases} \sigma_x = \frac{E}{1 - \mu^2} (\varepsilon_x + \mu\varepsilon_y); \\ \sigma_y = \frac{E}{1 - \mu^2} (\varepsilon_y + \mu\varepsilon_x); \\ \tau_{xy} = \frac{E}{2(1 + \mu)} \gamma_{xy}. \end{cases} \Rightarrow \begin{cases} \sigma_x = \frac{E}{1 - \mu^2} \left( \frac{\partial^2 \psi}{\partial y^2} + \mu \frac{\partial^2 \psi}{\partial x^2} \right); \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} = 0; \\ \sigma_y = \frac{E}{1 - \mu^2} \left( \frac{\partial^2 \psi}{\partial x^2} + \mu \frac{\partial^2 \psi}{\partial y^2} \right); \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} = 0; \\ \tau_{xy} = -\frac{E}{(1 + \mu)} \frac{\partial^2 \psi}{\partial x \partial y}. \end{cases}$$

$$\begin{cases} \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = F_1(y); \\ \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = F_2(x); \end{cases} \Rightarrow \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = const; \Rightarrow \varepsilon_x + \varepsilon_y = const.$$

Таким чином, за допомогою вказаної функції можна описати тільки частковий випадок деформованого стану, коли перший інваріант тензору деформацій залишається сталим.

### **Визначення полів деформацій і напружень з повним урахуванням розрахункових схем досліджуваних об'єктів**

Під час визначення полів напружень і деформацій у деякій області  $\mathbf{D}$  з границею  $\Gamma$  за їхніми відомими значеннями в окремих точках, які знаходять експериментально, найбільш точні результати може бути одержано з урахуванням не тільки диференціальних рівнянь задачі, але й граничних умов на контурі  $\Gamma$ . Але у такому разі достатньо експериментально визначити тільки граничні умови задачі. Після цього задачу буде зведено до розв'язування крайової задачі теорії пружності із заданими (хоч і знайденими експериментально) граничними умовами. Для визначення граничних умов на контурі  $\Gamma$  може бути використано відомі формули інтерполяції функції однієї змінної [2, 3, 4, 5]. Вкажемо деякі алгоритми визначення полів деформацій і напружень в області  $\mathbf{D}$  з границею  $\Gamma$  з повним урахуванням розрахункової схеми досліджуваного об'єкта.



У загальному випадку ця крайова задача містить систему з 8 невідомими і граничні умови.

Рівняння рівноваги:

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} = 0; \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} = 0. \quad (8)$$

Рівняння нерозривності деформацій:

$$\frac{\partial^2 \varepsilon_x}{\partial y^2} + \frac{\partial^2 \varepsilon_y}{\partial x^2} = \frac{\partial^2 \gamma_{xy}}{\partial x \partial y}. \quad (9)$$

Фізичні рівняння:

$$\begin{aligned} \varepsilon_x &= \frac{1}{E}(\sigma_x - \mu\sigma_y); \varepsilon_y = \frac{1}{E}(\sigma_y - \mu\sigma_x); \\ \gamma_{xy} &= \frac{\tau_{xy}}{G} = \frac{2(1+\mu)}{E}\tau_{xy}. \end{aligned} \quad (10)$$

Геометричні рівняння:

$$\varepsilon_x = \frac{\partial u}{\partial x}; \varepsilon_y = \frac{\partial v}{\partial y}; \gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}. \quad (11)$$

Умови на контурі, які повинно бути знайдено експериментально:

$$\begin{aligned} p_{xv} &= \sigma_x \cdot \cos(\nu, x) + \tau_{xy} \cdot \cos(\nu, y); \\ p_{yv} &= \tau_{xy} \cdot \cos(\nu, x) + \sigma_y \cdot \cos(\nu, y). \end{aligned} \quad (12)$$

Розглянемо різні варіанти розв'язування задачі.

1. Розв'язання задачі у напруженнях. Диференціальні рівняння задачі мають вигляд:

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} = 0; \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} = 0; \Delta(\sigma_x + \sigma_y) = 0, \quad (13)$$

де  $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$  – оператор Лапласа.

Експериментально повинно бути визначено нормальне та дотичне навантаження на контурі  $p_{xv}$  і  $p_{yv}$ .

2. Алгоритм, який використовує попереднє визначення першого інваріанта тензора деформацій у досліджуваній області.

Як відомо, перший інваріант тензора деформацій  $\varepsilon^I$

$$\varepsilon^I = \varepsilon_x + \varepsilon_y \quad (14)$$

задовольняє у досліджуваній області рівняння Лапласа:

$$\Delta \varepsilon^I = 0. \quad (15)$$

Граничні умови на контурі  $\Gamma$  мають вигляд

$$\varepsilon^I|_{\Gamma} = f(t) \quad (16)$$

і мають бути визначені експериментально. Частково для круга  $(O, R)$  розв'язком у полярних координатах буде інтеграл Пуассона.

Знаючи поле  $\varepsilon^I$  в області  $D$  та інтегруючи рівняння,

$$\begin{aligned} \Delta \varepsilon_x &= \frac{1}{(1-\mu)} \left( \frac{\partial^2 \varepsilon^I}{\partial y^2} - \mu \frac{\partial^2 \varepsilon^I}{\partial x^2} \right); \\ \Delta \varepsilon_y &= \frac{1}{(1-\mu)} \left( \frac{\partial^2 \varepsilon^I}{\partial x^2} - \mu \frac{\partial^2 \varepsilon^I}{\partial y^2} \right); \\ \Delta \tau_{xy} &= -\frac{2(1+\mu)}{1-\mu} \frac{\partial^2 \varepsilon^I}{\partial x \partial y}, \end{aligned} \quad (17)$$

визначасмо поля деформацій в області  $D_1 \subset D$  з границею  $\Gamma_1$ . Функції  $\varepsilon_x(t)$ ,  $\varepsilon_y(t)$ ,  $\gamma_{xy}(t)$  на контурі  $\Gamma_1$  повинно бути знайдено експериментально.

3. Можна визначати поля деформацій і напружень з урахуванням, що компоненти тензорів деформацій і напружень, а також переміщення, є бігармонічними функціями. У такому разі для формування крайових умов на контурі  $\Gamma$  має бути знайдено експериментально значення вказаних функцій і, наприклад, їхніх похідних по нормалі до контура. Таким чином, особливості поєднання теоретичного й експериментального методів дають змогу визначити кожну з вказаних функцій незалежно від інших.

4. Об'єм необхідної експериментальної інформації буде зменшено, якщо використати функцію напружень  $F(x, y)$ , яка задовольняє рівняння:

$$\Delta \Delta F = 0. \quad (18)$$

Її можна знайти також шляхом розв'язування системи рівнянь:

$$\{ \Delta \sigma^I = 0; \Delta F(x, y) = \sigma^I \}. \quad (19)$$

На контурі  $\Gamma$  експериментально обраховуємо нормальне і дотичне навантаження. Надалі, використовуючи різні аналогії, за допомогою відомих методів визначасмо граничні умови для функції напружень.

Напруження через цю функцію обчислюють за формулами (4).

5. Задачу може бути розв'язано і на основі методу переміщень із використанням рівнянь Ляме:

$$(\lambda + G) \frac{\partial \varepsilon^I}{\partial x} + G \Delta u = 0; (\lambda + G) \frac{\partial \varepsilon^I}{\partial y} + G \Delta v = 0; \quad (20)$$

де  $\lambda = 2\mu G / (1 - 2\mu)$  (стала Ляме). Функції переміщень на контурі  $\Gamma_1$  повинно бути знайдено експериментально.

Отже, у роботі на основі теоретико-експериментального методу розглянуто задачу визначення полів деформацій і напружень в елементах активних силових конструкцій, що знаходяться у плоскому напруженому стані, за експериментально знайденими значеннями цих полів в окремих точках. При цьому зовнішні навантаження на конструкцію можуть бути відомими, частково відомими або зовсім невідомими.

**Висновок.** Під час використання наближених методів для визначення полів напружень і деформацій за їхніми відомими значеннями в окремих точках необхідно виявляти обмеження, які накладає вибір функцій, використовуваних для апроксимації, на досліджувані поля. У разі не виявлення згаданих обмежень результати досліджень і висновки на їхній основі можуть бути помилковими. Аналіз досліджень [2, 3, 4, 5] дає підстави зробити висновок про раціональність визначення полів напружень і деформацій з повним урахуванням розрахункових схем досліджуваних об'єктів, а також використання експериментальних значень для знаходження крайових умов відповідних диференціальних рівнянь або їхніх систем та для контролю визначених таким чином полів.

1. Брушковский А. Л. К расчёту на прочность активных силовых элементов, связей и конструкций / А. Л. Брушковский // Укр. ин-т инж. водн. хоз-ва. – Ровно, 1989. – 21 с. – Библиогр. 9 назв. – Рус.-Деп. в УкрНИИТИ 30.03.89, № 904-Ук89.
2. Брушковский А. Л. Плоская задача теории упругости с определением граничных условий по показаниям тензодатчиков / А. Л. Брушковский // Изв. вузов. Машиностроение. – 1985. – № 9. – С. 9–12.
3. Брушковский А. Л. Решение плоской задачи теории упругости для прямоугольной полосы методом замены / А. Л. Брушковский, А. И. Макеев // Прочность конструкций летательных аппаратов : сб. науч. тр. – Харьков : ХГУ, 1973. – С. 96–105.
4. Брушковский А. Л. Восстановление полей деформаций в районе концентраторов по показаниям тензодатчиков / А. Л. Брушковский // Авиационная техника : сб. науч. тр. – Казань, 1978. – С. 142–144.
5. Брушковский А. Л. Определение полей деформаций в районе концентраторов по показаниям тензодатчиков для ортотропных элементов / А. Л. Брушковский, Л. П. Литвиненко // Прикладная механика. – Том 18. – № 1. – 1982. – С. 82–85.

**Власюк А. П., професор, д.т.н., Гульчук В. А., магістр ФПМ і КІС**  
 (Національний університет водного господарства та природокористування,  
 м. Рівне)

**ПОБУДОВА МОНОТОННИХ РІЗНИЦЕВИХ СХЕМ ДЛЯ  
 НЕЛІНІЙНОГО БАГАТОВИМІРНОГО РІВНЯННЯ ПАРАБОЛІЧНОГО  
 ТИПУ ТА НЕЛІНІЙНИХ СИСТЕМ РІВНЯНЬ ПАРАБОЛІЧНОГО  
 ТИПУ**

**Побудовано монотонну різницеву схему для тривимірного нелінійного рівняння параболічного типу. Показано, що вона апроксимує рівняння з точністю  $O(\tau + h_1^2 + h_2^2 + h_3^2)$ . Здійснено числові обчислення для граничних умов I-го роду.**

**Monotonic difference scheme for three dimensional nonlinear parabolic equation is constructed. It is shown that approximation order of this scheme is determined by  $O(\tau + h_1^2 + h_2^2 + h_3^2)$  term. The first type boundary conditions are used for numerical experiments.**

На сьогодні розв’язування задач тепло-масоперенесення в ґрунтових середовищах, хімічної кінетики, плазмі передбачає числове розв’язування крайових задач для нелінійного багатовимірного рівняння параболічного типу, що містить конвективні члени.

Розглянемо крайову задачу для тривимірного нелінійного рівняння параболічного типу, що містить перші похідні

$$q(X, t) \frac{\partial u}{\partial t} = \frac{\partial \left( p_x(X, u) \frac{\partial u}{\partial x} \right)}{\partial x} + \frac{\partial \left( p_y(X, u) \frac{\partial u}{\partial y} \right)}{\partial y} + \tag{1}$$

$$+ \frac{\partial \left( p_z(X, u) \frac{\partial u}{\partial z} \right)}{\partial z} + r_x(X, u) \frac{\partial u}{\partial x} + r_y(X, u) \frac{\partial u}{\partial y} + r_z(X, u) \frac{\partial u}{\partial z} + f(X, u),$$

$$u(X, 0) = U_0(X), \tag{2}$$

$$u(0, y, z, t) = U_1(y, z, t), \quad 0 \leq t \leq T, \quad X \in \Gamma_1, \tag{3}$$

$$u(l, y, z, t) = U_3(y, z, t), \quad 0 \leq t \leq T, \quad X \in \Gamma_2, \tag{4}$$

$$u(x, 0, z, t) = U_4(x, z, t), \quad 0 \leq t \leq T, \quad X \in \Gamma_3, \tag{5}$$

$$u(x, l, z, t) = U_5(x, z, t), \quad 0 \leq t \leq T, \quad X \in \Gamma_4, \quad (6)$$

$$u(x, y, 0, t) = U_6(x, y, t), \quad 0 \leq t \leq T, \quad X \in \Gamma_5, \quad (7)$$

$$u(x, y, l, t) = U_7(x, y, t), \quad 0 \leq t \leq T, \quad X \in \Gamma_5, \quad (8)$$

де  $X = (x, y, z)$ ,  $\Gamma_i$ ,  $i = \overline{1,6}$  межі області,  $p_x(X, u)$ ,  $p_y(X, u)$ ,  $p_z(X, u)$ ,  $r_x(X, u)$ ,  $r_y(X, u)$ ,  $r_z(X, u)$ ,  $q(X, t)$ ,  $f(X, u)$ , – досить гладкі функції.

Для крайової задачі (1)–(8) отримано числовий розв’язок із використанням монотонної різничевої схеми [1] та локально-одновимірного методу О. А. Самарського [2].

Розщеплені крайові задачі розв’язуємо шляхом використання монотонної різничевої схеми з порядком апроксимації  $O(h_i^2 + \tau)$ ,  $i = \overline{1,3}$ .

Загалом задача (1)–(8) володіє порядком апроксимації  $O(h_1^2 + h_2^2 + h_3^2 + \tau)$ .

Канонічний вигляд різничевої схеми по змінній  $X$  є таким:

$$\left( \frac{\mu_{x,i}^s d_{x,i}^s}{h_1^2} + \frac{\mu_{x,i+1}^s d_{x,i+1}^s}{h_1^2} + \frac{|r_x^s|}{h_1} + \frac{q_i^s}{\tau} \right) U_i^{s+\frac{1}{3}} =$$

$$\left( \frac{\mu_{x,i}^s d_{x,i}^s}{h_1^2} - \frac{r_{x,i-}^s}{h_1} \right) U_{i-1}^{s+\frac{1}{3}} + \left( \frac{\mu_{x,i+1}^s d_{x,i+1}^s}{h_1^2} + \frac{r_{x,i+}^s}{h_1} \right) U_{i+1}^{s+\frac{1}{3}} + \frac{q_i^s U_i^s}{\tau} + \frac{f_i^s}{3}.$$

Прогоночний вигляд цієї різничевої схеми є таким:

$$a_{x,i}^s U_{i-1}^{s+\frac{1}{3}} - c_{x,i}^s U_i^{s+\frac{1}{3}} + b_{x,i}^s U_{i+1}^{s+\frac{1}{3}} = F_{x,i}^s, \quad (9)$$

$$U_0^{s+1} = \chi_{x,1} U_1^{s+1} + \mu_{x,1}, \quad U_n^{s+1} = \chi_{x,2} U_{n-1}^{s+1} + \mu_{x,2}, \quad (10)$$

де

$$a_{x,i}^s = \frac{\mu_{x,i}^s d_{x,i}^s}{h_1^2} - \frac{r_{x,i-}^s}{h_1}, \quad b_{x,i}^s = \frac{\mu_{x,i+1}^s d_{x,i+1}^s}{h_1^2} + \frac{r_{x,i+}^s}{h_1},$$

$$c_{x,i}^s = \frac{1}{h_1^2} (\mu_{x,i}^s d_{x,i}^s + \mu_{x,i+1}^s d_{x,i+1}^s) + \frac{|r_x^s|}{h_1} + \frac{q_i^s}{\tau},$$

$$F_{x,i}^s = -\frac{q_i^s U_i^s}{\tau} - \frac{1}{3} f_i^s.$$

Розв'язки знаходимо у такому вигляді:

$$U_i^{s+\frac{1}{3}} = \alpha_{i+1} U_{i+1}^{s+\frac{1}{3}} + \beta_{i+1}, \quad (11)$$

де прогоночні коефіцієнти рівні

$$\alpha_{i+1} = \frac{b_{x,i}^s}{c_{x,i}^s - \alpha_{i+1} a_{x,i}^s}, \quad \beta_{i+1} = \frac{a_{x,i}^s \beta_i + F_{x,i}^s}{c_{x,i}^s - \alpha_{i+1} a_{x,i}^s}. \quad (12)$$

Для проведення прогонки використаємо граничні умови:

$$\alpha_1 = \chi_{x,1}, \quad \beta_1 = \mu_{x,1}. \quad (13)$$

На правій стороні грані по змінній  $x$  значення обчислюємо за формулою

$$U_n^{s+\frac{1}{3}} = \frac{\chi_{x,2} \beta_n + \mu_{x,2}}{1 - \alpha_n \chi_{x,2}}.$$

Канонічний вигляд різницевої схеми для змінної  $y$  є таким:

$$\left( \frac{\mu_{y,j}^{s+\frac{1}{3}} d_{y,j}^{s+\frac{1}{3}}}{h_2^2} + \frac{\mu_{y,j+1}^{s+\frac{1}{3}} d_{y,j+1}^{s+\frac{1}{3}}}{h_2^2} + \frac{|r_y^{s+\frac{1}{3}}|}{h_2} + \frac{q_j^{s+\frac{1}{3}}}{\tau} \right) U_j^{s+\frac{2}{3}} = \left( \frac{\mu_{y,j}^{s+\frac{1}{3}} d_{y,j}^{s+\frac{1}{3}}}{h_2^2} - \frac{r_{y,j-}^{s+\frac{1}{3}}}{h_2} \right) U_{j-1}^{s+\frac{2}{3}} + \left( \frac{\mu_{y,j+1}^{s+\frac{1}{3}} d_{y,j+1}^{s+\frac{1}{3}}}{h_2^2} + \frac{r_{y,j+}^{s+\frac{1}{3}}}{h_2} \right) U_{j+1}^{s+\frac{2}{3}} + \frac{q_j^{s+\frac{1}{3}} U_j^{s+\frac{1}{3}}}{\tau} + \frac{f_j^{s+\frac{1}{3}}}{3}.$$

Прогоночний вигляд цієї різницевої схеми такий

$$a_{y,j}^{s+\frac{1}{3}} U_{j-1}^{s+\frac{2}{3}} - c_{y,j}^{s+\frac{1}{3}} U_j^{s+\frac{2}{3}} + b_{y,j}^{s+\frac{1}{3}} U_{j+1}^{s+\frac{2}{3}} = F_{y,j}^{s+\frac{1}{3}}, \quad (14)$$

$$U_0^{s+1} = \chi_{y,1} U_1^{s+1} + \mu_{y,1}, \quad U_m^{s+1} = \chi_{y,2} U_{m-1}^{s+1} + \mu_{y,2}, \quad (15)$$

де

$$a_{y,j}^{s+\frac{1}{3}} = \frac{\mu_{y,j}^{s+\frac{1}{3}} d_{y,j}^{s+\frac{1}{3}}}{h_2^2} - \frac{r_{y,j-}^{s+\frac{1}{3}}}{h_2}, \quad b_{y,j}^{s+\frac{1}{3}} = \frac{\mu_{y,j+1}^{s+\frac{1}{3}} d_{y,j+1}^{s+\frac{1}{3}}}{h_2^2} + \frac{r_{y,j+}^{s+\frac{1}{3}}}{h_2},$$

$$c_{y,j}^{s+\frac{1}{3}} = \frac{1}{h_2^2} \left( \mu_{y,j}^{s+\frac{1}{3}} d_{y,j}^{s+\frac{1}{3}} + \mu_{y,j+1}^{s+\frac{1}{3}} d_{y,j+1}^{s+\frac{1}{3}} \right) + \frac{\left| r_y^{s+\frac{1}{3}} \right|}{h_2} + \frac{q_j^{s+\frac{1}{3}}}{\tau},$$

$$F_{y,j}^{s+\frac{1}{3}} = -\frac{q_j^{s+\frac{1}{3}} U_j^{s+\frac{1}{3}}}{\tau} - \frac{1}{3} f_j^{s+\frac{1}{3}}.$$

Розв'язки знаходимо у такому вигляді:

$$U_j^{s+\frac{2}{3}} = \alpha_{j+1} U_{j+1}^{s+\frac{2}{3}} + \beta_{j+1}, \quad (16)$$

де прогоночні коефіцієнти рівні

$$\alpha_{j+1} = \frac{b_{y,j}^{s+\frac{1}{3}}}{c_{y,j}^{s+\frac{1}{3}} - \alpha_{j+1} a_{y,j}^{s+\frac{1}{3}}}, \quad \beta_{j+1} = \frac{a_{y,j}^{s+\frac{1}{3}} \beta_j + F_{y,j}^{s+\frac{1}{3}}}{c_{y,j}^{s+\frac{1}{3}} - \alpha_{j+1} a_{y,j}^{s+\frac{1}{3}}}. \quad (17)$$

Для проведення прогонки використаємо граничні умови:

$$\alpha_1 = \chi_{y,1}, \quad \beta_1 = \mu_{y,1}. \quad (18)$$

На правій стороні грані по змінній  $y$  значення обчислюємо за формулою:

$$U_m^{s+\frac{2}{3}} = \frac{\chi_{y,2} \beta_m + \mu_{y,2}}{1 - \alpha_m \chi_{y,2}}.$$

Канонічний вигляд різницевої схеми для змінної  $y$  є таким:

$$\left( \frac{\mu_{z,k}^{s+\frac{2}{3}} d_{z,k}^{s+\frac{2}{3}}}{h_3^2} + \frac{\mu_{z,k+1}^{s+\frac{2}{3}} d_{z,k+1}^{s+\frac{2}{3}}}{h_3^2} + \frac{\left| r_z^{s+\frac{2}{3}} \right|}{h_3} + \frac{q_k^{s+\frac{2}{3}}}{\tau} \right) U_k^{s+1} = \left( \frac{\mu_{z,k}^{s+\frac{2}{3}} d_{z,k}^{s+\frac{2}{3}}}{h_3^2} - \frac{r_{z,k-}^{s+\frac{2}{3}}}{h_3} \right) U_{k-1}^{s+1} +$$

$$+ \left( \frac{\mu_{z,k+1}^{s+\frac{2}{3}} d_{z,k+1}^{s+\frac{2}{3}}}{h_3^2} + \frac{r_{z,k+}^{s+\frac{2}{3}}}{h_3} \right) U_{k+1}^{s+1} + \frac{q_k^{s+\frac{2}{3}} U_k^{s+\frac{2}{3}}}{\tau} + \frac{f_k^{s+\frac{2}{3}}}{3}.$$

Прогоночний вигляд цієї різницевої схеми є таким:

$$a_{z,k}^{s+\frac{2}{3}} U_{k-1}^{s+1} - c_{z,k}^{s+\frac{2}{3}} U_k^{s+1} + b_{z,k}^{s+\frac{2}{3}} U_{k+1}^{s+1} = F_{z,k}^{s+\frac{2}{3}} \quad (19)$$

$$U_{i_0}^{s+1} = \chi_{z,1} U_1^{s+1} + \mu_{z,1}, \quad U_l^{s+1} = \chi_{z,2} U_{l-1}^{s+1} + \mu_{z,2}, \quad (20)$$

де

$$a_{z,k}^{s+\frac{2}{3}} = \frac{\mu_{z,k}^{s+\frac{2}{3}} d_{z,k}^{s+\frac{2}{3}}}{h_3^2} - \frac{r_{z,k}^{s+\frac{2}{3}}}{h_3}, \quad b_{z,k}^{s+\frac{2}{3}} = \frac{\mu_{z,k+1}^{s+\frac{2}{3}} d_{y,k+1}^{s+\frac{2}{3}}}{h_3^2} + \frac{r_{z,k+1}^{s+\frac{2}{3}}}{h_3},$$

$$c_{z,k}^{s+\frac{2}{3}} = \frac{1}{h_3^2} \left( \mu_{z,k}^{s+\frac{2}{3}} d_{z,k}^{s+\frac{2}{3}} + \mu_{z,k+1}^{s+\frac{2}{3}} d_{z,k+1}^{s+\frac{2}{3}} \right) + \frac{\left| r_z^{s+\frac{2}{3}} \right|}{h_3} + \frac{q_k^{s+\frac{2}{3}}}{\tau},$$

$$F_{z,k}^{s+\frac{2}{3}} = -\frac{q_k^{s+\frac{2}{3}} U_k^{s+\frac{2}{3}}}{\tau} - \frac{1}{3} f_k^{s+\frac{2}{3}}.$$

Розв'язки знаходимо у такому вигляді:

$$U_k^{s+1} = \alpha_{k+1} U_{k+1}^{s+1} + \beta_{k+1}, \quad (21)$$

де прогоночні коефіцієнти дорівнюють:

$$\alpha_{k+1} = \frac{b_{y,k}^{s+\frac{2}{3}}}{c_{z,k}^{s+\frac{2}{3}} - \alpha_{k+1} a_{z,k}^{s+\frac{2}{3}}}, \quad \beta_{k+1} = \frac{a_{z,k}^{s+\frac{2}{3}} \beta_k + F_{z,k}^{s+\frac{2}{3}}}{c_{z,k}^{s+\frac{2}{3}} - \alpha_{k+1} a_{z,k}^{s+\frac{2}{3}}}. \quad (22)$$

Для проведення прогонки використаємо граничні умови:

$$\alpha_1 = \chi_{z,1}, \quad \beta_1 = \mu_{z,1}. \quad (23)$$

На правій стороні грані по змінній  $z$  значення обчислюємо за формулою:

$$U_l^{s+1} = \frac{\chi_{z,2} \beta_l + \mu_{z,2}}{1 - \alpha_l \chi_{z,2}}.$$

$\chi_{x,1}, \chi_{x,2}, \mu_{x,1}, \mu_{x,2}, \chi_{y,1}, \chi_{y,2}, \mu_{y,1}, \mu_{y,2}, \chi_{z,1}, \chi_{z,2}, \mu_{z,1}, \mu_{z,2}$

вибираємо згідно з граничними умовами:

$$\chi_{x,1} = 0, \mu_{x,1} = \mu_{0,jk}^{s+1} (Y_{jk}, (k+1)\tau), \chi_{x,2} = 0, \mu_{x,2} = \mu_{n,jk}^{s+1} (Y_{jk}, (k+1)\tau),$$

$$\chi_{y,1} = 0, \mu_{y,1} = \mu_{i_0,k}^{s+1} (Y_{ik}, (k+1)\tau), \chi_{y,2} = 0, \mu_{y,2} = \mu_{im,k}^{s+1} (Y_{ik}, (k+1)\tau),$$



$$\chi_{z,1} = 0, \mu_{z,1} = \mu_{ij,l}^{s+1}(Y_{ij}, (k+1)\tau), \chi_{z,2} = 0, \mu_{z,2} = \mu_{ij,l}^{s+1}(Y_{ij}, (k+1)\tau).$$

Порядок апроксимації цієї схеми  $O(\tau + h_1^2 + h_2^2 + h_3^2)$ .

Для доведення стійкості схеми змінних напрямків використовуємо принцип сумарної апроксимації.

Рівняння

$$q(X, t) \frac{\partial u}{\partial t} = \frac{\partial \left( p_x(X, u) \frac{\partial u}{\partial x} \right)}{\partial x} + \frac{\partial \left( p_y(X, u) \frac{\partial u}{\partial y} \right)}{\partial y} + \frac{\partial \left( p_z(X, u) \frac{\partial u}{\partial z} \right)}{\partial z} + r_x(X, u) \frac{\partial u}{\partial x} + r_y(X, u) \frac{\partial u}{\partial y} + r_z(X, u) \frac{\partial u}{\partial z} + f(X, u),$$

розщепимо на три:

$$\frac{1}{3} q(X, t) \frac{\partial u}{\partial t} = \frac{\partial \left( p_x(X, u) \frac{\partial u}{\partial x} \right)}{\partial x} + r_x(X, u) \frac{\partial u}{\partial x} + \frac{f(X, u)}{3},$$

$$\frac{1}{3} q(X, t) \frac{\partial u}{\partial t} = \frac{\partial \left( p_y(X, u) \frac{\partial u}{\partial y} \right)}{\partial y} + r_y(X, u) \frac{\partial u}{\partial y} + \frac{f(X, u)}{3},$$

$$\frac{1}{3} q(X, t) \frac{\partial u}{\partial t} = \frac{\partial \left( p_z(X, u) \frac{\partial u}{\partial z} \right)}{\partial z} + r_z(X, u) \frac{\partial u}{\partial z} + \frac{f(X, u)}{3},$$

з похибками апроксимації

$$\psi_1 = -\frac{1}{3} q(X, t) \frac{\partial u}{\partial t} + \frac{\partial \left( p_x(X, u) \frac{\partial u}{\partial x} \right)}{\partial x} + r_x(X, u) \frac{\partial u}{\partial x} + \frac{f(X, u)}{3} + O(\tau + h_1^2),$$

$$\psi_2 = -\frac{1}{3} q(X, t) \frac{\partial u}{\partial t} + \frac{\partial \left( p_y(X, u) \frac{\partial u}{\partial y} \right)}{\partial y} + r_y(X, u) \frac{\partial u}{\partial y} + \frac{f(X, u)}{3} + O(\tau + h_2^2),$$

$$\psi_3 = -\frac{1}{3} q(X, t) \frac{\partial u}{\partial t} + \frac{\partial \left( p_z(X, u) \frac{\partial u}{\partial z} \right)}{\partial z} + r_z(X, u) \frac{\partial u}{\partial z} + \frac{f(X, u)}{3} + O(\tau + h_3^2),$$

що в сумі  $\psi = \psi_1 + \psi_2 + \psi_3$  збігаються зі швидкістю  $O(\tau + h_1^2 + h_2^2 + h_3^2)$ .

Програмну реалізацію обчислювального алгоритму здійснено в середовищі Borland Delphi 7.0.

Числові експерименти проведено для такої початкової умови

$$U(X,0) = 0 \text{ } \mathcal{Z}/\text{літпру}$$

та граничних умов

$$\begin{aligned} u(0, y, z, t) &= 35 \text{ } \mathcal{Z}/\text{літпру}, & 0 \leq t \leq T, & & X \in \Gamma_1, \\ u(l, y, z, t) &= 0.02 \text{ } \mathcal{Z}/\text{літпру}, & 0 \leq t \leq T, & & X \in \Gamma_2, \\ u(x, 0, z, t) &= 35 \text{ } \mathcal{Z}/\text{літпру}, & 0 \leq t \leq T, & & X \in \Gamma_3, \\ u(x, l, z, t) &= 35 \text{ } \mathcal{Z}/\text{літпру}, & 0 \leq t \leq T, & & X \in \Gamma_4, \\ u(x, y, 0, t) &= 35 \text{ } \mathcal{Z}/\text{літпру}, & 0 \leq t \leq T, & & X \in \Gamma_5, \\ u(x, y, l, t) &= 35 \text{ } \mathcal{Z}/\text{літпру}, & 0 \leq t \leq T, & & X \in \Gamma_6, \end{aligned}$$

за таких вихідних даних

$$\begin{aligned} l_1 = l_2 = 30\text{м}, l_3 = 10\text{м} \quad T = 120\text{дiб}, \tau = 30\text{дiб}, r_x(X, u) &= -\frac{1}{T_p}, \\ r_y(X, u) &= -\frac{1}{T_p}, r_z(X, u) = -\frac{1}{T_p}, p_x(X, u) = Dm + \lambda|V(X, u)|, \\ p_y(X, u) &= Dm + \lambda|V(X, u)|, p_z(X, u) = Dm + \lambda|V(X, u)|, \sigma = 0.5, \\ q_x(X, t) &= \frac{\sigma}{V(X, u)T^2}, f(X, u) = -\frac{\gamma(u - u_m)}{V(X, u)T^2}, Dm = 0.02 \text{ } \mathcal{M}^2/\text{дoбy}, \\ T_p &= 10\text{м}, \gamma = 0.00065, u_m = 35 \text{ } \mathcal{Z}/\text{літпру}, \lambda = 5\text{м}, H_1 = 15\text{м}, H_2 = 1\text{м}, \\ r_1 &= 0.3\text{м}, r_2 = 100.3\text{м}, \bar{k} = 1\text{м}/\text{дoбy}, V(X, u) = \frac{k^2(u)}{\bar{k}^2} \frac{Q^2}{4\pi^2} e^{\left(\frac{4\pi(h_i + \varphi_0)}{Q}\right)}, i = \overline{0, n} \\ Q &= \frac{2\pi(\varphi_1 - \varphi_2)}{\ln \frac{r_2}{r_1}}, \varphi_0 = \frac{(\varphi_2 - \varphi_1) \ln r_1}{\ln \frac{r_2}{r_1}}, \varphi_1 = -\bar{k}T_p H_1, \varphi_2 = -\bar{k}T_p H_2, \\ a_0 &= 1.0054e - 3, a_1 = 1.0563e - 2, a_2 = -7.4311e - 2, a_3 = 1.7051e - 1, \\ a_4 &= -1.6703e - 1, a_5 = 5.9494e - 2, \end{aligned}$$

коли коефіцієнт фільтрації має таку залежність від концентрації

$$k(X, u) = a_0 + a_1 u + a_2 u^2 + a_3 u^3 + a_4 u^4 + a_5 u^5.$$

Розподіл показано на рис.

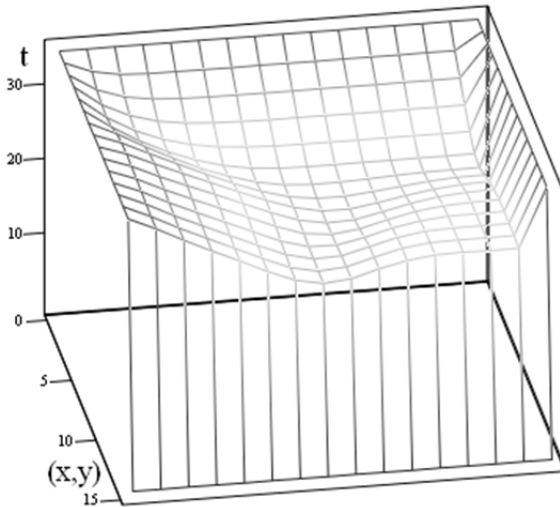


Рис. Розподіл концентрації у тривимірній області

### **Практичне значення отриманих результатів.**

Побудовано монотонні різницеві схеми та розроблено відповідне програмне забезпечення, яке може бути ефективно використане під час розв'язування нелінійних крайових задач параболічного типу, коли враховано конвективний тепло-масообмін.

Показано, що для побудованої різницевої схеми, для аналізованих задач, справедливий принцип максимуму. Представлено, що побудована різницева схема володіє принципом сумарної апроксимації. Також встановлено автоматичне задоволення виконання граничних умов різницевої схеми на цілих часових шарах.

Для побудованої різницевої схеми проведено порівняльний аналіз шляхом проведення числових експериментів за однакових вихідних даних із відомою в літературі різницевою схемою.

1. Власюк А. П. Монотонна різницева схема для нелінійного одновимірному рівняння параболічного типу, що містить першу похідну / А. П. Власюк, О. П. Остапчук, О. М. Степанченко // Вісник Київського університету. Серія фізико-математичних наук. – 2005. – Вип. 2. – С. 188–197. 2. Самарский А. А. Численные методы : учебное пособие для вузов / А. А. Самарский, А. В. Гулин. – М. : Наука, 1989. – 432 с.

Степанченко О. М., к.т.н., ст. викл., Мартинюк А. О., студент 5 курсу  
ННІ АКОТ (Національний університет водного господарства та  
природокористування, м. Рівне)

## **РОЗРОБКА ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ ПОШУКУ ОПТИМАЛЬНОЇ ПОЗИЦІЇ ТА МАНЕВРА ВІЙСЬК**

Запропоновано методи прийняття рішень, методи пошуку оптимального шляху в графах та методи ранжування для пошуку оптимального маневра військового підрозділу за різними параметрами: довжиною шляху, прохідністю та часом. У ході вирішення тактичних завдань, а особливо під час бойових дій виникають ситуації, коли потрібно знайти найбільш оптимальний маршрут: від цього залежить успіх ведення бою. Зроблено висновки про вибір оптимальної позиції та маневра військ. Ключові слова: прийняття рішень, граф, ранжування, ступінь важливості, коефіцієнт прохідності, пошук оптимального шляху, оптимальність.

There have been proposed decision making methods, shortest path problem in a graph and ranking methods. There have been considered a path finding methods for maneuver of army by different parameters: path length, passability and time. In solving tactical problems, especially during the battle actions there are situations when we need to find the optimal route: what determines the success of the battle. The conclusions about the choice of optimal position and maneuver of army have been made. Key words: decision making methods, graph, ranking methods, degree of importance, coefficient of passability, shortest path problem, optimality.

**Вступ.** Вагомою складовою організації й ведення бойових дій є орієнтування на місцевості. Останнє необхідне під час постановки бойових завдань підрозділам і вогневим засобам, для витримування напрямку дій, цілевказівки, нанесення на робочу карту результатів розвідки супротивника й місцевості, керування підрозділами в ході бою. Втрата орієнтування в бою може призвести до невиконання бойового завдання й невиправданих втрат особового складу й техніки. Тому вміння швидко й точно орієнтуватися на місцевості в будь-яких умовах є одним із найважливіших елементів польової виучки офіцерів [2–6].

Орієнтування на місцевості полягає у визначенні напрямків та сторін горизонту, свого місця розташування щодо навколишніх місцевих предметів і форм рельєфу, у витримуванні заданого або обраного маршруту руху, а

також у визначенні розташування на місцевості орієнтирів, рубежів своїх військ і військ супротивника, інженерних споруджень та інших об'єктів.

Під час маневрування військ і будь-якої іншої великої групи людей на місцевості, й у ході пошуку оптимального маршруту варто враховувати спектр різноманітних факторів, зокрема: складність ландшафту, матеріальні затрати у процесі пересування великої групи людей, втрати моральної та фізичної сили військ, що й вимагає автоматизації розв'язку вищезгаданого типу задач [1].

На сьогодні питання пошуку оптимального шляху з урахуванням ландшафтних умов та різноманітних додаткових параметрів, зокрема думки експертів та планів командування, а також питання побудови математичних моделей цієї задачі та створення програмного продукту, який би міг моделювати і прогнозувати ці процеси, постають як особливо актуальні.

**Постановка задачі.** Необхідним є створення програмного продукту, який би уможлиблював побудову схем руху військ на картах та моделювання оптимального руху військових підрозділів згідно з побудованою схемою.

У програмі повинно бути передбачено такі можливості:

- створення схеми шляхом розміщення на карті військових об'єктів та об'єктів ландшафту (річок, озер, боліт, лісопосадок та ін.);
- налаштування властивостей об'єктів та визначення напрямків і параметрів руху військ;
- накладання карт місцевості та введення розміру ділянки;
- експортування об'єктів створеної схеми у файл та завантаження з файлів усіх об'єктів;
- зберігання створеної схеми як зображення;
- пошук оптимального маршруту руху військ;
- ведення статистики по маршрутах;
- порівняння тактичних дій двох армій.

**Математичний розв'язок задачі.** Маршрути зручно задати у вигляді орієнтованого графа. Граф складається з дуг, кожна з яких має свою вагу. Для пошуку оптимального шляху у графі розглянемо метод Мінті [8].

Під час побудови графа на основі маршрутів військ важливо правильно задати ваги його дуг. У нашій задачі можливі два випадки:

- 1) вага дуги характеризується одним параметром (довжиною шляху);
- 2) вага дуги характеризується двома параметрами (довжиною шляху та коефіцієнтом прохідності).

Щодо першого, то вага дуги буде дорівнювати значенню параметра; а стосовно другого – потрібно зважити дугу так, щоб вона залежала від двох параметрів. У нашій задачі параметрами будуть: довжина та коефіцієнт прохідності. Коефіцієнт прохідності – це величина, яка характеризує ступінь прохідності певної ділянки ландшафту на карті та знаходиться в межах від 0% до 100%. Перед зважуванням дуг варто змінити значення коефіцієнта прохідності за формулою:

$$k = 100 - k, \quad (1)$$

де  $k$  – коефіцієнт прохідності (%).

Так, найбільш прохідна ділянка (100%) тепер буде дорівнювати 0%. Така зміна значень потрібна для правильного пошуку оптимального шляху: всі методи пошуку оптимального шляху передбачають пошук мінімальної ваги дуги.

Потрібно звести задачу з двома параметрами в нашому випадку (в загальному випадку з  $n$  параметрами) до задачі з одним параметром. Для цього застосовують метод скалярного ранжування параметрів [1]. Скалярне ранжування – це підхід до вирішення багатокритеріальних задач прийняття рішень, коли множину критеріїв оптимальності зводять в один за допомогою функції скаляризації – цільової функції задачі прийняття рішень. Формула для зваженої суми параметрів загалом має такий вигляд:

$$F_1(f(x)) = \sum_{i=1}^n w_i f_i(x), \quad (2)$$

де  $n$  – кількість часткових критеріїв,  $w_i$  – коефіцієнт важливості (вага) часткового критерію,  $f_i(x)$  – функція корисності часткового критерію,  $w_i \in [0; 1]$ .

Потрібно розрахувати коефіцієнти важливості  $w_i, i = 1..n$ . Розташуємо параметри за ступенем важливості, задаючи для кожного параметра важливість у межах від 0 до 100. Найважливіший параметр рекомендовано задавати таким, що дорівнює 100. Задавши ступені важливості, знайдемо їхню суму:

$$S = \sum_{i=1}^n s_i, \quad (3)$$

де  $s_i$  – значення ступенів важливості.

Коефіцієнти важливості розраховують за такою формулою:

$$w_i = \frac{s_i}{S}, i = 1..n \quad (4)$$

Далі знаходимо значення зваженої дуги графа за такою формулою:

$$d = \sum_{i=1}^n w_i p_i, \quad (5)$$

де  $w_i$  – коефіцієнт важливості,  $p_i$  – значення параметрів відповідних критеріїв.

**Інтерфейс програми.** Ця програма моделює нанесення бойової обстановки на карту та обробку даних щодо ведення бойових дій:

- вибір карти та задання ландшафту;
- нанесення районів розташування військ;
- нанесення маршрутів висування військ;
- нанесення пунктів призначення маршрутів військ;

– вибір оптимальних маршрутів висування військ.

Спочатку ми завантажуємо карту та вибираємо масштаб. Карту завантажуємо та покриваємо сіткою залежно від вибраного масштабу (рис. 1). Потім наносимо ландшафт на карту (рис. 2).



Рис. 1. Карта, покрита сіткою



Рис. 2. Ландшафт, нанесений на карту

Тепер можемо нанести райони розташування військ, пункти призначення та маршрути військ. Задамо дві армії із двома маршрутами для кожної до пункту призначення, що для двох армій співпадає (рис. 3).

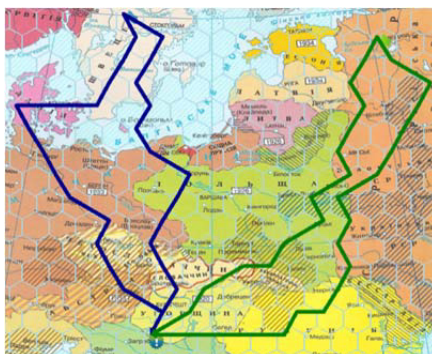


Рис. 3. Готова карта

Знайдемо оптимальний маршрут для першої армії:

- а) ступені важливості: довжина – 100%, прохідність – 50% (рис. 2.4, а);
- б) оптимальний маршрут за часом (рис. 2.4, б).

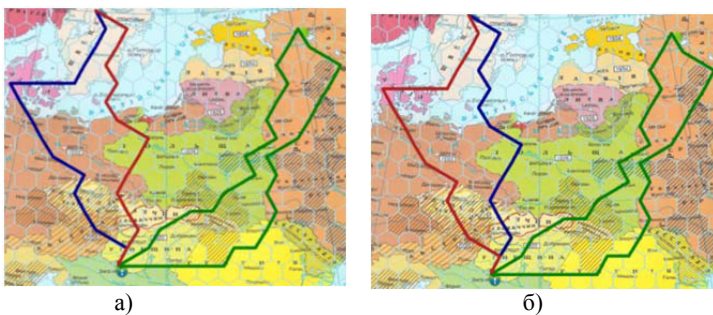
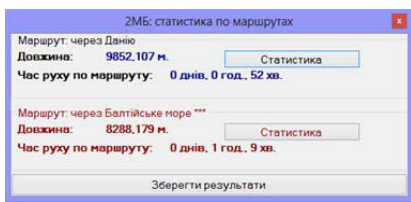
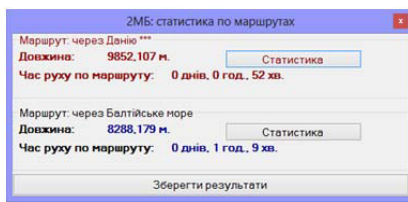


Рис. 2.4. Оптимальний шлях для першої армії (а, б)

Оптимальний маршрут відображаємо темно-червоним кольором. Після віднайдення оптимального шляху за тим чи іншим критерієм, відобразиться діалогове вікно зі статистикою (рис. 2.5).



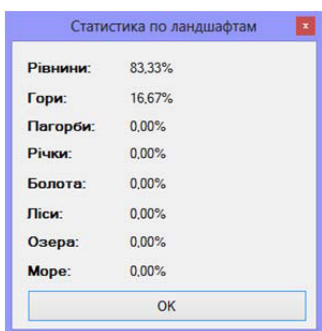
а)



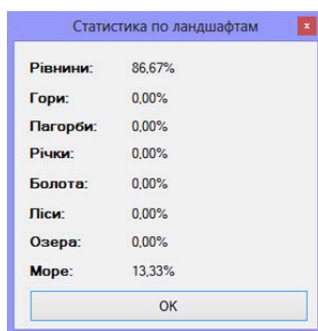
б)

Рис. 2.5. Статистика по маршрутах для випадків (а, б)

Також можемо переглянути статистику по ландшафту для кожного маршруту (рис. 2.6).



а)



б)

Рис. 2.6. Статистика по ландшафтах для маршруту «через Данію» (а) та маршруту «через Балтійське море» (б)



**Висновки.** В ході виконання роботи було розроблено програмний комплекс, який має такі функціональні властивості:

- завантаження карти та нанесення на неї ландшафту;
- розташування військових підрозділів та пунктів призначення;
- задання маршрутів руху військових підрозділів;
- збереження карти з усіма об'єктами у файл та завантаження карти з нього;
- збереження створеної схеми як зображення;
- пошук оптимального маршруту руху військ;
- статистика по маршрутах;
- порівняльний аналіз тактичних дій двох армій.

Практичне застосування запропонованої програми полягає у можливості використання останньої під час вивчення історії й у середніх, й у вищих навчальних закладах, у ході вивчення тактики і стратегії у спеціалізованих військових училищах та у процесі підготовки до військових навчань, для їхнього віртуального моделювання з метою покращення планування навчальних битв.

Подальше вдосконалення проекту передбачає розроблення задання початкових даних групою експертів із відповідною обробкою експертних даних та розроблення математичної моделі безпосередньо бойових дій.

1. Тараканов К. В. Математика и вооруженная борьба / К. В. Тараканов. – М. : Воениздат, 1974. – 240 с.
2. Ануреев И. Применение математических методов в военном деле / И. Ануреев, А. Татарченко. – М. : Воениздат, 1967. – 242 с.
3. Попов М. В. Сущность законов вооруженной борьбы / М. В. Попов – М. : Воениздат, 1964. – 133 с.
4. Трифоненков П. И. Об основных законах хода и исхода современной войны / П. И. Трифоненков. – М. : Воениздат, 1962. – 116 с.
5. Баграмян И. Х. История войн и военного искусства / И. Х. Баграмян, С. С. Лотоцкий. – М. : Воениздат, 1970. – 559 с.
6. Иванов Д. А. Основы управления войсками / Д. А. Иванов, В. П. Савельев, П. В. Шеманский. – М. : Воениздат, 1971. – 383 с.
7. Гольштейн Е. Г. Новые направления в линейном программировании / Е. Г. Гольштейн, Д. Б. Юдин. – М. : Советское радио, 1966. – 524 с.
8. Романовский И. В. Алгоритмы решения экстремальных задач / И. В. Романовский. – М. : Наука, 1977. – 352 с.

Степанченко О. М., к.т.н., ст. викл., Карачанський А. О., студент 5 курсу  
ННІ АКOT (Національний університет водного господарства та  
природокористування, м. Рівне)

## **РОЗРОБКА ГРАФІЧНОГО РЕДАКТОРА ДЛЯ ВИРІШЕННЯ ТАКТИЧНИХ ЗАВДАНЬ**

Статтю присвячено питанням тактичних завдань, їхнього вирішення, основних положень та перспектив розвитку. Показано шляхи вдосконалення застарілих методів для можливості комп'ютерної обробки інформації. Доведено логічність застосування у військовій діяльності сучасних технологій з огляду на її еволюцію та розвиток. Обґрунтовано необхідність створення нових наборів інструментів для вирішення вищеназваних питань. Вказано, що основна перевага таких технологій – це можливість швидкої зміни вхідних даних для внесення динамічних змін у модель.

**Ключові слова:** прийняття рішень, зважений граф, бойовий потенціал, граф, ймовірність, моделювання дуелі.

**This article is about tactical tasks and their problems, main statements and most important perspectives. It shows, how we can infiltrate new methods into tactics, add computer resources for better result. Warfare changes and gets new methods, develops a lot, so it's enough logical to use modern technologies for planning. It's so important to have new tools, they must be used to provide the best result. Main advantage of these technologies - dynamic model changes with fast reaction.**

**Key words:** decision making, graph, battle potential, probability, duel model.

**Вступ.** В сучасних умовах динаміка розвитку стилю ведення бойових дій є особливо інтенсивною. Ще століття назад інформацію вважали лише однією із вирішальних складових системи ведення бойових дій, а на сучасному етапі їй надано статус пріоритетного фактора. З огляду на це окреслюється важливість наявності якомога більш повного набору даних про ворога та територію, на якій він розміщується. Використання комп'ютерної техніки на сьогодні апробовано на практиці ведення бойових дій, саме тому справедливо припускати ефективність застосування її під час прогнозування бою та в ході навчання і підготовки особового складу.

Система ведення бойових дій у площині сьогодення зазнала значних трансформацій. Наприклад, літакам навіть не потрібно бачити один одного для завдання удару. Зважаючи на те, що цифрові технології визначають результат бою, як і загальне моделювання тактики та стратегії його

проведення, врахування всіх факторів, усіх технологій та можливих результатів бою визнано завданням комп'ютерної техніки.

Розрахунок вважають неефективним у разі використання вручну через те, що займе багато часу, а його візуальне представлення досить громіздке та негнучке. Саме використання комп'ютерних технологій дає змогу не лише краще уявити суть того, що буде відбуватися, а й уможливити прогнозування кількох варіантів розвитку подій без витрачання на це багато часу, а лише зі зміною декількох коефіцієнтів, або параметрів. Проте увиразнюється необхідність створення відповідного програмного забезпечення, яке буде ефективно виконувати своє призначення, а саме – спрощення роботи зі створення тактичних схем із подальшим прогнозуванням їхнього розвитку. Такі програми поширені в ігровій промисловості, втім, з кардинально відмінним призначенням. Так, якщо і «найоб'ємніші» за кількістю «юнітів» стратегії, які можна вважати найбільш правдоподібними, може бути використано за основу розроблення вищевказаного програмного забезпечення, то залучення більшості інших є нерациональним через їхній «локальний характер», тобто обмежену кількість бойових одиниць, пов'язану з обмеженістю кількості ресурсів. Проте займатися нівелюванням вказаної обмеженості не вигідно через негативний вплив таких змін на графічне представлення. У свою чергу військові структури були зацікавлені лише у розробленні безпечних методів тренувань (авіасимулятори і тощо), а тактичні завдання вирішували на основі застарілих методів. З огляду на це в ігровій індустрії не було створено подібних продуктів.

Створене програмне забезпечення може бути і демонстраційним, і безпосередньо виконувати функцію прогнозування, тобто забезпечення можливості формування уявлення про кінцевий результат виконання тактичної операції. Крім того, оскільки прогнозування неможливе без урахування дій суперників, йдеться про питання штучного інтелекту та його застосування. Втім, потрібно визнати, що навіть використання штучного інтелекту не забезпечить адекватного вирішення проблеми, тому очевидно, що повністю не враховувати людський фактор у розрахунках не буде можливим. Загалом з огляду на глобальність сфери використання аналізованих у роботі програм питання їхнього створення залишається відкритим та актуальним.

**Постановка задачі.** Необхідним є створення програмного забезпечення, у якому повинно бути передбачено введення наявних видів військ із послідовним установленням їхніх характеристик, формування сценарію поведінки та організації переміщення військ.

Програмний продукт повинен мати такі можливості:

- створення об'єктів, які є умовними позначеннями військ;
- забезпечення можливості редагування внутрішньої бази даних таких об'єктів, установлення їхніх характеристик;
- використання набору карт “Google Earth”;

- забезпечення можливості задавати рух військ на ціль для атаки;
- створення нерухомих об'єктів, оточення;
- прогнозування результатів бою;
- створення звіту;
- можливість пошкоджувати або послаблювати ворожі війська та втрачувати зміни їхньої ефективності.

**Математичний розв'язок задачі.** Розташування військ задано масивом зв'язаних графів, кожен з яких визначає їхнє положення на певний момент часу.

Граф у свою чергу задано за допомогою 2 таблиць:

1) таблиця вершин, де кожній із вершин поставлено у відповідність координати  $x, y$ ;

2) таблиця інцидентності, де показано зв'язки між вершинами шляхом присвоєння значень на перетині  $i$  стовпця та  $j$  рядка.

Крім того, кожній бойовій одиниці присвоєно значення бойового потенціалу – умовне визначення її сили.

Розрахунок виконуємо на основі моделювання дуелі, де кожному з учасників присвоєно певне значення – ймовірність ураження певного типу ворога, яке, відповідно, може бути визначено і як просто ймовірність влучення, і як середня кількість влучень, потрібних для ураження.

Розрахуємо з огляду на потенціали власного ( $n_1$ ) та ворожого ( $n_2$ ) значення  $p$  та  $q$ :

$$p = \left(1 - \frac{n_1}{n_1 + n_2}\right), \quad (1)$$

$$q = (1 - p). \quad (2)$$

Для кожного з моментів часу існує ймовірність ураження. Розрахувавши її на кожному кроці, можна встановити, який з об'єктів буде знищено першим. Розрахунок виконуємо за формулами:

$$\frac{dP_A(t)}{dt} = \lambda q - \lambda(p + q)P_A(t), \quad (3)$$

$$\frac{dP_B(t)}{dt} = \lambda p - \lambda(p + q)P_B(t), \quad (4)$$

де  $\lambda$  – кількість умовних атак;  $p, q$  – вище обчислені ймовірності.

Кількість атак, які необхідні для ураження ворожої цілі, застосовані для покрокового моменту бою з метою обчислення ймовірності ураження цілі, або ж невдачі, яка супроводжувала ці спроби, обчислюємо так:

$$N_A = \frac{(\lg(q) - \lg(p + q))}{\lg(1 - p)}, \quad (5)$$

$$N_B = \frac{(\lg(p) - \lg(p + q))}{\lg(1 - q)}. \quad (6)$$

**Інтерфейс програми.** Ця програма моделює виконання бойової задачі, працює з даними ведення бойових дій:

- створює карти за допомогою Google Earth;
- додає різних типів військ, які будуть брати участь у бойових діях;
- встановлює місце розташування загонів у різні моменти часу;
- прогнозує результати виконання бойової задачі;
- демонструє виконання бойового плану.

Початкове завантаження програми – це відкриття вікна з віртуальним представленням землі, яке буде використано для створення карти (рис. 1). У програмі також передбачено можливість завантаження власної карти, достатньої для виконання програми. Після виконання пункту меню вигляд карти буде збережено окремим файлом і завантажено на форму. Після чого можна задати на ній розташування військових загонів (рис. 2).

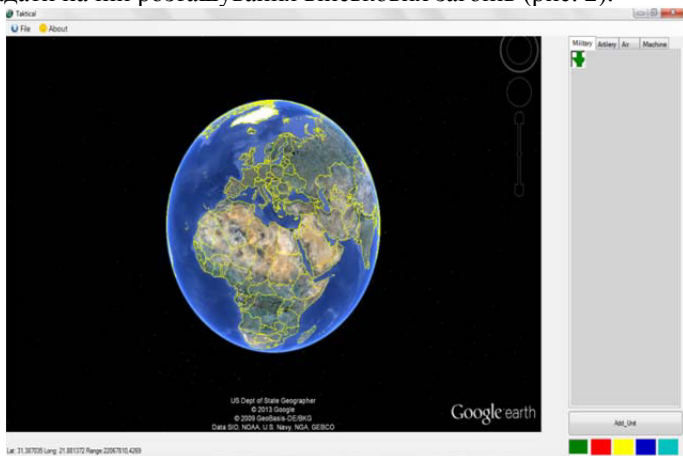


Рис. 1. Початкове вікно

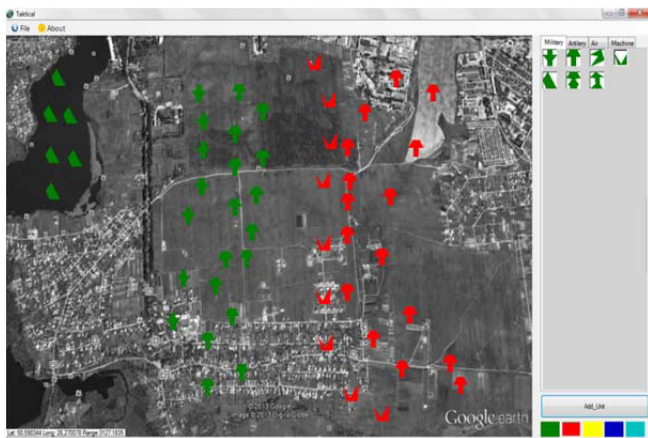


Рис. 2. Збережена карта із розміщеними військами

Тепер можна додати власні типи військ за допомогою спеціальної форми, яка буде містити інформацію про умовний вигляд, бойовий потенціал та бойову групу, що слугує для правильного застосування цього загону в бою (рис. 3).

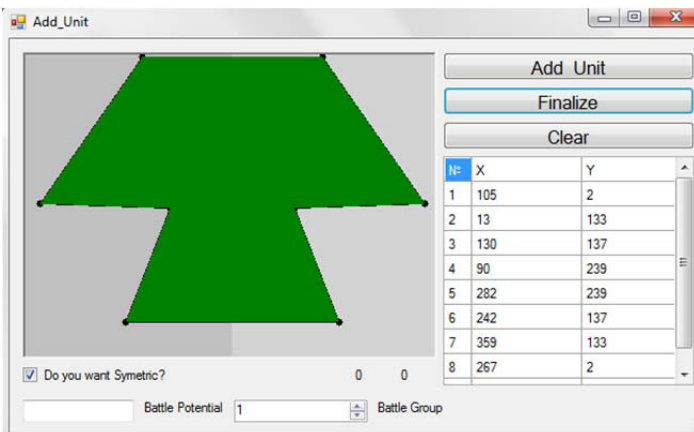


Рис. 3. Меню для додання типів військ

Задаємо маршрути для кожного із загонів та його кінцеву ціль (рис. 4).



Рис. 4. Зображення руху для вибраного загону

Відповідно, під час використання спеціально створеного меню можна змінити значення часу та побачити поточну ситуацію на полі бою (рис. 5).

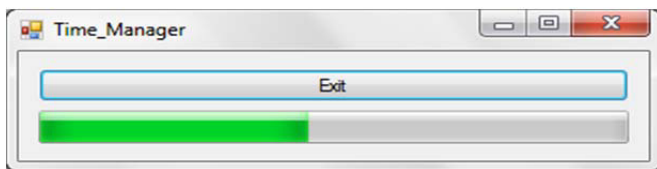


Рис. 5. Часова шкала для зміни розташування військ

Після демонстрації бойової операції буде представлено її результат на тій самій формі із можливістю його збереження та виведення.

**Висновки.** В ході виконання запропонованої роботи було створено графічний редактор, який має такі функції:

- створення об'єктів, які є умовними позначеннями військ;
- забезпечення можливості редагування внутрішньої бази даних таких об'єктів, встановлення їхніх характеристик;
- використання набору карт “Google Earth”;
- забезпечення можливості задавати рух військ на ціль для атаки;
- створення нерухомих об'єктів, оточення;
- прогнозування результатів бою;
- створення звіту;
- можливість пошкоджувати або послаблювати ворожі війська та враховувати зміни їхньої ефективності.

З огляду на неефективність математичних моделей, якими послуговувалися 50 років назад для прогнозування результатів ведення бою, актуалізується необхідність створення продукту, що уможливить осучаснення методів демонстрації тактичних завдань, застосовуваних і під час навчання, і в ході вирішення цілком реальної задачі, і в процесі демонстрації.

Подальше вдосконалення передбачає створення штучного інтелекту для симуляції ворога, можливість знищення об'єктів, які були нерухомими під час атаки на інший об'єкт, можливість читання карт програмою та її роботи не із зображенням карти, а безпосередньо на Google Maps, використовуючи мову розмітки.

1. Тараканов К. В. Математика и вооруженная борьба / К. В. Тараканов. – М. : Воениздат, 1974. – 240 с.
2. Ткаченко П. Н. Математические модели боевых действий / П. Н. Ткаченко, Л. Н. Куцев, Г. А. Мещеряков, А. М. Чавкин, А. Д. Чебыкин. – М. : Изд-во «Советское радио», 1969. – 240 с.
3. Попов М. В. Сущность законов вооруженной борьбы / М. В. Попов. – М. : Воениздат, 1964. – 133 с.
4. Алексеев О. Г. Марковские модели боя / О. Г. Алексеев, В. Г. Анисимов, Е. Г. Анисимов. – М.: Министерство обороны СССР, 1985. – 85 с.
5. Скурихин В. И. Математическое моделирование / В. И. Скурихин, В. Б. Шифрин, В. В. Дубровский. – М. : Техника, 1983. – 265 с.
6. Иванов Д. А. Основы управления войсками / Д. А. Иванов, В. П. Савельев, П. В. Шеманский. – М. : Воениздат, 1971. – 383 с.

Турбал Ю. В., к.ф.-м.н., доцент, Радовенюк О. В., аспірант,  
Турбал М. Ю., студентка 3 курсу ННІ АКОТ (Національний університет  
водного господарства та природокористування, м. Рівне)

## **ГРАНИЧНІ ВЛАСТИВОСТІ ОРБІТ ДИСКРЕТНИХ ДИНАМІЧНИХ СИСТЕМ, ЩО ВИЗНАЧАЮТЬСЯ КЛАСОМ КВАДРАТИЧНИХ ОПЕРАТОРІВ**

**У роботі показано, що орбіти дискретних динамічних систем, які визначаються квадратичним оператором певного виду, характеризуються збіжністю параметра кривизни до деякого граничного значення.**

**Ключові слова:** динамічна система, орбіта, оператор, рукав.

**It is shown that the orbits of discrete dynamical systems defined by a quadratic operator, are characterized by the curvature parameter convergence to a certain limit.**

**Key words:** dynamic system, orbit operator sleeve.

**Вступ.** Одним із відомих методів дослідження неперервних динамічних систем є метод перетинів Пуанкаре, сутність якого полягає у побудові множин точок перетину фазової траєкторії системи з деякою поверхнею (у найпростішому варіанті – площиною). Відповідний підхід має важливе практичне значення, оскільки у багатьох задачах, пов'язаних із дослідженням фазових траєкторій динамічних систем, відомими є лише певні дискретні “вияви” траєкторій, спостереження їх на певних умовних поверхнях, коли фіксація всієї траєкторії неможлива.

Так, наприклад, можна досліджувати траєкторії солітонів, що рухаються в глибині Землі лише крізь призму їхньої взаємодії з областями накопичення сейсмічної енергії, яка призводить до сейсмічних поштовхів, які фіксують сейсмічні станції. В такому разі області, де відбувається накопичення сейсмічної енергії (так звані сейсмоактивні зони), логічно вибирати як поверхні Пуанкаре.

Аналогічно поводяться космічні тіла, проходячи через рукави спіральних галактик: вони зменшують швидкість, інтенсивніше взаємодіють із складовими рукава, що дає змогу з більшою ймовірністю їх зафіксувати. В цьому разі спіральні рукави логічно вибрати як поверхні Пуанкаре. Солітони в газових дисках галактик інтенсивно взаємодіють з областями підвищеної щільності газу, які у галактиках часто є послідовністю кіл певного радіуса. Ці кола – криві Пуанкаре.

Зауважимо, що траєкторії природних об'єктів часто є просторовими спіралями, а поверхні Пуанкаре – сферами. За таких умов відображення



Пуанкаре добре апроксимуються квадратичними операторами спеціального виду [3, 4].

У запропонованій роботі будемо досліджувати властивість постійності кутів для орбіт дискретних динамічних систем, які породжує клас квадратичних операторів. Така властивість важлива для доведення теорем про апроксимацію неперервних траєкторій (зокрема логарифмічних спіралей) за допомогою орбіт дискретних динамічних систем, що визначаються певними класами операторів, а це, у свою чергу, є основою для моделювання відображень Пуанкаре.

**Необхідні визначення та постановка задачі.** Нехай  $A$  – оператор, який діє у просторі  $R^n$  і задає деяке відображення Пуанкаре.

Означення 1. Позитивною напівтраєкторією точки  $x_0$  (орбітою) будемо називати множину точок виду:

$$O_A^+(x_0) = \{x_n : x_n = A(x_{n-1}), n = 1, 2, \dots\}.$$

Означення 2. Від'ємною напівтраєкторією точки  $x_0$  будемо називати множину точок виду:  $O_A^-(x_0) = \{x \in C : A^k(x) = x_0, k = 0, 1, 2, \dots\}$ , де  $A^k(x) = A(A(\dots A(x)\dots))$ .

Означення 3. Рукавом послідовності  $O_A^+(x_0)$  будемо називати таку її підпослідовність  $x_j, x_{j+m}, x_{j+2m}, x_{j+3m}, \dots$ , що

$$\rho(x_{j+(k-1)m}, x_{j+km}) = \min_{x_i \in O_A^+(x_0), i > j+(k-1)m} \rho(x_{j+(k-1)m}, x_i).$$

Число  $m$  – це кількість рукавів.

Означення 4. Басейном притягання  $B(x_0)$  точки  $x_0$  будемо називати множину всіх тих точок  $x$ , чиї позитивні напівтраєкторії  $O_A^+(x)$  збігаються до  $x_0$ .

Розглянемо оператор, що діє у просторі  $R^2$  і визначається співвідношенням виду:

$$A \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x^2 - y^2 + c_x \\ 2xy + c_y \end{pmatrix}, \quad (1)$$

де  $c_x, c_y$  – деякі константи. Нехай  $\tilde{x}_c = (x_c, y_c)$  – нерухома точка оператора  $A$ ,  $O_A^+(x)$  – деяка позитивна напівтраєкторія,  $x \in B(\tilde{x}_c)$ . Побудуємо такий оператор  $\tilde{A}$ , для якого виконується співвідношення:

$$O_{\tilde{A}}^+(x - \tilde{x}_c) = O_A^+(x) - \tilde{x}_c. \quad (2)$$

У такому разі будь-яка позитивна напівтраєкторія оператора  $A$ , зсунута на вектор  $\tilde{x}_c$ , є позитивною напівтраєкторією оператора  $\tilde{A}$ . Очевидно, що оператор, який задовольняє властивість (2), має вигляд:

$$\tilde{A} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} (x + x_c)^2 - (y + y_c)^2 + c_x - x_c \\ 2(x + x_c)(y + y_c) + c_y - y_c \end{pmatrix}. \quad (3)$$

Нехай  $\bar{x} = x + iy$  – точка в комплексній площині. Запишемо (3) у вигляді деякої раціональної функції. Внаслідок нескладних міркувань визначено, що цю функцію можна записати у вигляді:  $\tilde{A}(\bar{x}) = \bar{x}^2 + 2\tilde{x}\tilde{x}_c$ , де  $\tilde{x}_c$  – притягуюча нерухома точка оператора  $A$ .

Будемо розглядати послідовність  $O_{\tilde{A}}^+(c_x, c_y)$ . Нашим завданням є знаходження границі косинуса кута, що характеризує кривизну рукава (див. рис.), а саме величину виду:  $\cos\psi_n = \frac{(-\tilde{A}^{1+nm}c, \tilde{A}^{1+(n+1)m}c - \tilde{A}^{1+nm}c)}{|\tilde{A}^{1+nm}c| |\tilde{A}^{1+(n+1)m}c - \tilde{A}^{1+nm}c|}$ ,

де  $m$  – кількість рукавів відповідної структури.

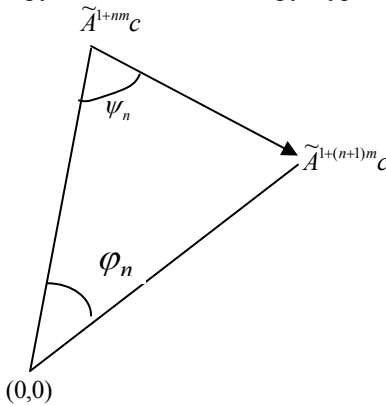


Рис.

**Основні результати.** Використовуючи структуру оператора  $\tilde{A}$ , можемо побудувати таку послідовність:

$$\tilde{A}x = x^2 + 2x\tilde{x}_c \quad (4)$$

$$\tilde{A}^2x = (x^2 + 2x\tilde{x}_c)^2 + 2(x^2 + 2x\tilde{x}_c)\tilde{x}_c =$$

$$= x^4 + 4x^3\tilde{x}_c^2 + 4x^2\tilde{x}_c^2 + 2x^2\tilde{x}_c + 4x\tilde{x}_c^2$$

...

$$\tilde{A}^n x = x^{2^n} + a_{2^{n-1}}^n x^{2^{n-1}} + \dots + a_1^n x$$

$$\tilde{A}^{n+1} x = (\tilde{A}^n x)^2 + 2\tilde{A}^n x\tilde{x}_c = x^{2^{n+1}} + a_{2^{n+1}-1}^{n+1} x^{2^{n+1}-1} + \dots + a_1^{n+1} x$$

Тоді легко отримати співвідношення між коефіцієнтами представлень (4):

$$a_{2^{n+1}-1}^{n+1} = 2a_{2^n-1}^n$$

$$a_{2^{n+1}-2}^{n+1} = 2a_{2^n-2}^n + (a_{2^n-1}^n)^2$$

$$a_{2^{n+1}-3}^{n+1} = 2a_{2^n-3}^n + 2a_{2^n-1}^n a_{2^n-2}^n \quad (5)$$

.....

$$a_{2^{n+1}-k}^{n+1} = \sum_{i,j=0, i+j=k}^k a_{2^n-i}^n a_{2^n-j}^n, \quad k < 2^n$$

$$a_{2^{n+1}-k}^{n+1} = \sum_{i,j=1, i+j=2^{n+1}-k}^{2^{n+1}-k-1} a_i^n a_j^n + 2\tilde{x}_c a_{2^{n+1}-k}^n, \quad 2^n \leq k \leq 2^{n+1} - 2$$

$$a_{2^{n+1}-k}^{n+1} = 2\tilde{x}_c a_{2^{n+1}-k}^n, \quad k = 2^{n+1} - 1$$

Останнє зі співвідношень (5) запишемо у вигляді:

$$a_1^{n+1} = 2\tilde{x}_c a_1^n = 2^n (\tilde{x}_c)^n. \quad (6)$$

Нехай  $\tilde{A}^{1+nm} c = x + iy = \tilde{x}_{1+nm}$ . Тоді

$$\tilde{A}^{1+(n+1)m} c = \tilde{A}^m (\tilde{A}^{1+nm} c) = \tilde{x}^{2^m} + a_{2^m-1}^m \tilde{x}^{2^m-1} + \dots + a_1^m \tilde{x}$$

Позначимо

$$\cos \psi_n = \frac{(-\tilde{A}^{1+nm} c, \tilde{A}^{1+(n+1)m} c - \tilde{A}^{1+nm} c)}{\left| \tilde{A}^{1+nm} c \right| \left| \tilde{A}^{1+(n+1)m} c - \tilde{A}^{1+nm} c \right|} = \beta(\operatorname{Re}(\tilde{x}_{1+nm}), \operatorname{Im}(\tilde{x}_{1+nm})).$$

Оскільки розглядаємо початкові точки з басейну притягання  $(0,0)$ , то

$\lim_{n \rightarrow \infty} \tilde{A}^{1+nm} c = (0,0)$ . Якщо існує границя  $\lim_{\substack{x \rightarrow 0 \\ y \rightarrow 0}} \beta(x, y)$ , то, очевидно, існує

границя  $\lim_{n \rightarrow \infty} \beta(\operatorname{Re}(\tilde{x}_{1+nm}), \operatorname{Im}(\tilde{x}_{1+nm}))$ , і при цьому виконується

співвідношення:  $\lim_{\substack{x \rightarrow \infty \\ y \rightarrow \infty}} \beta(x, y) = \lim_{n \rightarrow \infty} \beta(\operatorname{Re}(\tilde{x}_{1+nm}), \operatorname{Im}(\tilde{x}_{1+nm}))$ .

Для доведення сумісної збіжності будемо розглядати відповідні послідовні границі:  $\lim_{x \rightarrow 0} \lim_{y \rightarrow 0} \beta(x, y)$ ,  $\lim_{y \rightarrow 0} \lim_{x \rightarrow 0} \beta(x, y)$ .

$$\text{Нехай } f(\tilde{x}) = \tilde{x}^{2^m} + a_{2^m-1}^m \tilde{x}^{2^m-1} + \dots + (a_1^m - 1)\tilde{x}$$

Можемо сформулювати такі властивості.

$$\text{Лема 1. } \lim_{y \rightarrow 0} f(\tilde{x}) = x^{2^m} + a_{2^m-1}^m x^{2^m-1} + \dots + (a_1^m - 1)x$$

Ця лема безпосередньо ґрунтується на операції множення комплексних чисел.

$$\text{Лема 2. } \lim_{y \rightarrow 0} \operatorname{Re}(f(\tilde{x})) = \operatorname{Re}(f(x)); \lim_{y \rightarrow 0} \operatorname{Im}(f(\tilde{x})) = \operatorname{Im}(f(x)).$$

Ця властивість безпосередньо ґрунтується на Лемі 1. Очевидними є й такі леми:

$$\text{Лема 3. } \lim_{x \rightarrow 0} f(\tilde{x}) = \operatorname{Im}(f(iy)).$$

$$\text{Лема 4. } \lim_{x \rightarrow 0} \operatorname{Re}(f(\tilde{x})) = \operatorname{Re}(f(iy)), \lim_{x \rightarrow 0} \operatorname{Im}(f(\tilde{x})) = \operatorname{Im}(f(iy)).$$

Тоді

$$\begin{aligned} \lim_{x \rightarrow 0} \lim_{y \rightarrow 0} \frac{- (\tilde{x}, f(\tilde{x}))}{\|\tilde{x}\| \sqrt{(\operatorname{Re} f(\tilde{x}))^2 + (\operatorname{Im} f(\tilde{x}))^2}} &= \lim_{x \rightarrow 0} \frac{-x \operatorname{Im}(f(x))}{\sqrt{x^2} \sqrt{\left(\frac{\operatorname{Re} f(x)}{x}\right)^2 + \left(\frac{\operatorname{Im} f(x)}{x}\right)^2}} = \\ &= \lim_{x \rightarrow 0} \frac{-\operatorname{Re}(f(x))}{\sqrt{\left(\frac{\operatorname{Re} f(x)}{x}\right)^2 + \left(\frac{\operatorname{Im} f(x)}{x}\right)^2}} = \frac{-\operatorname{Re}(a_1^m - 1)}{\sqrt{(\operatorname{Re}(a_1^m - 1))^2 + (\operatorname{Im}(a_1^m - 1))^2}} = \\ &= \frac{-\operatorname{Re}(2^m \tilde{x}_c^m - 1)}{\sqrt{(\operatorname{Re}(2^{m-1} \tilde{x}_c^{m-1} - 1))^2 + (\operatorname{Im}(2^{m-1} \tilde{x}_c^{m-1} - 1))^2}}. \end{aligned}$$

Аналогічно знаходимо іншу послідовну границю:

$$\begin{aligned} \lim_{y \rightarrow 0} \lim_{x \rightarrow 0} \frac{- (\tilde{x}, f(\tilde{x}))}{\sqrt{x^2 + y^2} \sqrt{(\operatorname{Re} f(\tilde{x}))^2 + (\operatorname{Im} f(\tilde{x}))^2}} &= \lim_{y \rightarrow 0} \frac{-y \operatorname{Im}(f(iy))}{\sqrt{y^2} \sqrt{(\operatorname{Re} f(iy))^2 + (\operatorname{Im} f(iy))^2}} = \\ &= \lim_{y \rightarrow 0} \frac{-\operatorname{Im}(f(iy)/y)}{\sqrt{(\operatorname{Re} f(iy)/y)^2 + (\operatorname{Im} f(iy)/y)^2}} = \frac{-\operatorname{Im}(i(a_1^m - 1))}{\sqrt{(\operatorname{Re}(i(a_1^m - 1)))^2 + (\operatorname{Im}(i(a_1^m - 1)))^2}} = \\ &= \frac{-\operatorname{Re}(a_1^m - 1)}{\sqrt{(\operatorname{Im}(a_1^m - 1))^2 + (\operatorname{Re}(a_1^m - 1))^2}} = \frac{-\operatorname{Re} \mathcal{Q}^{m-1} \tilde{x}_c^{m-1} - 1}{\sqrt{(\operatorname{Re} \mathcal{Q}^{m-1} \tilde{x}_c^{m-1} - 1)^2 + (\operatorname{Im} \mathcal{Q}^{m-1} \tilde{x}_c^{m-1} - 1)^2}} \end{aligned}$$

Отже, можемо сформулювати таку теорему:

**Теорема 1.** Нехай  $\tilde{x}_c$  – нерухома точка оператора (1). Тоді для довільної точки  $x$ , що належить басейну притягання точки  $\tilde{x}_c$ , існують границі виду:

$$\begin{aligned} \lim_{n \rightarrow \infty} \cos \psi_n &= \lim_{n \rightarrow \infty} \frac{(-\tilde{A}^{1+nm} x, \tilde{A}^{1+(n+1)m} x - \tilde{A}^{1+nm} x)}{\left| \tilde{A}^{1+nm} x \right| \left| \tilde{A}^{1+(n+1)m} x - \tilde{A}^{1+nm} x \right|} = \\ &= \frac{-\operatorname{Re}(2^m \tilde{x}_c^m - 1)}{\sqrt{(\operatorname{Re}(2^m \tilde{x}_c^m - 1))^2 + (\operatorname{Im}(2^m \tilde{x}_c^m - 1))^2}} \\ \lim_{n \rightarrow \infty} \cos \varphi_n &= \lim_{n \rightarrow \infty} \frac{(\tilde{A}^{1+nm} x, \tilde{A}^{1+(n+1)m} x)}{\left| \tilde{A}^{1+nm} x \right| \left| \tilde{A}^{1+(n+1)m} x \right|} = \\ &= \frac{\operatorname{Re}(2^{m-1} \tilde{x}_c^{m-1})}{\sqrt{(\operatorname{Re}(2^{m-1} \tilde{x}_c^{m-1}))^2 + (\operatorname{Im}(2^{m-1} \tilde{x}_c^{m-1}))^2}}. \end{aligned}$$

Ця теорема безпосередньо детермінована наведеними вище лемами, а також збіжністю величини  $\tilde{A}^{1+nm} x$  до  $\tilde{A}^{1+nm}(c_x, c_y)$ , розглянутою у роботі [2].

**Висновки.** У роботі досліджено властивість дискретної динамічної системи, що визначається квадратичним оператором (1), яка полягає у збіжності кутів, які утворюють окремі точки системи, та нерухомої точки оператора (1) до деяких граничних значень. Відповідні граничні значення знайдено, результат сформульовано у вигляді теореми. Доведена теорема фактично підтверджує отриманий раніше результат [2] про можливість апроксимації логарифмічних спіралей орбітами дискретних динамічних систем, що визначаються оператором виду (1). Отриманий у роботі результат уможливило моделювання відображень Пуанкаре за допомогою квадратичних операторів у тому разі, коли фазова траєкторія системи має спіральну структуру.

1. Солодухина А. М. Сечения Пуанкаре фазового портрета нелинейного RLS контура, содержащего сигнатурный элемент / А. М. Солодухина, А. С. Сидоркин, А. А. Шевченко // Физика твердого тела. – т. 35. – №7. – 1993. 2. Постнов В. Э. Бифуркации регулярных аттракторов / В. Э. Постнов. – Саратов : Изд. ГосУНЦ Колледж, 1996. 3. Турбал Ю. В. Деякі властивості позитивних напівтраєкторій Жюліа / Ю. В. Турбал // Волинський математичний вісник. – 2006. – №4. – С. 87–96. 4. Турбал Ю. В. Про самоорганізацію динамічних систем у векторних полях / Ю. В. Турбал // Вісник Київського університету. Сер.: Фізико-математичні науки. – 2007. – № 1. – С. 171–179.

Гладун Л. В., к.ф.-м.н., доцент (Національний університет водного господарства та природокористування, м. Рівне)

## ОПТИМІЗАЦІЙНИЙ АНАЛІЗ ПРЯМИХ МЕТОДІВ РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ

Проаналізовано результати експериментальних досліджень поведінки розв'язків систем лінійних алгебраїчних рівнянь, отриманих із використанням пакетів Matlab і Maple.

**Ключові слова:** системи лінійних алгебраїчних рівнянь, розв'язки системи, прямі методи знаходження розв'язку, кількість знаків під обчислень.

The results of experimental researches of conduct of decisions of the systems of the linear algebraic equations got with the use of packages of Matlab and Maple are analyzed.

**Key words:** the systems of linear equalizations of algebra, upshot of the system, direct methods of being of decision, amount of signs, are at calculations.

Ще здавна математиків усього світу цікавила проблема знаходження розв'язків систем лінійних алгебраїчних рівнянь

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2, \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases} \quad (1)$$

На сьогодні розроблено і перевірено на практиці досить багато методів їхнього розв'язання [1–5]. Ці методи умовно можна розділити на дві групи: прямі та ітераційні. До прямих методів знаходження розв'язків (1), зокрема, належать матричний, Гауса та метод визначників.

В сучасних умовах для знаходження розв'язків систем рівнянь використовують пакети Matlab, Maple та інші [6–9]. Отримані цими пакетами розв'язки не завжди співпадають.

Знаходження розв'язків систем лінійних алгебраїчних рівнянь є досить актуальною проблемою і в наш час, оскільки ними широко послуговуються під час розв'язання багатьох прикладних задач.

**Мета досліджень:** проаналізувати розв'язки певного типу систем лінійних алгебраїчних рівнянь, які отримано на основі прямих методів із

використанням пакетів Matlab і Maple; визначити, застосування якого з методів дає можливість отримати точніші результати.

У ході досліджень було проаналізовано розв'язки близько трьох тисяч систем лінійних алгебраїчних рівнянь, коефіцієнти яких задавали за певними правилами.

### Розглянемо систему лінійних алгебраїчних рівнянь

$$\begin{pmatrix} 1 & 2 & 1 & 1 & 1 & 2 & 1 & 2 & 3 & 4 \\ \sqrt{5} & \sqrt{3} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \sqrt{3} & \sqrt{5} & 5 & 0 & 3 & 3 & 5 & 2 & 7 & 5 \\ \sqrt{5} & \sqrt{3} & 6 & 0 & 3 & 0 & 5 & 0 & 5 & 0 \\ \sqrt{2} & \sqrt{7} & 0 & 5 & 3 & 2 & 5 & 4 & 7 & 5 \\ \sqrt{7} & \sqrt{2} & 1 & 1 & 1 & 1 & 1 & 5 & 4 & 3 \\ \sqrt{2} & \sqrt{7} & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 8 \\ \sqrt{5} & \sqrt{6} & 1 & 3 & 3 & 3 & 5 & 6 & 6 & 6 \\ \sqrt{6} & \sqrt{7} & 4 & 2 & 0 & 4 & 5 & 0 & 7 & 0 \\ \sqrt{7} & \sqrt{6} & 1 & 0 & 0 & 0 & 5 & 8 & 5 & 8 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{pmatrix} = \begin{pmatrix} 3 \\ \sqrt{5} + \sqrt{3} \\ \sqrt{3} + \sqrt{5} \\ \sqrt{5} + \sqrt{3} \\ \sqrt{2} + \sqrt{7} \\ \sqrt{7} + \sqrt{2} \\ \sqrt{2} + \sqrt{7} \\ \sqrt{5} + \sqrt{6} \\ \sqrt{6} + \sqrt{7} \\ \sqrt{7} + \sqrt{6} \end{pmatrix} \quad (2)$$

Оскільки визначник системи відмінний від нуля, то розв'язком системи є числа  $x_1 = x_2 = 1$ ,  $x_3 = x_4 = \dots = x_{10} = 0$ . Систему рівнянь було спеціально вибрано такого вигляду, щоб легко можна було аналізувати отримані розв'язки.

Разом із системою (2) було розглянуто системи лінійних алгебраїчних рівнянь подібного типу, в яких перший, другий і стовпець вільних членів задано квадратними коренями із чисел, а коефіцієнти при змінних  $x_3, x_4, \dots, x_{10}$  набували таких значень: числа порядку першого степеня, восьмого степеня, шістнадцятого степеня, різні числа від першого до восьмого степеня, числа порядку мінус четвертого степеня, мінус восьмого степеня, мінус дванадцятого степеня та мінус шістнадцятого степеня.

Дослідження проводили таким чином: із коефіцієнтів, які задані квадратними коренями, вибирали один або кілька й у програмі задавали функцією SQRT(). Решта коефіцієнтів задавали через наближення з певною кількістю знаків після коми.

У таблиці 1 наведено розв'язки системи (2), отримані з використанням пакета Matlab, в якій перший, другий і стовпець вільних членів задано

квадратними коренями, а коефіцієнти при невідомих  $x_3, x_4, \dots, x_{10}$  – числа першого степеня.

Із таблиці 1 зрозуміло, що застосування методу Гауса дає змогу отримати більш точний розв'язок, ніж оперування матричним методом та методом визначників. Крім того, цей результат співпадає з тим, коли всі коефіцієнти, задані через квадратний корінь, записують за допомогою функції SQRT(). Такою властивістю відзначаються всі три методи.

Таблиця 1

Розв'язки системи рівнянь

Метод визначників	Метод Гауса	Матричний метод
$X =$	$X =$	$X =$
1,00000000000E+00	1,00000000000E+00	1,00000000000E+00
1,00000000000E+00	1,00000000000E+00	1,00000000000E+00
-3,44518072135E-16	-3,18207052651E-17	-3,33066907388E-16
-2,18544621906E-16	-3,76242551745E-17	0,00000000000E+00
1,66553948710E-16	-1,69256568144E-17	-1,11022302463E-16
2,59451901979E-17	-2,33594920024E-17	2,22044604925E-16
1,82711438350E-16	2,41770374608E-17	2,22044604925E-16
-2,36811965712E-16	-4,52209957385E-18	-9,43689570931E-16
-1,02111550413E-16	8,94270339421E-18	-7,49400541622E-16
2,94766271809E-17	2,35376783599E-18	-6,93889390391E-17

Стосовно різноманітних комбінацій коефіцієнтів, узятих із наближеннями і «точно», тобто через функцію SQRT(), спостережено певну закономірність: залежно від кількості знаків після коми в коефіцієнтах, які взято з наближеннями, отримуємо результат з відповідною точністю. Наприклад, якщо знаків після коми в наближеннях беремо чотири, то кількість нулів після коми в розв'язку в основному також буде чотири. Таку властивість простежено за кількості знаків від 2-ох до 15-ти за будь-яких комбінацій коефіцієнтів.

Описана вище властивість виконується для всіх систем такого типу з тією відмінністю, що в системах, де коефіцієнти при невідомих  $x_3, x_4, x_5, \dots, x_{10}$  – числа різного порядку, розв'язки мають іншу точність, яка залежить від порядку тих чисел.

У ході знаходження розв'язків системи (2) з використанням Maple було встановлено, що точність отриманих розв'язків залежить від кількості знаків під час обчислень (табл. 2). Тобто, якщо кількість знаків під час обчислень, наприклад, – десять, то точність отриманих розв'язків буде порядку мінус 10–11-ого степеня; а якщо кількість знаків – двадцять, то точність буде



порядка мінус двадцятого степеня. Це справедливо для будь-якої кількості знаків після коми під час обчислень.

Таблиця 2

Розв'язки системи рівнянь за різної кількості знаків

Кількість знаків після коми під час обчислень			
	3	10	20
$x_1$	0.983	1.0000000003	1.00000000000000000000
$x_2$	0.987	1.0000000001	1.00000000000000000001
$x_3$	0.0151	-1.9803119619e-10	-6.80490410561081222248e-20
$x_4$	0.00976	-1.2592095879e-10	-7.89000726997583869653e-20
$x_5$	-0.00978	7.5505452140e-11	4.90651119037705760544e-20
$x_6$	-0.00013	1.7443418014e-11	1.1e-21
$x_7$	-0.0057	3.6e-11	2.96331830823822767814e-20
$x_8$	0.00957	-1.2520885259e-10	-3.16166244016782026599 e-20
$x_9$	0.00549	-2.3665835857e-11	-1.34134009573148757442e-20
$x_{10}$	0.	2 e-11	0

Розглянемо систему рівнянь вигляду:

$$\begin{pmatrix} 1 & 2 & \dots & \dots & \dots \\ \sqrt{5} & 2\sqrt{5} & \dots & \dots & \dots \\ \sqrt{3} & 2\sqrt{3} & \dots & \dots & \dots \\ \sqrt{5} & 2\sqrt{5} & \dots & \dots & \dots \\ \sqrt{2} & 2\sqrt{2} & \dots & \dots & \dots \\ \sqrt{7} & 2\sqrt{7} & \dots & \dots & \dots \\ \sqrt{2} & 2\sqrt{2} & \dots & \dots & \dots \\ \sqrt{5} & 2\sqrt{5} & \dots & \dots & \dots \\ \sqrt{6} & 2\sqrt{6} & \dots & \dots & \dots \\ \sqrt{7} & 2\sqrt{7} & \dots & \dots & \dots \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{pmatrix} = \begin{pmatrix} 3 \\ 3\sqrt{5} \\ 3\sqrt{3} \\ 3\sqrt{5} \\ 3\sqrt{2} \\ 3\sqrt{7} \\ 3\sqrt{2} \\ 3\sqrt{5} \\ 3\sqrt{6} \\ 3\sqrt{7} \end{pmatrix}, \quad (3)$$

де у стовбцях три-десять матриці системи – числа шістнадцятого порядку.

Системи (3) має безліч розв’язків. Унаслідок розв’язання (3) в системі Matlab на основі методу визначників та за допомогою матричного методу результатів не отримано, а із використанням методу Гауса отримуємо розв’язок – числа порядку мінус п’ятнадцятого та мінус шістнадцятого степенів (табл. 3).

Таблиця 3

Розв’язок системи рівнянь за допомогою метода Гауса

$X =$
1,644227463420350e-15
2,442420516729730e-15
1,183829775283980e-15
3,675465950746420e-16
-1,212709626471670e-16
-2,737032732262120e-15
-2,197174398553280e-15
-1,829953818988590e-15
0,000000000000000e+00
0,000000000000000e+00

Оскільки коефіцієнти при невідомих  $x_3, x_4, \dots, x_{10}$  – числа порядку шістнадцятого степеня, то під час підстановки стає очевидним, що ці числа є розв’язком системи.

Досліджено розв’язки систем, які отримано із системи (3), коли замість різних коефіцієнтів, заданих квадратними коренями, взято числа з різною кількістю знаків після коми в наближеннях. Оскільки квадратні корені чисел взято через наближення, то визначники отриманих систем були вже відмінні від нуля, тобто такі системи мали єдиний розв’язок.

Розглянемо, наприклад, випадок, коли коефіцієнт  $a_{21} = \sqrt{5}$  задано через наближення з кількістю знаків після коми – від двох до десяти. Кожна із таких систем має єдиний розв’язок  $x_1 = 0, x_2 = 1,5, x_3 = x_4 = \dots = x_{10} = 0$ .

На основі методу визначників та матричного методу в Matlab отримано розв’язки, які залежать від кількості знаків після коми в наближеннях числа  $\sqrt{5}$ . Розв’язки, отримані із використанням матричного методу, наведеного в таблиці 4.

Таблиця 4

Розв'язки системи рівнянь за допомогою матричного методу

Кількість знаків після коми за наближення			
	2	5	10
$x_1$	-3,9164139376e-12	-4,8367747601e-09	-3,2001392159e-04
$x_2$	1,5000000000e+00	1,5000000032e+00	1,4999484917e+00
$x_3$	-7,3695953554e-31	-6,6386236221e-31	-8,0685451565e-33
$x_4$	-1,3459415218e-32	-4,0559050521e-32	-1,4390060756e-31
$x_5$	3,0064564583e-31	3,2549089875e-31	-1,6476873052e-31
$x_6$	1,1552811014e-30	9,7725568268e-31	-2,3126871467e-31
$x_6$	1,1552811014e-30	9,7725568268e-31	-2,3126871467e-31
$x_7$	1,1026037631e-30	8,7261447072e-31	-1,7957072312e-31
$x_8$	-7,7913310122e-31	-6,5205689232e-31	2,6281428567e-31
$x_9$	-1,1324129772e-30	-1,0363907343e-30	2,9754139403e-31
$x_{10}$	1,7743653621e-31	1,3465080026e-31	1,1734484358e-31

Точність значень  $x_1$  та  $x_2$  залежить від кількості знаків після коми в наближеннях. Ця залежність є закономірною. Чим менша кількість знаків в наближеннях, тим точність  $x_1$  та  $x_2$  буде більшою.

За наявності п'ятнадцяти знаків у наближеннях числа  $\sqrt{5}$  застосування матричного методу і методу визначників не дає змогу отримати результат. На відміну від них оперування методом Гауса дає змогу отримати розв'язки за наявності п'ятнадцяти і більше знаків після коми в наближеннях. Крім того, отримані розв'язки співпадають із розв'язками, наведеними в таблиці 3.

Розв'язки системи (3), в якій  $a_{12} = \sqrt{5}$  замінено на 2.23, отримані з використанням Maple, приведено в таблиці 5.

Таблиця 5

Розв'язки системи рівнянь

Кількість знаків після коми під час обчислення			
	10	15	20
$x_1$	3.571428571	-0.0000621109	-0.00000497281
$x_2$	-0.285714285	1.50003105545006	1.500002486409738
$x_3$	5.702676437e-10	-2.69113480183412e-14	-3.556863354945194e-15
$x_4$	-7.86908783e-11	4.13699319817521e-15	-1.492539249725768e-15

$x_5$	2.221563699e-10	1.22564547710409e-14	1.158429363372739e-15
$x_6$	-6.284064665e-10	-3.25828328314556e-15	-1e-15
$x_7$	4e-10	1.10514162186035e-14	6.243790160161115e-16
$x_8$	-3.001402621e-10	-2.06802053379884e-14	-3.142999619934048e-15
$x_9$	-1.76149945e-11	-1.01116714592186e-14	5.487993918944772e-16
$x_{10}$	-2e-10	7e-15	1e-15

Точність отриманих результатів залежить від того, з якою кількістю знаків проведено обчислення. Якщо кількість знаків після коми менша за десять, тоді похибка отриманих розв'язків є досить значною. У разі збільшення кількості знаків під час обчислення похибка стає меншою.

Аналогічні дослідження було проведено для систем лінійних алгебраїчних рівнянь розмірності 5x5, 10x10, 15x15, 20x20. Отримані результати справедливі і для цих систем.

**Досліджено системи рівнянь** різної розмірності, у яких частину коефіцієнтів системи було задано наближено, а решту порівняно великими або малими числами. Проведено аналіз розв'язків систем, отриманих за допомогою різних прямих методів за різної кількості знаків у наближеннях.

Установлено, що здебільшого використання методу Гауса дає змогу отримати точніший результат, ніж під час застосування матричного методу і методу визначників.

Якщо визначник системи дорівнює нулю, тоді оперування методом визначників і матричним методом у Matlab результату не дає. Система Maple у такому разі дає різні розв'язки залежно від того, з якою кількістю знаків проведено обчислення.

Аналіз результатів експериментальних досліджень доцільно враховувати під час розв'язування прикладних задач, для знаходження розв'язків яких необхідно розв'язувати системи лінійних алгебраїчних рівнянь.

1. Самарский А. А. Введение в численные методы : учеб. пособие для вузов по спец. «Прикл. математика» / А. А. Самарский. – М. : Наука, 1987. – 286 с.
2. Ляшенко Н. С. Чисельні методи : підручник / Н. С. Ляшенко, В. А. Головань. – К. : Либідь, 1996. – 288 с.
3. Ляшко Н. Н. Методы вычислений / Н. Н. Ляшко, В. Л. Макаров, А. А. Скоробагатко. – К. : Вища школа, 1977. – 406 с.
4. Калиткин Н. Н. Численные методы : учеб. пособие для вузов / Н. Н. Калиткин ; под ред. А. А. Самарского. – М. : Наука, 1978. – 512 с.
5. Бахвалов Н. С. Численные методы / Н. С. Бахвалов и другие. – М. : Наука, 1987. – 598 с.
6. Васильев А. Н. Maple 8. Самоучитель / А. Н. Васильев. – М. : Издательский дом «Вильямс», 2003. – 352 с.
7. Курбатова Е. А. Matlab 7. Самоучитель / Е. А. Курбатова. – М. : Издательский дом «Вильямс», 2006. – 256 с.
8. Бидасюк Ю. М. MathCAD 12. Самоучитель / Ю. М. Бидасюк. – М. : Издательский дом «Вильямс», 2006. – 224 с.
9. Мэтьюз Д. Г. Численные методы. Использование Matlab / Д. Г. Мэтьюз, К. Д. Куртис. – М. : Издательский дом «Вильямс», 2001. – 720 с.

Остапчук О. П., к.т.н., ст. викл., Гриб О. П., студент 5 курсу ННІ АКОТ (Національний університет водного господарства та природокористування, м. Рівне)

### **ЧИСЛОВЕ ДОСЛІДЖЕННЯ ОДНОВИМІРНОЇ ЗАДАЧІ МАСОПЕРЕНЕСЕННЯ СОЛЬОВИХ РОЗЧИНІВ ЗА НЕЛІНІЙНОЇ СОРБЦІЇ**

**Проведено математичне моделювання процесу масоперенесення солей в ґрунтових масивах за нелінійної сорбції. Отримано числовий розв'язок відповідної одновимірної крайової задачі за допомогою методу скінченних різниць. Створено програмне забезпечення, проведено та проаналізовано серію числових експериментів.**

**Проведено математическое моделирование процесса массопереноса солей в грунтовых массивах при нелинейной сорбции. Получено числовое решение соответствующей одномерной краевой задачи методом конечных разностей. Создано программное обеспечение, проведена серия численных экспериментов и их анализ.**

**Mathematical modeling is the process of mass transfer of salts in the soil mass in nonlinear sorption. Obtain a numerical solution of the corresponding one-dimensional boundary value problem by finite differences. The software conducted a series of numerical experiments and analysis.**

На сьогодні енергетична та інші галузі промисловості знаходяться на стадії інтенсивного розвитку, з огляду на що особливої актуальності набувають проблеми, пов'язані зі стійкістю гідротехнічних та енергетичних об'єктів, надійністю та безпекою їхньої експлуатації. Досить часто експлуатація останніх відбувається за наявності фільтрації підземних вод у їхніх основах [1, 2], у яких знаходяться дисперсно-розчинені солі або залягають у вигляді окремих включень, пластів тощо. Відтак, очевидною є виняткова актуальність питання розроблення, дослідження й упровадження методів моделювання для розв'язання задач підземного масоперенесення під час фільтрації підземних вод [6, 9].

У роботі розглянуто процес перенесення сольових розчинів у горизонтальному протяжному ґрунтовому пласті довжиною  $l$  з одного водного басейну в інший, напори в яких –  $H_1$  і  $H_2$  (рис. 1).

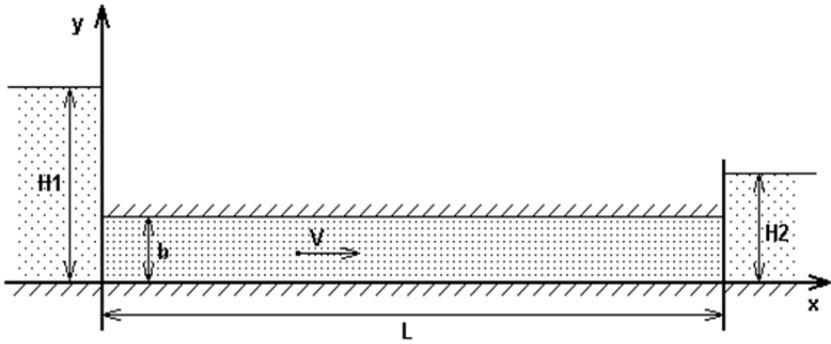


Рис.1. Фільтрація сольового розчину до водозабірника

Будемо вважати, що процес перенесення солевих розчинів у ґрунтовому пласті відбувається в ізотермічних умовах, а ґрунтове середовище є недеформівним. Крім того, вважатимемо, що  $l \gg b$  і процес масоперенесення можна розглядати в рамках одновимірної моделі.

**Математичну модель** масоперенесення солевих розчинів за одновимірної фільтрації підземних вод до водозабірника (рис. 1) з урахуванням залежності параметрів фільтрації і масоперенесення від концентрації сольового розчину, впливу осмотичних явищ на процес перенесення можна описати такою крайовою задачею [3, 4, 5]:

$$\frac{\partial}{\partial x} \left( D(x,c) \frac{\partial c}{\partial x} \right) - \frac{\partial}{\partial x} (V(x,c) \cdot c) - \gamma(c - C_*)^a = \frac{\partial(\sigma c)}{\partial t}, \quad (1)$$

$$V(x,c) = -k(x,c) \frac{\partial h}{\partial x} + v(x,c) \frac{\partial c}{\partial x}, \quad \frac{\partial V}{\partial x} = 0, \quad (2)$$

$$h(0) = H_1, \quad h(l) = H_2 \quad (3)$$

$$c(x, 0) = \tilde{C}_0(x), \quad 0 \leq x \leq l, \quad (4)$$

$$l_1 c(0, t) = \tilde{C}_1(t), \quad t \geq 0, \quad (5)$$

$$l_2 c = \begin{cases} c(l, t) = \tilde{C}_2(t), \\ \frac{\partial c(l, t)}{\partial x} = 0, \\ \left[ V(x,c) \cdot (c - \tilde{C}_2) - D(x,c) \cdot \frac{\partial c}{\partial x} \right]_{x=l} = 0. \end{cases} \quad (6)$$

де  $l_i, i = \overline{1,2}$  – оператори, що задають граничні умови на вході та виході фільтраційного потоку,  $C^*$  – концентрація граничного насичення.

Потрібно знайти розподіл концентрації сольових розчинів  $c(x,t)$  по області фільтрації з часом, розподіл поля швидкостей фільтрації ( $V(c)$ ) та напорів  $h(x)$  по області фільтрації.

На основі чисельного моделювання потрібно також установити відмінність перебігу процесів масоперенесення у припущенні незалежності величин  $k, v$  і  $D$  від концентрації сольового розчину та у разі наявності такої залежності.

Очевидно, побачити що (1)–(6) є нелінійною крайовою задачею.

Доведено [7], що в разі задання граничних умов першого роду існує єдиний класичний розв'язок крайової задачі (1)–(7).

У разі  $\left| k(x,c) \frac{\partial h}{\partial x} \right| \gg \left| v(x,c) \frac{\partial c}{\partial x} \right|$ , використовуючи (2), (3), отримаємо значення п'єзометричного напору на  $[0,1]$  за відомого розподілу концентрації  $c(x,t)$ ,  $x \in [0,1]$

$$h(x,t) = \frac{H_2 - H_1}{l} \cdot \frac{x}{\int_0^x \frac{ds}{k(c(s,t))}} + H, \quad (7)$$

а звідси і значення швидкості фільтрації  $V(x, c(x,t))$

$$V(x, c(x,t)) = \frac{H_1 - H_2}{l} \cdot \frac{1}{\int_0^x \frac{dx}{k(c(x,t))}}, \quad (8)$$

де  $x \in \{0; l\}$ .

За наявності осмосу швидкість фільтрації обчислюють за формулою:

$$V(x, c(x,t)) = \frac{H_1 - H_2 \pm \int_0^l \frac{v(c(x,t))}{k(c(x,t))} \cdot \frac{\partial c}{\partial x} dx}{\int_0^l \frac{dx}{k(c(x,t))}}. \quad (9)$$

Для отримання числового розв'язку задачі (1)–(6) введемо рівномірну сітку  $\omega_{h_l \tau}$  з кроками  $h_l$  і  $\tau$  по осях  $Ox$  та  $Ot$  відповідно:

$$\omega_{h_l \tau} = \left\{ (x_i, t_j) \left| \begin{array}{l} x_i = i \cdot h_l, t_k = k \cdot \tau, i = \overline{0, N_1}, k = \overline{0, N_2}, \\ h_l \cdot N_1 = l, \tau \cdot N_2 = T. \end{array} \right. \right\}.$$

Для крайової задачі (1)–(6) побудуємо монотонну різницеву схему, яка має порядок апроксимації  $O(h_1^2 + \tau)$  [8]:

$$\begin{aligned} & \frac{\mu_i^k}{h_1} \left( \frac{C_{i+1}^{k+1} - 2 \cdot C_i^{k+1} + C_{i-1}^{k+1}}{h_1^2} \right) + r_{+i}^k \cdot \frac{C_{i+1}^{k+1} - C_i^{k+1}}{h_1} + \\ & + r_{-i}^k \cdot \frac{C_i^{k+1} - C_{i-1}^{k+1}}{h_1} - \gamma(C_i^{k+1} - C_*)^a = \sigma \frac{C_i^{k+1} - C_i^k}{\tau}, \end{aligned} \quad (10)$$

$$C_i^0 = \tilde{C}_0(x_i), \quad x_i = i \cdot h_1, \quad i = \overline{0, N_1}; \quad (11)$$

$$C_0^k = \tilde{C}_1(t_k), \quad t_k = k \cdot \tau, \quad k = \overline{0, N_2}; \quad (12)$$

$$C_{N_1}^k = \tilde{C}_2(t_k), \quad t_k = k \cdot \tau, \quad k = \overline{1, N_2}; \quad (13)$$

де

$$V_i^k = -k_i^k \frac{h_{i+1}^k - h_{i-1}^k}{2h_1} + \nu \frac{C_{i+1}^k - C_{i-1}^k}{2h_1}.$$

$$\begin{aligned} \mu_i^k &= \frac{1}{h_1 \cdot |r_i^k| + \frac{2D_i^k}{2D_i^k}} & r_i^k &= r_{+i}^k + r_{-i}^k \\ r_{-i}^k &= \frac{-V_i^k - |V_i^k|}{2} \leq 0, & r_{+i}^k &= \frac{-V_i^k + |V_i^k|}{2} \geq 0, \end{aligned}$$

Запишемо (10)–(13) у канонічному вигляді. Маємо:

$$\left( \frac{2\mu_i^k}{h_1^2} - \frac{r_{-i}^k}{h_1} + \frac{r_{+i}^k}{h_1} + \frac{\sigma}{\tau} \right) C_i^{k+1} - \left( \frac{\mu_i^k}{h_1^2} - \frac{r_{-i}^k}{h_1} \right) C_{i-1}^{k+1} - \left( \frac{\mu_i^k}{h_1^2} + \frac{r_{+i}^k}{h_1} \right) C_{i+1}^{k+1} - \frac{\sigma}{\tau} C_i^k = \gamma \cdot (C_i^k - C_*)^a, \quad (14)$$

де

$$a_i = \left( \frac{\mu_i^k}{h_1^2} - \frac{r_{-i}^k}{h_1} \right); \quad b_i = \left( \frac{\mu_i^k}{h_1^2} + \frac{r_{+i}^k}{h_1} \right);$$



$$c_i = \left( \frac{2\mu_i^k}{h_1^2} - \frac{r_{-i}^k}{h_1} + \frac{r_{+i}^k}{h_1} + \frac{\sigma}{\tau} \right); \quad F_i^k = \frac{\sigma}{\tau} C_i^k - \gamma (C_i^k - C_*)^\alpha.$$

Отже, побудована різницева схема має порядок апроксимації  $O(h_1^2 + \tau)$  відповідної диференціальної крайової задачі і є монотонною за  $\forall h_1, \tau > 0$ . Тому до цієї різницевої схеми можна застосувати принцип максимуму [9] та його наслідки, з чого можна робити висновок про коректність цієї різницевої схеми і збіжність її розв'язку і розв'язку відповідної крайової задачі. Розв'язок різницевої схеми на кожному часовому шарі отримано за допомогою методу прогонки. Монотонність різницевої схеми гарантує стійкість методу прогонки:

$$C_i^{k+1} = \alpha_{i+1} C_{i+1}^{k+1} + \beta_{i+1}$$

де

$$\alpha_{i+1} = \frac{b_i}{c_i - a_i \alpha_i}; \quad \beta_{i+1} = \frac{a_i B_i + F_i^k}{c_i - a_i A_i};$$

$$\alpha_1 = 0; \quad \beta_1 = C_0^{k+1};$$

Кількість мігруючих речовин, яку виносить фільтраційний потік за час  $t$  з області фільтрації, можна обчислити за формулою:

$$Q_C = \int_0^t c(l, t) \cdot V(x_*, c(x_*, t)) dt - \int_0^t D(x_*, c(x_*, t)) \cdot \frac{\partial c(l, t)}{\partial x} dt. \quad (15)$$

У разі задання оператором  $l_2c$  граничних умов (6), відповідно, маємо:

$$Q_C = \int_0^t \tilde{C}_2(t) \cdot V(x_*, c(x_*, t)) dt - \int_0^t D(x_*, c(x_*, t)) \cdot \frac{\partial c(l, t)}{\partial x} dt, \quad (16)$$

$$Q_C = \int_0^t V(x_*, c(x_*, t)) \cdot c(l, t) dt, \quad (17)$$

$$Q_C = \int_0^t V(x_*, c(x_*, t)) \cdot \tilde{C}_2(t) dt. \quad (18)$$

Використовуючи (16)–(18), можна визначити масу солей, яка потрапляє у басейн стічних вод за проміжок часу  $t$ , а також знайти, за який проміжок часу у басейн потрапить певна маса забруднень.

**Програмну реалізацію** обчислюваного алгоритму розв'язку задачі представлено в інтегрованому середовищі програмування Visual Studio 2010 за допомогою мови об'єктно-орієнтованого програмування C#.

Для реалізації алгоритму було використано такі вхідні дані:

$\sigma=0,9$ ;  $\gamma=0,00065$ ;  $c_*=50 \text{ кг/м}^3$ ;  $C1=50 \text{ кг/м}^3$ ;  $C2=5 \text{ кг/м}^3$ ,  $T=360 \text{ діб}$ ,  $l=10 \text{ м}$ ,  $D=0,1 \text{ м}^2/\text{добу}$ ,  $H_1=1,5 \text{ м}$ ,  $H_2=0,5 \text{ м}$ ,  $k=2 \text{ м/добу}$ .

На основі програмної реалізації було проведено низку чисельних експериментів для розв'язання задачі з граничними умовами 1-го роду (рис. 2, рис. 3).

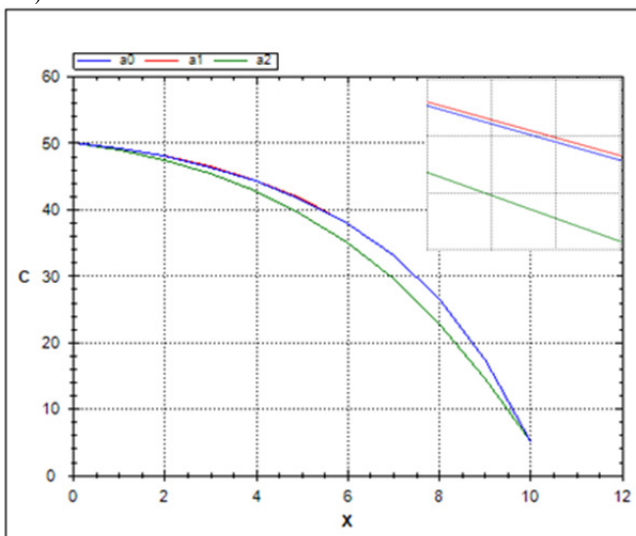


Рис. 2. Порівняльний графік розподілу концентрації за наявності різних значень коефіцієнта сорбції за  $t=30$  діб

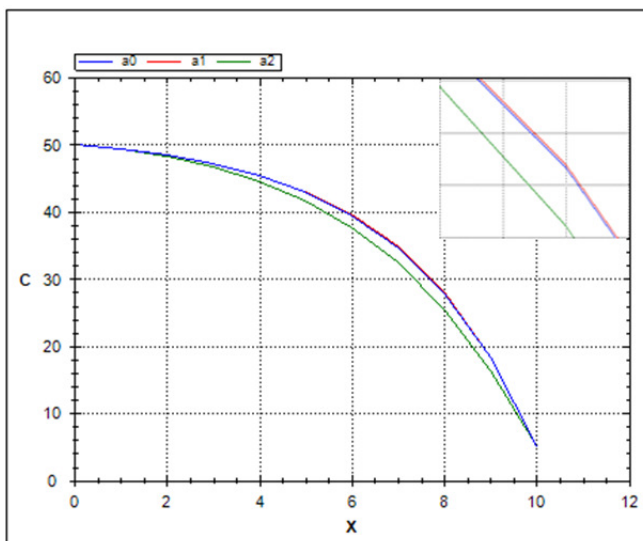


Рис. 3. Порівняльний графік розподілу концентрації за наявності різних значень коефіцієнта сорбції за  $t=200$  діб

**На основі аналізу проведених числових експериментів** та отриманих графічних зображень можна зробити такі висновки:

- за умови  $a=0$ ,  $a=1$  розподіли концентрації сольових розчинів по області фільтрації суттєво не відрізняються, за умови  $a=2$  значення концентрації сольових розчинів зменшується порівняно з  $a=0$ ,  $a=1$  в будь-якій точці області фільтрації з часом (рис. 2, рис. 3);
- чим більша пористість ґрунту, тим меншою є концентрація солей;
- чим більшим є коефіцієнт конвективної дифузії, тим менша концентрація солей;
- зі збільшенням коефіцієнта фільтрації відбувається збільшення швидкості фільтрації й у результаті – зменшення концентрації сольового розчину;
- збільшення коефіцієнта масообміну  $\gamma$  спричиняє збільшення концентрації сольових розчинів.

Отже, під час математичного моделювання процесу масоперенесення речовини необхідним є врахування залежності параметрів фільтраційного потоку від його фізико-хімічних властивостей та властивостей пористого середовища.

1. Аравин В. И. Теория движения жидкостей и газов в недеформируемой пористой среде / В. И. Аравин, С. Н. Нумеров. – М. : Гостехиздат, 1953. – 616 с. 2. Веригин Н. Н. Распространение фронта загрязнения в районах наземных и подземных хранилищ с учетом диффузии и равновесного массообмена / Н. Н. Веригин, А. В. Шибанов // Труды ин-та ВОДГЕО. – 1975. – Вып. 54. – С. 6–17. 3. Власюк А. П. Математичне моделювання процесів масопереносу в областях з криволінійними фіксованими границями / А. П. Власюк // Вісник УДАВГ. – Вип. 1. Ч. 1. – Рівне, 1998. – С. 203–207.
4. Власюк А. П. Про деякі нові математичні моделі підземної гідромеханіки / А. П. Власюк // Вісник РДТУ. – Вип. 3(5). Ч. 2. – 2000. – С. 57–62. 5. Власюк А. П. Чисельне дослідження задачі масопереносу сольових розчинів у ґрунтових масивах з врахуванням осмосу / А. П. Власюк, О. П. Остапчук // Проблеми математичного моделювання сучасних технологій : тези Міжн. наук.-техн. конф. – 2002. – С. 24.
6. Ляшко И. И. Численное решение задач тепло- и массопереноса в пористых средах / И. И. Ляшко, Л. И. Демченко, Г. Е. Мистецкий. – К. : Наук. думка, 1991. – 262 с.
7. Мартинюк П. М. Дослідження існування та єдності розв'язку однієї квазілінійної параболічної системи диференційних рівнянь другого порядку / П. М. Мартинюк, А. П. Власюк // Препр. НАН України. Ін-т математики; 2001.03. – К. : 2001. – 20 с.
8. Самарский А. А. Теория разностных схем / А. А. Самарский. – [3-е изд.]. – о М. : Наука, 1989. – 616 с. 9. Сергиенко И. В. Математическое моделирование и исследование процессов в неоднородных средах / И. В. Сергиенко, В. В. Скопецкий, В. С. Дейнека. – К. : Наук. думка, 1991. – 432 с.

Герус В. А., ст. викладач, Сідлярук І., студентка 3 курсу ННІ АКОТ  
(Національний університет водного господарства та природокористування,  
м. Рівне)

## ДОСЛІДЖЕННЯ РОБОТИ КЛІТКОВИХ АВТОМАТІВ

**Досліджено роботу кліткових автоматів на прикладі моделювання динаміки натовпу. Описано програму, яка моделює рух натовпу з використанням кліткових автоматів.**

**Study work on the example of cellular automata modeling the dynamics of crowds. Described a program that simulates the movement of the crowd using cellular automata**

У ході розгляду питань безпеки й управліннь ризиками в умовах сучасного великого міста виникає, серед інших, задача вивчення руху неорганізованих груп людей за наявності різних перешкод цьому руху. Очевидно, що таке вивчення доцільно проводити шляхом побудови і дослідження моделей означених процесів.

У міру появи все більш крупних громадських споруд, на зразок гіпермаркетів, великих торговельних центрів, спортивних споруд, посилюється актуальність проблеми динаміки натовпу. Моделювання людських потоків із подальшим використанням результатів у процесі проектування будівель відіграє ключову роль з точки зору безпеки й організації евакуації під час стихійних лих, аварій, пожеж і терактів. У названих ситуаціях найважливішою загрозою здоров'ю і життю людей часто є паніка.

Під час проектування споруд, розрахованих на розташування у них значної кількості людей, доцільно змоделювати ситуацію безладного руху великої неорганізованої групи людей (яку надалі називатимемо скорочено натовпом) в умовах паніки й усунути особливості конструкцій, що можуть призвести до виникнення заторів і тисняви. Крім того, жителі великих міст щодня стикаються з проблемою руху в натовпі під час користування громадським транспортом (наприклад, у підземних переходах, у турнікетах і на ескалаторах метро). Планування будівництва міських пішохідних комунікацій доцільно було б вести, не тільки враховуючи загальний потік пасажирів, але і детально розглядаючи рух людей на конкретних ділянках шляху.

У замкнутому просторі (під час проведення концерту або іншого масового заходу) в разі виникнення небезпеки люди починають одночасно всі шукати виходу з приміщення, щоб урятуватися. Здебільшого це відбувається

хаотично. Особливо активними стають люди, що знаходяться далеко від виходів. Вони починають щосили напірати на тих, хто попереду, й у результаті більшість «передніх» виявляються притиснутими до стін. Утворюється тиснява, що може призвести до масової загибелі людей, які опиняються між кам'яною стіною і стіною людських тіл.

У математичному моделюванні ситуацій, в яких активно діють люди, відомою є проблема неможливості на сучасному етапі розвитку науки повного математичного опису поведінки окремо взятої людини з огляду на зумовленість її дій широким спектром чинників – і раціональних, й ірраціональних. Проте поведінка великої групи людей у стандартній ситуації є легко прогнозованою і добре описуваною імовірнісним чином. На основі врахування закону великих чисел: навіть якщо одна людина з певних причин вирішить діяти нетривіально, її дії ніяк не вплинуть на групу загалом.

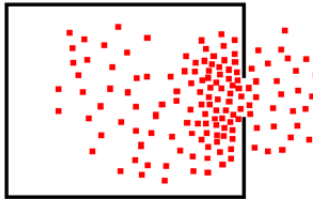


Рис. 1. Моделювання натовпу

Для математичного моделювання динаміки натовпу виявилось можливим застосувати клас украй спрощених дискретних моделей – кліткових автоматів, які раніше успішно використовували в багатьох інших сферах досліджень.

*Клітковий автомат* є дискретною динамічною системою, поведінка якої повністю визначається в термінах локальних залежностей [2]. Основними властивостями класичної моделі кліткового автомата є локальність правил, однорідність системи, скінченність множини можливих станів клітки, одночасність змін значень у всіх клітках наприкінці ітерації, а не в міру обчислення.

Клітковий автомат, що моделює рух натовпу, повинен аналізувати стани кліток і відповідно до отриманих результатів визначати напрям зміщення кожної клітки, «пам'ятаючи» про пріоритетний напрям руху. Також цей клітковий автомат повинен зберігати число кількості кліток, що характеризує наявність людей у цих точках.

Запропонований нами клітковий автомат відзначається такими властивостями: вибрано квадратну решітку; як окіл вибрано тип сусідства за Муром, тобто для кожної клітки будемо розглядати сусідів першого кільця – безпосередніх сусідів, тобто вісім кліток, які оточують аналізовану клітку. Нехай кожна з них може набувати значення 1 або 0 (у термінології теорії

кліткових автоматів кожна клітка може знаходитися в одному із двох цих станів), що відповідають наявності або відсутності людини в цій точці. На кожному кроці для кожної клітки поля кліткового автомата, що знаходиться в стані 1, далі розрахуємо ймовірність зміщення людини з цього місця розташування в одну із восьми сусідніх кліток.

Для цього перевіряємо стан сусідніх кліток. Наявність порожнього місця у клітинці, що знаходиться в тому боці, в який відбувається рух всієї групи людей, означає, що саме в цю клітку буде здійснено перехід на наступній ітерації. Якщо ця клітка вже зайнята, то перевіряємо стан решти кліток, в результаті чого відбувається рух до тієї клітки з околу (якщо це можливо), яка є найбільш зміщеною в бік руху щодо клітки, яку розглядаємо.

Якщо всі клітки околу – зайняті, то ця клітка не змінює своє місце розташування на наступній ітерації.

Також перевіряємо стани тих кліток, в які буде проходити зміщення для визначення, чи не було проведено зміщення в ці клітки раніше, тобто чи не перейшло декілька людей в одну і ту ж точку. Якщо така ситуація мала місце, то випадковим чином виберемо  $n-1$ , ( $2 \leq n \leq 8$ ) сусідню клітку, яка знаходиться в стані 0 (такі знайдуться, адже саме з них було проведено зміщення в цю клітку). Присвоїмо всім вибраним кліткам, а також центральній, значення 1.

Описані вище правила задають динаміку моделі, за якої кожна клітка переміщується у вибраному напрямі, прагнучи обходити перешкоди такому руху в напрямі, найбільш вільному для руху. Треба відзначити, що заданий таким чином закон зміни конфігурації поля зберігає кількість кліток, що знаходяться в стані 1, що є природною вимогою до моделі.

Для дослідження роботи кліткових автоматів було розроблено програму в середовищі програмування Delphi7.

Під час розроблення програмного продукту було реалізовано такі задачі:

- моделювання руху великої неорганізованої групи людей в умовах наявності перешкод на шляху;

- класичний клітковий автомат – гра «Життя» Джона Хортон Конвея.

Для реалізації цих задач було організовано таку функціональність програми:

- надання користувачам дружнього інтерфейса взаємодії;
- можливість зручного та наочного спостереження за еволюцією станів решітки;
- можливість налаштовувати стан решітки кліткового автомата для вирішення конкретного завдання;
- зручне подання результатів експериментів у вигляді графіків тощо;
- зберігання і відновлення з файла станів решітки.

### **Інтерфейс програми.**

Вигляд головного вікна програми зображено на рис. 2.

Користувачеві надано можливість створювати моделі, що охоплюють довільний тип небезпечного простору, кількісний склад і початкове зосередження натовпу за допомогою миші. Лівою кнопкою миші користувач створює клітки, які пересуватимуться полем чи будуть створювати перешкоди (залежно від вибраного кольору). Неправильно введений об'єкт можна видалити шляхом натиснення правої кнопки миші по цьому об'єкту. На чорній смузі, що знаходиться над полем, виставлено граничне значення замкнутої ділянки, в разі досягнення якої клітку можна вважати такою, що покинула небезпечну ділянку.

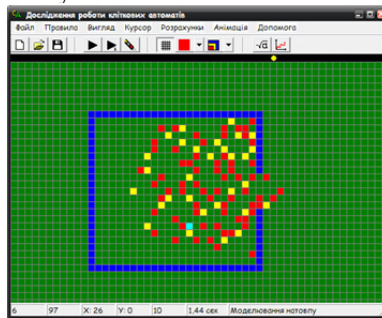


Рис. 2. Головне вікно програми

Різний колір кліток характеризує різний тип їхнього руху. Так, червоний колір означає, що ця клітка є рухомою та рухається зі швидкістю 1,5 м/с. Жовтий колір означає, що ця клітка є рухомою та рухається зі швидкістю 0,75 м/с. Жовті та червоні клітки «намагаються» рухатись вправо. Блакитний колір означає, що ця клітка є рухомою та рухається зі швидкістю 1,5 м/с, але напрям руху можна вибирати довільний з усіх можливих напрямів. Синій колір означає, що ця клітка нерухома, тобто є перешкодою.

Для збереження конфігурацій з метою їхньою подальшого відновлення передбачено можливість збереження конфігурації моделі у файл. Такий файл має розширення .cla. Для швидкого доступу до функцій програми команди меню, часто використовувані, дублюють на панелі завдань у вигляді кнопок із відповідними малюнками.

Також передбачено можливість покрокового й автоматичного перегляду результатів моделювання. Для покрокового перегляду необхідно вибрати пункт меню «Один цикл» чи кнопку із відповідним зображенням, або натиснути пробіл.

Для автоматично перегляду потрібно вибрати пункт меню «Пуск» або натиснути клавішу F9, після чого з'явиться форма вибору параметрів запуску (рис. 3), на якій можна задати число кількості ітерацій або час, протягом яких буде відбуватися робота кліткового автомата.

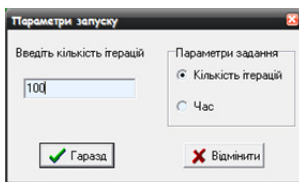


Рис. 3. Параметри запуску

За допомогою команди «Розмір клітинки» з пункту меню «Вигляд» можна змінювати розмір клітинок, вибираючи один із заданих варіантів або ж встановивши самостійно за допомогою команди «Свій» (рис. 4).

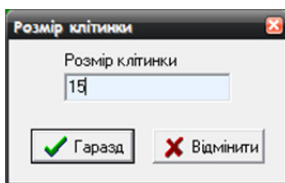


Рис. 4. Розмір клітинки

По замовчуванню розмір клітинки встановлюється таким, що дорівнює 10, тобто елемент кліткового автомата буде зображено у вигляді квадрата зі стороною 10 пікселів.

Для відображення решітки на полі кліткового автомата можна використовувати команду «Решітка» з пункту меню «Вигляд» або відповідну кнопку на панелі завдань.

За допомогою команди «Очистити» з пункту меню «Вигляд» можна видалити всі рухомі клітки, залишивши лише перешкоди.

У програмі реалізовано два кліткові автомати: моделювання руху натовпу та гру «Життя». Для вибору правил того чи іншого автомата можна скористатись пунктом меню «Правила».

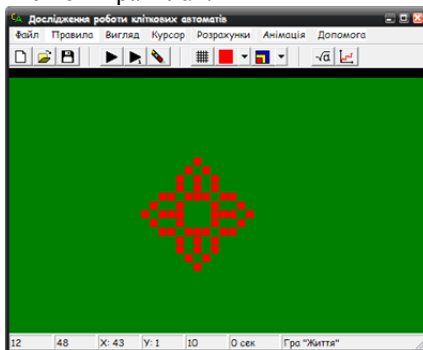


Рис. 5. Гра «Життя»



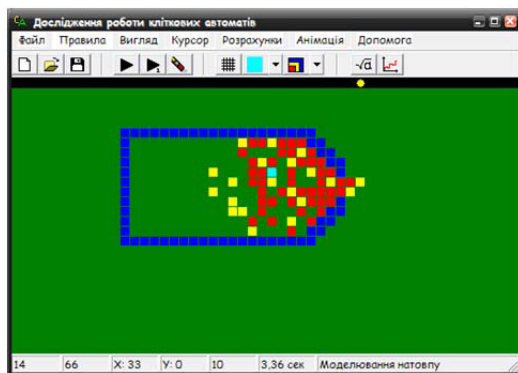


Рис. 6. Моделювання руху натовпу

У даній програмі передбачено розрахунок даних. Для цього потрібно вибрати команду «Визначити площу» з пункту меню «Розрахунки». Тут можна отримати інформацію про площу замкнутого простору, його ширину та довжину, максимальну кількість людей, яких може вмістити цей простір, та про те, скільки на цей момент там знаходиться людей, а також про середню густоту натовпу (рис. 7).

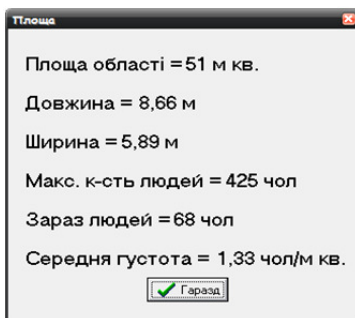


Рис. 7. Розрахунки

У програмі всі розрахунки проводять з урахування реальних пропорцій, що дає змогу застосовувати цей програмний продукт до реальних об'єктів і отримувати реальні результати.

Після закінчення роботи алгоритму можна вивести статистику кількості кліток у небезпечній зоні по всіх ітераціях (або проміжках часу) у вигляді графіка (рис. 8).

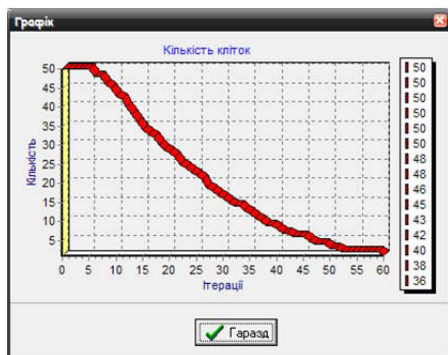


Рис. 8. Графік статистики

Із використанням цієї програми було проведено низку тестів.

За однакової початкової кількості людей було застосовано різні типи звужень деякого коридору (рис. 9, 10).



Рис. 9. Звуження під кутом  $90^\circ$



Рис. 10. Звуження під кутом  $45^\circ$

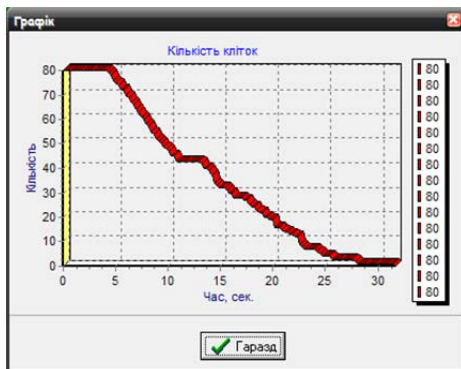


Рис. 11. Результат тесту в разі звуження під кутом  $90^\circ$

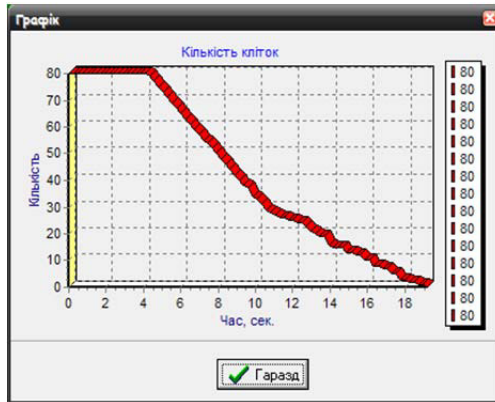


Рис. 12. Результат тесту в разі звуження під кутом 45°

За результатами, визначено, що в разі звуження першого типу (рис. 9) затори утворюються за відносно невеликої кількості людей, у результаті чого час евакуації збільшується (рис. 11); з'ясовано, що в разі звуження другого типу (рис. 10) затори майже не виникали, а час евакуації був значно меншим порівняно з першим типом звуження (рис. 12).

1. Фон Нейман Дж. Теория самовоспроизводящихся автоматов / Дж. Нейман Фон ; пер. с англ. – М. : Мир, 1971. 2. Тоффоли Т. Машины клеточных автоматов / Т. Тоффоли, Н. Марголюс; пер. с англ. – М. : Мир, 1991. 3. Gardner M. The Fantastic Combinations of John Conway's New Solitaire Game "Life" / M. Gardner // Scientific American. – 1970. – №223. – pp. 120–123. 4. Наумов Л. Метод введения обобщенных координат и инструментальное средство для автоматизации проектирования программного обеспечения вычислительных экспериментов с использованием клеточных автоматов : дисс. на соискание ученой степени канд. техн. наук / Л. Наумов. – Санкт-Петербург, 2007. 5. Малинецкий Г. Г. Моделирование динамики движения толпы при помощи клеточных автоматов с окрестностью Марголюса / Г. Г. Малинецкий, М. Е. Степанцов // Изв. вузов. Прикл. нелинейная динамика. – 1997. – Т. 5. – № 5. – С. 75–79.

Степанченко О. М., к.т.н., ст. викладач, Харів Н. О., ст. викладач, Чугай М. І., студент 5 курсу ННІ АКОТ (Національний університет водного господарства та природокористування, м. Рівне)

## **РОЗРОБКА ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ РОЗМІЩЕННЯ КОНТЕНТА НА WEB-САЙТАХ**

Статтю присвячено питанням оцінювання ефективності візуальної реклами загалом та її компонентів зокрема – і на етапі її створення, і під час використання. Питання ефективності реклами – одне із найбільш складних і важливіших питань, що стоять перед рекламістами. Це є предметом постійної заклопотаності творців реклами. Як зробити рекламу максимально ефективною? Однозначної відповіді на це запитання не дає жоден фахівець, оскільки виділити чистий ефект, отриманий від реклами, вкрай складно, тому що на цей же результат – продаж товару/послуги – працюють й інші складові маркетингового комплексу: товар, ціна, дистрибуція й уся система маркетингових комунікацій, частиною якої є реклама.

**Ключові слова:** медіаджерела, веб-камера, відносні координати, ефективність витрати, товарозвернення, оптимальна траєкторія, перешкода, мінімальний час.

**This paper focuses on the evaluation of the effectiveness of visual advertising as a whole and its individual components at the stage of its inception and during use. Indeed, the effectiveness of advertising - one of the most complex and critical issues facing advertisers. It is a matter of constant concern of the creators of advertising. How to make the most effective advertising? There is no single answer to this question makes no expert as to allocate the net effect of the advertising is extremely difficult. Because for the same result - the sale of goods / services - work and other components of the marketing mix: product, price, distribution and marketing communications entire system, part of which is also advertising.**

**Вступ. Актуальність роботи.** У цей час, мабуть, немає жодної людини, якої б так чи так не стосувалася реклама. Не можна собі уявити, що 10–15 років тому під час перегляду кінофільму глядач не зазнавав впливу реклами, що пропонує купити, спробувати, вкласти і т.д. Сьогоднішнє засилля реклами може радувати чи дратувати, але такою є реальність. У нашій країні реклама таких масштабів з'явилася зовсім недавно, а відтак, це

зумовлює наявність широкого спектру невирішених питань, пов'язаних із нею.

Визначення ефективності рекламних кампаній, як важливого контролюючого елемента рекламної діяльності, видається актуальною проблемою. Рекламодавцві бажано налагодити облік ефективності рекламування, що уможливить отримання інформації про доцільність й результативність реклами, визначити умови оптимального впливу на потенційних покупців.

Більшість фахівців вважає, що витрати на рекламу потрібно розглядати як неминучі, як-от витрати на дослідження, навчання, обладнання, тобто для досягнення запланованого обсягу реалізації необхідно підтримувати певний, вироблений ринковою практикою, відсоток відрахувань на рекламу, що залежить від цього обсягу.

Ефективність рекламних кампаній – складний, багатоаспектний й часто тривалий процес, який здійснює колектив фахівців, які, як правило, представляють різні організації. Такий процес прямо залежить від вибору й використання методів і форм керування, відповідних до вимог складної маркетингової ситуації.

**Розробленість.** Попри наявність достатньої кількості перекладної й вітчизняної літератури з проблем реклами, відчутними є гострий дефіцит ефективних ідей і нових розв'язків у всіх сферах рекламної діяльності.

У посібниках й інструкціях здебільшого розглянуто рекламу загалом, а не окремі її види, практично не висвітлено питання ефективності різних засобів реклами. Втім, у теоретичних і практичних дослідженнях усе ж таки певною мірою опрацьовано візуальну рекламу. Зокрема, все більшої популярності та теоретичної обґрунтованості у нашій країні набуває зовнішня реклама: щити, тумби, електронні табло і т.д. Незважаючи на це, накопичених знань є недостатньо, що й зумовлює актуальність дослідницьких пошуків в окремому напрямі.

Історія вивчення проблем ефективності реклами є досить давньою: охоплює використання широкого спектра методів вивчення, застосування різних методологій дослідження й інструментарію, інтерпретацію отриманих результатів і на основі доморощених засобів, й із застосуванням методів аналізу, запозичених із таких дисциплін: економіка, соціологія, психологія. Проте результат наукових пошуків на сучасному етапі можна вважати швидше екстенсивним: замість однозначної відповіді або чітких системи розв'язків на сьогодні напрацьовано багатий набір підходів, концепцій, точок зору тощо. Йдеться про більш-менш коректні локальні розв'язки цього завдання, але не про певну загальну відповідь. Щодо питання ефективності реклами з'явилося багато різних езотеричних теорій, міфів та навіть анекдотів.

Загалом питання про оцінювання ефективності реклами дотепер залишається відкритим. Непрямою ознакою неоднозначності й заплутаності

питання оцінювання ефективності реклами є кількість запропонованих дослідниками моделей сприйняття її споживачами – більше ніж 50 у двох з половиною сотнях різних книг, статей і монографій.

Небезпечність такого стану вивчення аналізованого питання полягає у неухильному зростанні кількості різних теорій, гіпотез, моделей, точок зору тощо, що зумовлює подальше „заплутування” у його трактуванні.

З огляду на вищевикладене і на те, що ефективність – це основна вимога до реклами, вивчення її варто визнати на сьогодні особливо актуальним.

**Мета і задачі дослідження.** Метою досліджень є розроблення програмного продукту, який би дозволяв оцінити ефективність візуальної реклами загалом та її компонентів зокрема – і на етапі її створення, і під час використання.

Відповідно до сформульованої мети для її досягнення у роботі було поставлено та вирішено такі завдання:

1. Провести огляд методів оцінювання ефективності візуальної реклами.
2. Провести огляд технологій програмного відстеження погляду людини.
3. Сформулювати фізичну постановку задачі відстеження погляду.
4. Розробити математичну модель задачі відстеження погляду людини.
5. Здійснити програмну реалізацію такої математичної моделі.
6. Розробити методологію застосування технології відстеження погляду людини до оцінювання ефективності візуальної реклами.

**Словесна постановка задачі.** З огляду на те, що ефективність реклами – це час, упродовж якого її переглядає людина, задача полягає у визначенні точки погляду з потоку відео веб-камери за допомогою відстежування руху ока. Для реалізації цієї задачі її потрібно розділити на 2 основні підзадачі:

1) перша задача полягає у визначенні місця розташування ока на кожному кадрі, тобто у розпізнаванні ока на кожному кадрі та отримуванні його координат розміщення;

2) друга задача полягає у зіставленні місця розташування ока на потоці відео з точками на екрані, в які насправді дивиться користувач, тобто у реалізації процесу проектування координат ока на координати екрану.

**Математичний розв’язок задачі.** Побудуємо математичну модель відстежування погляду згідно з рис. 1, де  $x$  – поточна координата на осі  $X$ ;  $y$  – поточна координата на осі  $Y$ ;  $x'$  – поточна координата на осі  $X'$ ;  $y'$  – поточна координата на осі  $Y'$ ;  $R$  – радіус від центра ока до лівого верхнього кута;  $r$  – радіус мітки ока.

$$\begin{cases} \frac{x}{R} = \frac{x'}{r} \\ \frac{y}{R} = \frac{y'}{r} \end{cases} \quad (1)$$

Тоді (1) – це відношення, з якого знаходимо  $x$ ,  $y$ .

$$\begin{cases} x = \frac{R}{r} x' \\ y = \frac{R}{r} y' \end{cases} \quad (2)$$

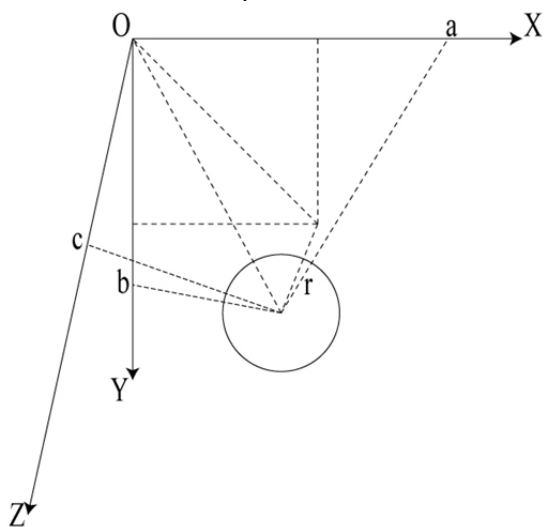


Рис. 1. Математична модель відстежування погляду людини у трьохвимірному випадку

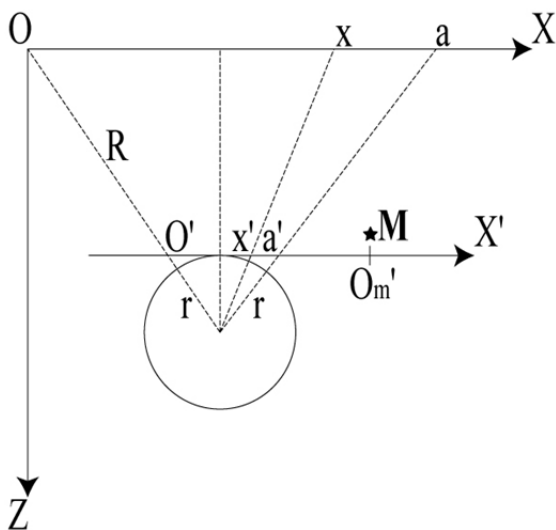


Рис. 2. Математична модель відстежування погляду людини у двохвимірному випадку

Проведемо початкову ініціалізацію фіксування погляду. Для цього зафіксуємо зір на екрані в точках  $o, a, b$ , де  $o$  – це верхній лівий кут екрана;  $a$  – ширина екрана (верхній правий кут екрана);  $b$  – висота екрана (нижній лівий кут екрана).

Фіксуємо координати мітки  $M = O_m'$ , де  $M$  – допоміжна мітка на обличчі людини.

Визначаємо  $o', a', b', O_m'$  на осі  $O'X'$  (рис. 2), де  $o'$  – проекція  $o$  на вісь  $O'X'$ ;  $a'$  – проекція  $a$  на вісь  $O'X'$ ;  $b'$  – проекція  $b$  на вісь  $O'Y'$ ;  $O_m'$  – проекція мітки  $M$  на вісь  $O'X'$ :

$$\begin{cases} \frac{a}{R} = \frac{a'}{r} \\ \frac{b}{R} = \frac{b'}{r} \end{cases} \quad (3)$$

$$\begin{cases} a = \frac{R}{r} a' \\ b = \frac{R}{r} b' \end{cases} \quad (4)$$

Ділимо (2) на (4) й отримуємо:

$$\begin{cases} \frac{x}{a} = \frac{x'}{a'} \\ \frac{y}{b} = \frac{y'}{b'} \end{cases} \quad (5)$$

Після зміщення ока у нове місце розташування концентруємо зір на тому ж об'єкті на екрані. При цьому мітка  $M$  змінила координати на  $x'_M$  і  $y'_M$ .

$$\begin{cases} \frac{x}{a} = \frac{x' + x'_M}{a'}; x = [0, a]; x' = [0, a'] \\ \frac{y}{b} = \frac{y' + y'_M}{b'}; y = [0, b]; y' = [0, b'] \end{cases} \quad (6)$$

Уведемо відносні координати  $X, Y, X', Y'$ .

$$\begin{aligned} X = \frac{x}{a}; Y = \frac{y}{b}; X' = \frac{x'}{a'}; Y' = \frac{y'}{b'} \\ X = [0, 1]; Y = [0, 1]; X' = [0, 1]; Y' = [0, 1] \end{aligned} \quad (7)$$

$$\begin{cases} X = X' + X'_M; X'_M = \frac{x'_M}{a'} \\ Y = Y' + Y'_M; Y'_M = \frac{y'_M}{b'} \end{cases} \quad (8)$$



У результаті отримуємо величини, що нівелюють залежність від роздільної здатності екрана. Розіб'ємо екран на комірки:

$$\begin{cases} x = 0, \frac{a}{k}, 2\frac{a}{k}, 3\frac{a}{k}, \dots, k\frac{a}{k} \\ y = 0, \frac{b}{e}, 2\frac{b}{e}, 3\frac{b}{e}, \dots, e\frac{b}{e} \end{cases} \quad (9)$$

У відносних координатах екран розіб'ється на такі комірки:

$$\begin{cases} X = 0, \frac{1}{k}, 2\frac{1}{k}, 3\frac{1}{k}, \dots, k\frac{1}{k} \\ Y = 0, \frac{1}{e}, 2\frac{1}{e}, 3\frac{1}{e}, \dots, e\frac{1}{e} \end{cases} \quad (10)$$

Алгоритм вимірювань. Вимірювання:

1. На момент часу  $t_0 = 0$  фіксуємо місце розташування, а також зміщення міток ока  $(x', y')$  і мітки  $M(x'_M + y'_M)$ .

2. Обчислюємо відносну координату  $X$ :

$$X = \frac{x}{a} = \frac{X' + X'_m}{a'} \quad (11)$$

3. За формулою  $m = Xk$  (заокруглення до цілого значення) визначаємо номер комірки  $m = [0, k]$ .

4. Обчислюємо відносну координату  $Y$ :

$$Y = \frac{y}{b} = \frac{Y' + Y'_m}{b'} \quad (12)$$

5. За формулою  $n = Ye$  (заокруглення до цілого значення) визначаємо номер комірки  $n = [0, e]$ .

6. Додаємо до пам'яті комірки за номером  $(m, n)$  одиницю.

Для моменту часу  $t = t_0 + \Delta t$  повторюємо пункти 1–6.

$$t = t_0 + N\Delta t \quad (13)$$

Частотність активації кожної комірки обчислюємо за формулою (14)

$$P_{m,n} = \frac{\sum(\text{чисел комірки}(m, n))}{N} \quad (14)$$

**Висновки.** Наукова новизна отриманих результатів полягає у тому, що:

- 1) сформульовано фізичну постановку задачі відстеження погляду;
- 2) розроблено математичну модель задачі відстеження погляду людини;
- 3) здійснено програмну реалізацію такої математичної моделі;
- 4) розроблено методологію застосування технології відстеження погляду людини до оцінювання ефективності візуальної реклами. Достовірність отриманих результатів забезпечено строгою фізичною та математичною постановкою задачі та коректністю розглянутої у роботі математичної

моделі, застосуванням для їхнього розв'язання теоретично обґрунтованих алгоритмів та коректною програмною реалізацією.

**Практичне значення отриманих результатів.** Сформульовану математичну модель технології програмного відстеження погляду людини може бути використано для більш точного оцінювання порівняно із наявними методами ефективності візуальної реклами – і на етапі її створення, і під час безпосереднього її використання.

Запропонований у програмі метод програмного відстеження погляду людини за допомогою веб-камери, через свою простоту та дешевизну, може бути застосовуваним для цілого класу задач, що виходять за межі запропонованої роботи, зокрема для розроблення програмних комплексів, що призначені для людей з особливими потребами, для віддаленого керування складними технологічними процесами чи різноманітними пристроями тощо.

1. Березин И. С. Маркетинговый анализ. Рынок. Фирма. Товар. Продвижение / И. С. Березин. – [3-е изд., перераб. и доп.]. – М. : Вершина, 2007. – 480 с. : ил., табл. – ISBN 978-5-9626-0335-3.
2. Бернадская Ю. С. Основы рекламы : учебник [для студентов вузов] / Ю. С. Бернадская [и др.] ; под ред. Л. М. Дмитриевой. – М. : ЮНИТИ-ДАНА, 2007. – 351 с. – (Серия «Азбука рекламы»). – ISBN 978-5-238-01252-0.
3. Котлер Ф. Маркетинг, менеджмент / Ф. Котлер, К. Л. Келлер ; пер. с англ. С. Жильцов [и др.]. – [12-е изд.]. – СПб. : Питер, 2010. – 816 с. : ил. – (Серия «Классический зарубежный учебник»). – ISBN 978-5-469-00989-4.
4. Кутлалиев А. Эффективность рекламы / А. Кутлалиев, А. Попов. – М. : Эксмо, 2005. – 416 с. – (Профессиональные издания для бизнеса). – ISBN 5-699-10796-7.
5. Мудров А. К. Основы рекламы : учебник / А. Н. Мудров. – М. : Экономистъ, 2006. – 319 с. : ил. – (Номо faber). – ISBN 5-98118-112-5.
6. Подлесных В. И. Менеджмент: учебн. пособие [для вузов] / В. И. Подлесных [и др.] ; под ред. В. И. Подлесных. – СПб. : Бизнес-пресса, 2002. – 472 с. – ISBN 5-8110-0003-0.
7. Хиббинг-мл. Р. Настольная книга директора по маркетингу: маркетинговое планирование. Полное пошаговое руководство / Р. Хиббинг, С. Купер ; пер. с англ. Д. Куликова. – М. : Эксмо, 2007. – 832 с. – (Настольная книга специалиста). – ISBN 918-5-699-22478-4.
8. Glenstrup A. J. Eye-controlled media: present and future state / A. J. Glenstrup, T. Engell-Nielsen // University of Copenhagen DIKU, June, 1995.
9. Ярбус А. Л. Роль движений глаз в процессе зрения / А. Л. Ярбус. – М. : Наука, 1965. – 195 с.
10. Duchowski A. T. Eye Tracking Methodology: Theory and Practice. Springer, 2007. – 22 с.
11. Троелсен Э. С# и платформа .NET. Библиотека программиста / Э. Троелсен ; [пер. с англ.]. – СПб. : Apress, 2008. – 800 с.
12. Троелсен Э. С# и платформа .NET 3.0 / Э. Троелсен ; [пер. с англ.]. – СПб. : Apress, 2009. – 1456 с.
13. Нейгел К. С# 2005 и платформа .NET 3.0 для профессионалов / К. Нейгел, Б. Ивьен, Дж. Глинн, М. Скиннер, К. Уотсон ; [пер. с англ.]. – СПб. : Диалектика, 2007. – 1376 с.
14. Фленов М. Е. Библия С# / М. Е. Фленов. – [2-е изд., перераб. и допов.]. – СПб. : БХВ-Петербург, 2011. – 560 с.

Степанченко О. М., к.т.н., ст. викладач, Харів Н. О. ст. викладач,  
Личак Ю. В., студент 4 курсу ННІ АКОТ (Національний університет  
водного господарства та природокористування, м. Рівне)

## ОПТИМІЗАЦІЯ ГРАФІКИ ДЛЯ РОБОТИ У СЕРЕДОВИЩІ ADOBE FLASH

**Ключові слова:** оптимізація графіки, Adobe Flash, трасування, векторна графіка, растрова графіка.

**Вступ.** Під час створення додатків та банерів у середовищі Adobe Flash виникає низка проблем, пов'язаних із застосуванням графіки, а саме:

1. За неоптимізованої графіки збільшується час на перемальовування канви, що зменшує швидкість роботи програми, а також якість зображення [4].

2. Неоптимізована графіка займає більше пам'яті, ніж оптимізована, що збільшує розмір проекту, а також час початкового запуску додатка [3–6].

Для вирішення цих проблем необхідно максимально оптимізувати графіку.

Шляхом аналізу останніх публікацій можна виокремити декілька правил щодо оптимізації растрової і векторної графік у середовищі Adobe Flash.

**Правила щодо застосування растрової графіки.** Не варто імпортувати у проект растрові зображення плакатного розміру. Спочатку картинку необхідно зменшити у графічному редакторі до потрібного розміру. Якщо взяти картинку, що займає багато пам'яті, і масштабувати її за допомогою трансформації – флеш усе одно врахує її розмір і розмір проекту не зміниться [4].

Зменшити розмір растрових картинок можна й у середовищі Adobe Flash. Для цього необхідно відкрити у бібліотеці властивості растра; вибрати якість імпорту; обов'язково ввімкнути галочку Allow smoothing (Дозволити згладжування). Пам'ятаючи, що картинка з поганою якістю зменшує естетичну привабливість продукту, стискати растр потрібно в міру, щоб зберегти якість зображення [4].

**Правила щодо застосування векторної графіки.** Відомо, що розмір векторної графіки набагато менший за розмір растрової, але неоптимізована векторна графіка також істотно додає зайвих кілобайтів [4].

Гradient займає пам'яті більше, ніж однотонний колір [4]. Якщо немає великої необхідності у градієнтах, потрібно відмовитися від них. Також останнє твердження справедливе щодо фільтрів [3].

Використовуйте символи, щоб не створювати зайвої графіки, бо два однакові об'єкти займають більше, ніж два символи [4].

Якщо це можливо, потрібно відмовитись від ліній або перевести їх в області заливки [3]. Якщо це зробити не можна, треба намагатися зменшувати кількість різноманітних типів ліній (пунктир, штрихова тощо) [4]. Базова й оптимальна за займаною пам'яттю та відтворенням є суцільна лінія, тому бажано застосовувати її якомога частіше. Слід також відзначити доцільність використання замість ліній областей заливки.

**Постановка завдання.** Під час розроблення гри, розрахованої для роботи в браузері, необхідно дотримуватися двох принципів: гра має займати якомога менше місця та швидко працювати.

Прикладом може слугувати гра, яка знаходиться на стадії розроблення. На цей момент триває створення проекту стратегічної гри. Файл, у якому знаходяться графічні об'єкти, має великий об'єм порівняно з рештою файлів. Сумарний розмір проекту становить 602 КБ, а розмір власне файла з графікою – 532 КБ, що складає 88% проекту. Відтак, можна зробити висновок, що проблема оптимізації графіки проекту в такому разі є однією із найважливіших.

Зміст задачі полягає в тому, щоб шляхом зменшення розмірів зображення досягти кращої якості та вищої швидкості відмальовування зображення.

**Результат досліджень.** Виникає проблема: що робити, якщо є декілька растрових зображень або послідовність кадрів анімації, яку не можна змінювати, але в разі їхнього використання проект досягає розмірів, що перевищують задані вимоги.

Є два підходи до вирішення цієї проблеми.

Перший підхід простий у виконанні і полягає у зменшенні об'єму пам'яті, який займає картинка, шляхом зниження якості та розмірів зображення і використання певних утиліт. У такому разі значне зменшення розмірів зображення погано впливатиме на якість. Проте для зображень з дуже великою кількістю кольорових переходів такий метод буде ефективним.

Другий підхід полягає у переведенні растрового зображення у векторне (трасування) та його подальшій оптимізації. Перевагою цього шляху є зменшення зображення в декілька разів без втрати якості. Застосування цього підходу доцільне лише для зображень, що складаються з великих криволінійних або прямолінійних областей. У тому разі якщо зображення має багато малих криволінійних областей, потрібно обирати між якістю і розміром. Якщо сукупність зображень складає анімаційний ряд, то можна звести їх до одного символу, що дасть змогу зменшити розмір в декілька десятків разів. Здебільшого метод є оптимальним, але таким, що займає багато часу, коли анімація складається з великої кількості кадрів. Крім того, якщо в кадрах будуть наявні об'єкти, що не входять до початкової картини, яку оптимізують, їх доведеться теж оптимізувати на основі такого підходу.

З огляду на вищевикладене оптимальним є комбіноване використання обох підходів.

Перший підхід реалізують за допомогою графічних редакторів та утиліт для стиснення зображення. Він складається з трьох етапів.

Нехай маємо картинку, зображену на рис. 1. Її розмір – 26 КБ. Зображення має формат PNG.

Перший етап полягає у доборі оптимального формату для зображення.

За майже однакової якості розмір картинки, яку зображено на рис. 1, в різних форматах буде займати: у JPEG – 3.05 КБ (в разі 50%-го стискання), у GIF – 8.63 КБ (256 кольорів), у BMP – 38.48 КБ, у PNG – 26.15 КБ.

Оптимальним графічним форматом для такого зображення є JPEG, оскільки картинка буде займати менше пам'яті. Хоча, якщо зображення має прозорі та напівпрозорі області, застосування цього формату є недоцільним, і потрібно використовувати формат PNG або GIF залежно від потрібної кількості кольорів.

Другий етап полягає у маніпуляції властивостей зображення.

Уся оптимізація зображень у форматі JPEG – це добір оптимального коефіцієнта стиснення. Зображення у форматі JPEG за різних ступенів стискання – 100%, 50%, 0%, а розмір, відповідно, – 0.98 КБ, 2.3 КБ і 3.05 КБ. Проте чим більше ми стискаємо зображення, тим меншою стає його якість.

Унаслідок збереження зображення у форматі JPEG з коефіцієнта стиснення 80% було незначно втрачено якість, проте розмір зображення став 1.7 КБ.

Крім того, якщо в зображенні є області, що не потрапляють у видимість екрана, то їх бажано обрізати. Також можна зменшити кількість кольорів зображення, що дасть змогу зменшити кількість доступних кольорів, використаних у зображенні [7].

Після зменшення кількості кольорів розмір зображення зменшився до 1.3 КБ.

Третім етапом оптимізації є застосування утиліт, що видаляють непотрібні байти із зображення, а також проводять детальну оптимізацію [8].

У «Smush.it» [8] передбачено оперування методами оптимізації для конкретного формату зображення з метою видалення непотрібних байтів з файлів зображень. Вона є інструментом оптимізації «без втрат», що означає: дає можливість оптимізувати зображення без змін у зовнішньому вигляді або візуальній якості. «Smush.it» працює на веб-сторінці. У додатку вказано, скільки байт буде заощаджено за рахунок оптимізації зображення, і надано файл розширення Zip з обробленими файлами. Шляхом використання «Smush.it!» розмір зображення зменшено до 1.1 КБ.



Рис. 1.  
Зображення дюни

«RIOT»[8] – абрєвіатура від „інструмент радикальної оптимізації зображень”. Останнє є назвою оптимізатора картинок для користувачів Windows, який доступний як окремий додаток або як розширення IrfanView[8]. Додаток підтримує JPG, PNG і GIF-файли і здатний вирізати зображення метаданих для подальшого скорочення файлу. Також він має утиліти, що дозволяють здійснювати основні види редагування зображень, такі як і масштабування, поворот і відображення. Після використання «Riot» розмір зображення може бути зменшено до 0.99 КБ.

Другий підхід, реалізовуваний власне у середовищі Adobe Flash, також складається з трьох етапів.

Нехай маємо растрову картинку, зображену на рис. 2, розмір якої – 44 КБ, а формат – PNG. Оскільки вона є растровою, то не можна довільно змінювати розміри, повертати та нахилити зображення без втрати якості. Крім того, анімацію з таким зображенням можна зробити лише у разі створення послідовності кадрів з перемальованими зображеннями, що дасть змогу збільшити проект приблизно на  $44 \cdot n$  КБ, де  $n$  – кількість кадрів анімації.

Перший етап полягає у перетворенні растрової картинку на векторну.

Для зменшення розміру растрового зображення його необхідно трасувати. В Adobe Flash можна трасувати зображення вручну або програмно.

Для виконання програмного трасування необхідно виділити зображення та перейти на вкладку «Модифікувати», вибрати пункт «Растрове зображення» і підпункт «Трасувати». У вікні, що впливає, потрібно задати кольоровий (від 0 до 500) поріг та мінімальний розмір області (від 0 до 1000).

Перевагою цього підходу є те, що трасування можна виконати досить швидко, але із суттєвими недоліками, а саме: на зображенні все одно залишаються області з непотрібними перехідними кольорами, які потрібно редагувати вручну, і загалом якість зображення стає більш низькою. Крім того, очевидно є залежність: чим більше таких областей, тим вищою буде якість зображення, і навпаки, чим



Рис. 2.  
Зображення лицаря



Рис. 3.  
Картинка після програмного трасування

менше таких областей залишилось після трасування, тим якість буде нижчою.

Трасування вручну – це кропіткий процес. Його основним недоліком є те, що він займає досить багато часу. Проте до переваг належить: ненаявність непотрібних областей і за умови вмілого виконання не втрата якості.

Унаслідок застосування програмного трасування картинка матиме вигляд, зображений на рис. 3. Без значної втрати якості розмір зображення зменшено на 3 КБ, після вилучення залишкових областей – ще на 10 КБ, загалом розмір став 31 КБ. Такий великий розмір можна пояснити значною кількістю зайвих опорних точок.

При ручному трасуванні картинка матиме вигляд, зображений на рис. 4. Зображення стало займати 12 КБ, що доводить перевагу ручного трасування.

Однак у разі використання будь-якого з двох попередніх методів виграємо в тому, що можемо вільно трансформувати картинку без втрати якості.

Другим етапом оптимізації є розбиття зображення на окремі частини. У нас векторне зображення можна розбити на окремі частини (щит, меч, голова, тулуб, рука, нога), кожен з яких зберегти в окремі символи для подальшого використання. Деякі з частин можливо розбити на ще менші, оскільки їх можна складати із дзеркально відображених частин, що подано на рис. 5.

У результаті проведених маніпуляцій картинка стала придатною до анімації, буде швидше відмальовуватися, оскільки не є монолітною, а складається із частин (комп'ютеру не потрібно створювати 2 об'єкти одного типу). Крім того, розмір зображення зменшено до 7 КБ.

Перші два етапи оптимізації потрібно виконувати лише в разі растрового зображення. За умови створення графіки у середовищі Adobe Flash бажано відразу розбивати намальовані об'єкти на окремі символи. Крім того, перед їхнім створенням бажано скласти список частин, що будуть окремими символами. Проведення надто значного подібнення зображення є недоцільним, бо негативно позначиться на швидкості та розмірі зображення. Тому окремий символ повинна становити та частина картини, що бере участь в анімації або має розмір, що більший за 1/8 зображення.



Рис. 4. Картинка після ручного трасування



Рис. 5. Картинка, розбита на основні частини

Третій етап полягає у вилученні зайвих опорних вершин. Картинка після програмного трасування саме через велику кількість опорних вершин має більший розмір (12479 проти 2432 вершин в разі ручного трасування). На цьому етапі це можна виправити.

До оптимізації кількості вершин також розроблено 2 підходи, аналогічні до трасування.

У разі програмної оптимізації комп'ютер замінює декілька кривих на пряму лінію, за рахунок чого можна зменшити кількість опорних точок у декілька разів. Утім, застосування такого підходу негативно впливає на якість. Програмно задаємо кількість можливих прямолінійних відрізків у відсотках, що дає змогу розпочати процес оптимізації. Останній є ефективним за наявності дуже великого векторного зображення, в якому можна нехтувати деталями, або зображення, що складається зі значної кількості прямих ліній. Для малих криволінійних областей реалізація такого підходу є неефективною. Основна перевага, як і для програмного трасування, – це швидкість виконання.

Виконання оптимізації вручну забезпечує збереження і навіть покращення якості, а її детальне проведення зумовлює отримання меншої кількості вершин, ніж у разі програмної оптимізації. Основним недоліком залишається те, що це тривалий кропіткий процес.

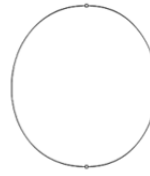
На рис. 6 зображено коло: на рис. 6 а) – неоптимізоване коло із 8 вершинами; на рис. 6 б) – програмно оптимізоване коло з відсотком прямолінійності, що дорівнює 100, та 4 вершинами (якість втрачено повністю); на рис. 6 в) – оптимізоване вручну коло, що має 2 вершини. Відтак, можемо зробити висновок про доцільність застосування для криволінійних областей ручної оптимізації опорних точок.



а)



б)



в)

Рис. 6.

Оптимізація кола

Таблиця

Назва \ Символ	Неоптимізована	Програмно оптимізована ВП 50%	Програмно оптимізована ВП 100%	Оптимізована вручну
Наплічник	46	36	35	18
Меч	113	46	45	27
Щит	133	65	60	50
Нога	444	180	180	95
Рука	265	117	115	88
Тулуб	170	77	73	46
Голова	656	294	287	171



Проведемо аналіз оптимізації картинки. В таблиці 1 наведено результати порівняння кількості опорних точок, що залишилися після певного підходу до оптимізації.

Для оптимізації було взято картинку, трасовану шляхом застосування ручного методу, оскільки оптимізація великої кількості вершин вручну – це досить тривалий процес. Для програмного трасування дані відрізнятимуться лише в перших 3 стовпцях таблиці (вершин буде в декілька разів більше), оскільки в разі оптимізації вручну після трасування за допомогою будь-якого з методів результат буде однаковим. Крім того, на основі даних рис. 7 можна порівняти якість кожного з методів оптимізації кількості опорних вершин: рис. 7 а) – це не оптимізовано; рис. 7 б) – програмно оптимізовано ВП 50%; рис. 7 в) – програмно оптимізовано ВП 100%; рис. 7 г) – оптимізовано вручну.

З огляду на вищевикладене можна констатувати про доцільність та ефективність проведення оптимізації вручну.

На третьому етапі оптимізації зменшено розмір зображення до 5 КБ, значно пришвидшено відмальовування зображення, а також досягнуто вихідної якості зображення, тобто як у картинки до оптимізації.

**Висновки.** Унаслідок проведеного дослідження обґрунтовано доцільність використання двох висвітлених вище підходів до оптимізації графіки проекту. На ґрунті одержаних результатів можна стверджувати, що правильний вибір методу оптимізації для конкретного графічного об'єкта забезпечить зменшення його розміру у першому випадку в 20 разів, а в другому випадку – в 10 разів, що має суттєвий вплив на розміри проекту.

1. Мук К. ActionScript 3.0 для Flash. Подробное руководство / К. Мук. – Спб. : Из-во «Питер», 2009. 2. Бхангал Ш. Flash. Трюки. 100 советов и рекомендаций профессионала / Ш. Бхангал. – Спб. : Из-во «Питер», 2005. 3. Оптимизация графики и кода // <http://flashzone.ru/s/flash-training/optimizatsija>. 4. Как уменьшить вес баннера? // <http://flash-animated.com/kak-umenshit-ves-bannera>. 5. 10 хитростей, о которых должен знать флеш-художник // <http://flash-animated.com/10-khitrastei-o-kotorykh-dolzhen-znat-flesh-khudozhnik>. 6. Передовой опыт – оптимизация FLA-файлов для создания SWF-объектов // [http://help.adobe.com/ru\\_RU/flash/cs/using/W5d60f23110762d6b883b18f10cb1fe1af6-7b23a.html](http://help.adobe.com/ru_RU/flash/cs/using/W5d60f23110762d6b883b18f10cb1fe1af6-7b23a.html). 7. Уменьшение количества цветов // <http://www.lessonsphotoshop.ru/photoshop/Phs454.html>. 8. 7+ превосходных инструментов для оптимизации изображений // <http://habrahabr.ru/post/79325/>.

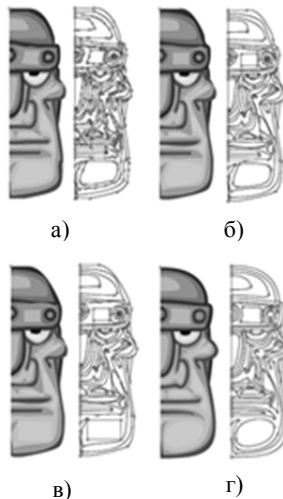


Рис. 7. Оптимізація фрагмента голови за допомогою різних методів

УДК 355/359:51

**Зубик Я. Я., ст. викладач, Чижевський В. С., студент 5 курсу ННІ АКОТ (Національний університет водного господарства та природокористування, м. Рівне)**

## **МЕТОДИ ПОШУКУ ОПТИМАЛЬНОГО МАРШРУТУ**

**Розглянуто вирішення проблеми побудови оптимального маршруту з використанням математичного апарату теорії графів та деяких алгоритмів.**

**Ключові слова: графова мережа, векторно-графовий метод, критерії відбору, маршрут, мурашиний алгоритм.**

**Рассмотрено решение проблемы построения оптимального маршрута с использованием математического аппарата теории графов и некоторых алгоритмов.**

**Ключевые слова: графовая сеть, векторно-графовый метод, критерии отбора, маршрут, муравьиный алгоритм.**

**The article present of the decision problem for determine optimal itinerary route on the base mathematical apparatus of the theory of a graph and some algorithms.**

**Key words: graph net, vector-graph method, selection criterion, itinerary route, ant colony optimization.**

Однією з найважливіших стадій побудови маршруту вважають визначення та вибір його оптимального пролягання, що здійснюють з обов'язковим урахуванням показників транспортної доступності. Основним методом обчислення транспортної доступності є графоаналітичний. Після побудови графа, що відображає конкретну транспортну мережу чи потенційну можливість (пішохідного) доступу до певного об'єкта, необхідно побудувати матрицю найкоротших відстаней (чи найменших затрат) між усіма вершинами цього графа. Однак це досить складно зробити для графа із значною кількістю вершин, бо в таких умовах важко здійснити вибір: який саме маршрут найбільш оптимальний.

Оптимальність тих чи інших алгоритмів оцінюють за різними критеріями. Найбільш поширені є такі:

1. Мінімальна кількість переходів – вартість використання маршруту вимірюють як кількість переходів (пройдені канали або вузли мережі) на найкоротшому маршруті. Відтак, вибирають шлях із найменшою кількістю можливих переходів.

2. Найкоротший шлях – кожне ребро статично характеризується певною вагою, тому вартість маршруту визначають як суму вагових коефіцієнтів кожного з ребер маршруту. Відповідно, вибирають шлях із найнижчою

можливою вартістю маршруту.

3. Мінімальна затримка (час) – кожне ребро динамічно характеризується ваговим коефіцієнтом залежно від трафіка, тому вибирають маршрут із мінімальною загальною затримкою.

З огляду на пріоритетність в багатьох випадках саме цього питання окреслюється необхідність розроблення конкретних методичних рекомендацій щодо вирішення означеної проблеми.

Під час розроблення схем маршрутів та їхньої оптимізації застосовують математичний апарат теорії графів, тобто графоаналітичні методи. Основне завдання при цьому полягає у побудові графа логістичної організації маршруту. Відомі також деякі алгоритми побудови оптимальних за різними критеріями маршрутів на графах. Але такі відомості може бути прийнято тільки за основу, оскільки і моделі, і відповідні алгоритми оптимізації передбачають рівноправність ребер графа, а для реальних ситуацій таке припущення є неприйнятним. Науковці наголошують, що використання теорії графів найбільш придатне для обчислення транспортної доступності.

Мета дослідження полягає у вирішенні проблеми визначення оптимального маршруту з урахуванням різноманітних критеріїв відбору та транспортної доступності. У роботі передбачено вирішення таких завдань:

- 1) огляд векторно-графового методу та точного алгоритму розв'язування для обчислення оптимального маршруту;
- 2) складення алгоритму вибору оптимального маршруту;
- 3) розгляд особливостей застосування задачі вибору оптимального маршруту.

Графова мережа становить графічну побудову, що складається з вершин, які сполучені між собою відрізками (графі). Кожен відрізок відображає зв'язок між вершинами, який у реальності представлений комунікаційними лініями (транспортними шляхами природного та антропогенного походження) чи територіями, що не є непрохідними для людини (враховуючи й транспортні засоби, крім повітряних).

Здебільшого відрізки графа мають лише якісний характер, тому для проектування маршруту необхідно їм присвоїти певні показники, за якими буде проведено відбір. Лише після визначення критеріїв відбору побудована графова мережа матиме практичне застосування.

Топологічна основа графа відображає порядок розміщення об'єктів і характер зв'язків між ними, тому вона залишається незмінною: як об'єкти розміщено на карті, так їх буде залишено у відповідно побудованій графівій мережі. Якщо відстань між об'єктами є обов'язковим показником, то в графівій мережі її має бути виражено в єдиному масштабі та відповідно до реальної відстані на місцевості. У всіх інших випадках першочергова та єдина вимога – це збереження топологічної основи. За умови дотримання цих вимог розпочинаємо побудову маршруту.

Нехай у нас є графова мережа  $G$ , яка охоплює певну кількість  $n$  об'єктів-

вершин ( $n \in G$ ), що сполучені зв'язками (визначеними векторами)  $S$ . Кожен зв'язок характеризується додатковими коефіцієнтами ( $a_1, a_2, \dots, a_n$ ), які необхідні для побудови та не змінюються в часі (точніше є квазістабільними, наприклад, умовно незмінювані протягом певного часу доби чи пори року)  $a = \text{const } (t)$ . Далі оберемо початок та кінець маршруту – точки  $K_n$  і  $K_k$  відповідно.

Якщо будуть замкнений маршрут, то визначають його серединну точку (щодо місця збору чи виїзду), яку умовно приймають за кінцеву. Тобто маршрут розділяють на дві частини: від місця виїзду до серединної точки і від серединної точки до місця виїзду.

Побудову замкненого маршруту, що проходить через усі вершини графа і який має мінімальну суму ваг ребер, названо „загальною задачею комівояжера” (ЗЗК). Маршрутом у графі названо послідовність вершин і ребер, які перетинаються, а ланцюгом – маршрут, всі ребра якого відрізняються. Простий ланцюг – це маршрут, що складається з різних вершин. Замкнений маршрут називають циклом, а цикл, в якому всі вершини різні, – простим. Простий цикл, який проходить через усі вершини точно один раз, називають гамільтоновим циклом, або обходом. ЗЗК, де потрібно знайти обхід із мінімальною сумою ваг ребер, відома як гамільтонова задача комівояжера (ГЗК). Не всякий граф містить обхід, тобто не кожен граф є гамільтоновим. Відтак, ГЗК, на відміну від ЗЗК, не завжди має розв'язок.

Розглянемо один із методів розв'язання, а саме векторно-графовий (критеріальний) метод для складних графових мереж. Будемо умовну прямокутну систему координат  $XU$ , у якій вісь  $x$  розміщена нижче від найнижчої точки графової мережі  $G$ . Це стосується й осі  $y$ . Граф  $G$  може охоплювати всі вершини-об'єкти чи бути частиною графової мережі, виділеної за регіональним принципом, тобто його можна вважати цілісним та єдиним. Від крайніх точок графа  $G$  через один топологічний відрізок (незалежно від масштабу) проводимо лінії перпендикулярно до осі  $x$ , що розбивають граф  $G$  на зони та які нумеруємо у напрямі простягання додатнього променя осі  $x$ . Усі точки графа  $G$  довільно нумеруємо та отримуємо відповідні значення координат у системі координат  $XU$  разом із прив'язуванням до відповідної зони. Далі визначаємо точки  $K_n$  (початкова точка) і  $K_k$  (кінцева точка) та виділяємо загальну тенденцію зміни координат  $x$  та  $y$  на прямій  $K_n K_k$ . Потім складаємо зведену таблицю.

На основі зведеної таблиці будемо матрицю порядку  $n \times n$ , де  $n$  – кількість усіх точок графа  $G$ , у чарунках якої відображено відповідність зміни координат у межах  $S_i(x; y)$  до тенденції зміни координат на прямій  $K_n K_k$ . Якщо  $x_1 \rightarrow x_2$  та  $y_1 \rightarrow y_2$  у  $S_i(x; y)$  так само, як  $x_{K_n} \rightarrow x_{K_k}$  та  $y_{K_n} \rightarrow y_{K_k}$ , то ставимо знак  $+$ , якщо ні, то знак  $-$ , якщо координати не змінюються, то ставимо знак  $0$ . Далі проводимо відсів точок. Серед точок, що залишилися, проведемо відбір, перевагу мають відрізки у такому порядку  $++>+0>+-00>-0>--$ .

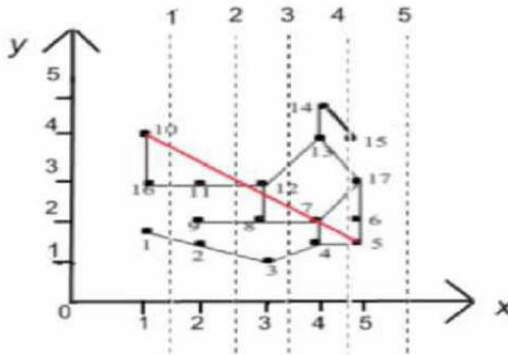


Рис. Приклад графової мережі

Іншим варіантом обчислення задачі комівояжера є точний алгоритм, опис якого приведено нижче.

Постановка задачі:  $H = (V, U)$  – зв’язний зважений граф із множиною вершин  $V$ ,  $|V| = n$ , та множиною ребер  $U$ ,  $[d_{ij}]_n$  – матриця ваг ребер графа  $H$ , де  $d_{ij} \in R_0^+$ , якщо  $\{i, j\} \in U$ , та  $d_{ij} = \infty$  інакше,  $i, j = \overline{1, n}$ ,  $R_0^+$  – множина дійсних невід’ємних чисел.

1) На основі алгоритму Флойда побудувати матрицю  $[a_{ij}]_n$  найкоротших ланцюгів між усіма парами вершин графа  $H$  і матрицю  $[D(a_{ij})]_n$ , в якій елемент  $(i, j)$  дорівнює вазі  $D(a_{ij})$  ланцюга  $a_{ij}$ ; матриці  $[a_{ij}]_n$  та  $[D(a_{ij})]_n$  визначають повний зважений граф  $H_a = (V, E)$ , де кожне ребро  $\{i, j\}$  замінює ланцюг  $a_{ij}$  у графі  $H$ .

2) У графі  $H_a$  знайти обхід мінімальної вартості  $\sigma$  за допомогою будь-якого відомого методу розв’язання метричної ЗК.

3) Побудувати оптимальний розв’язок ЗК у результаті заміни кожного ребра  $\{i, j\}$  гамільтонового циклу  $\sigma$  на ланцюг  $a_{ij}$  графа  $H$ .

Загалом зв’язний граф  $H$  охоплює декілька замкнених маршрутів мінімальної вартості, які містять різну кількість ребер. Представлений алгоритм дає змогу будувати будь-який з них.

Розроблено ще один з ефективних поліноміальних алгоритмів для знаходження наближених рішень задачі комівояжера, а також аналогічних завдань пошуку маршрутів на графах. Підхід запропонований бельгійським дослідником Марко Доріго (англ. *Marco Dorigo*). Суть підходу полягає в аналізі та використанні моделі поведінки мурах, що шукають дорогу від колонії до їжі. Нижче наведено базовий мурашиний алгоритм оптимізації маршруту комівояжера.

Ініціалізація параметрів алгоритму  $\alpha, \beta, \epsilon, Q, \tau_0$ .

$m = n$  {Кількість мурах дорівнює кількості місць}

**For**  $i = 1$  to  $n$

```

For  $j = 1$  to  $n$            {Для кожного ребра}
    If  $i <> j$ 
         $\eta(i, j) = 1 / D(i, j)$    {Видимість}
         $\tau(i, j) = \tau_0$  {Феромон}
        Else  $t(i, j) = 0$  End

```

**End**

**End**

**End**

**For**  $k = 1$  to  $m$

Розмістити мурашу  $k$  у випадково обране місце.

**End**

Умовно обрати найкоротший маршрут  $T^+$  та обчислити його довжину  $L^+$ .

{Головна програма}

**For**  $t = 1$  to  $t_{max}$  {Кількість ітерацій алгоритму}

**For**  $k = 1$  to  $m$  {Для кожної мурахи}

    Побудувати маршрут  $T^k(t)$  за правилом (1) та обчислити його довжину  $L^k(t)$ .

**End**

**If** «Кращий розв'язок знайдено»

    Оновити  $T^+$  та  $L^+$ .

**End**

**For**  $i = 1$  to  $n$

**For**  $j = 1$  to  $n$  {Для кожного ребра}

        Оновити сліди феромону за правилом (2).

**End**

**End**

**End**

Вивести найкоротший маршрут  $T^+$  та його довжину  $L^+$ . Наведений вище мурашиний алгоритм було протестовано для задач невеликої розмірності. Цей алгоритм швидко знаходить оптимальний розв'язок.

$\alpha$  і  $\beta_0$  – два регульовані параметри, які є вагами інтенсивності сліду феромону та видимості;  $T^k(t)$  – маршрут, прокладений мурахою  $k$  на ітерації  $t$ ;  $L^k(t)$  – його довжина;  $Q$  – регульований параметр, значення якого обирають одного порядку з довжиною оптимального маршруту. Кількість феромону, що відкладається на ребрах маршруту  $T^+$ , приймається такою, що дорівнює  $Q/L^+$ , де  $L^+$  – довжина маршруту  $T^+$ . Підсилений слід феромону вздовж маршруту  $T^+$  буде спрямовувати інших мурах до пошуку розв'язків, що містять декілька ребер найкращого на цей момент маршруту  $T^+$ .

Використання векторно-графового методу полягає у застосуванні апарату теорії графів із присвоєнням графам певних значень критеріїв відбору відповідно до потреб дослідника (користувача). Виділено окремі модифікації цього методу для деяких графових мереж та показано приклад графової

мережі. Коли графова мережа не розвинена, тобто середній степінь вершини  $p_{(x)} = 3$ , краще послуговуватись полярною системою координат, а відбір здійснювати за значенням проекції радіус-вектора на прямокутну систему координат та за величиною полярного кута. Для складних графових мереж потрібно використовувати прямокутну систему координат.

Оперування мурашиними алгоритмами дає змогу отримати розв'язки багатьох дискретних комбінаторних задач не гірше, ніж на основі загальних метаевристичних технологій оптимізації. Надзвичайно високі результати мурашиної оптимізації для розподілених систем, параметри яких зазнають змін у часі. Перспективними шляхами покращення мурашиних алгоритмів, на нашу думку, є їхня гібридизація з іншими методами природних обчислень, наприклад, з генетичними алгоритмами. Таку гібридизацію може бути реалізовано за острівною схемою, коли мурашиний та генетичний алгоритми працюють паралельно – кожен на своєму острові, з обміном найкращими розв'язками через певний час.

1. Резер С. М. Математические методы оптимального планирования в транспортных системах / С. М. Резер, С. Е. Ловецкий, И. И. Меламед. – М. : Итоги науки и техники, серия «Организация управления транспортом», 1990. – 171 с.
2. Панишев А. В. О решении задачи коммивояжера на транспортной сети / А. В. Панишев, И. В. Гаращенко, В. А. Шевченко // Глобальные информационные системы. Проблемы и тенденции развития : материалы первой Междунар. науч. конф. (3–6 окт. 2006 г.). – Харьков-Туапсе : ХНУРЭ, 2006. – С. 244–245.
3. Нечипуренко М. И. Алгоритмы и программы решения задач на графах и сетях / М. И. Нечипуренко, В. К. Попков, С. М. Маймагалиев и др. – Новосибирск : Наука, Сиб. отд-ние, 1990. – 515 с.
4. Бейдик О. О. Методологія та методика аналізу рекреаційно-туристичних ресурсів України : дис. ... д-ра геогр. наук : 11.00.02 / О. О. Бейдик. – К., 2005. – 437 с.
5. Бугроменко В. Н. Транспорт в территориальных системах / В. Н. Бугроменко. – М. : Наука, 1987. – 112 с.
6. Харари Ф. Теория графов / Харари Ф. – М. : Мир, 1973. – 300 с.
7. Каючкин Н. П. Географические основы транспортного освоения территории / Н. П. Каючкин. – Новосибирск : Наука, 2003. – 166 с.
8. Тархов С. А. Эволюционная морфология транспортных сетей / С. А. Тархов. – Смоленск ; М. : Универсум, 2005. – 382 с.
9. Меламед И. И. Задача коммивояжера. Точные алгоритмы / И. И. Меламед, С. И. Сергеев, И. Х. Сигал // Автоматика и телемеханика. – 1989. – № 10. – С. 3–29.
10. Dorigo M., Maniezzo V., Colomi A. The Ant System: Optimization by a Colony of Cooperating Agents. Part B. – 1996. – № 1. –Vol. 26.

## ЗМІСТ

Цветкова Т. П., Стрілець Н. І.	Математичне моделювання масоперенесення солей при вологоперенесенні у шаруватих ґрунтах у нелінійному випадку .....3
Цветкова Т. П., Волянік Г. В.	Математичне моделювання солеперенесення при вологоперенесенні у шаруватих ґрунтах у лінійній постановці .....10
Федорчук Н. А., Бормат Д. М.	Математичне моделювання напружено-деформованого стану багат шарового ґрунтового масиву з рухомою вільною поверхнею під впливом тепло-масоперенесення.....17
Демчук О. С., Харів Н. О., Грицюк О. В.	Оцінка екологічного стану басейну річки Жовта за допомогою інформаційно-аналітичної системи .....23
Остапчук О. П., Мартинчук А. І.	Математичне моделювання перенесення сольових розчинів при фільтрації підземних вод через земляну дамбу з вільною поверхнею.....29
Білецький А. І., Чуй Ю. В.	Застосування алгебр С. Лі для деяких класів диференціальних рівнянь.....36
Прищепка О. В., Бабиць Я. А.	Гістерезисні стратегії керування для одноканальних систем з однією спробою повтору .....39
Прищепка О. В., Бурмус М. В.	Оптимізація роботи багатоканальних систем масового обслуговування з однією спробою повтору .....49
Бачишина Л. Д., Мукоєд О. В.	Розробка web-ресурсу „Фотостудія-школа” .....59
Остапчук О. П., Габ Р. І.	Математичне моделювання міграції радіонуклідів до дренажних вловлювачів у разі плоско-вертикальної напірної фільтрації.....67
Громадченко Т. В., Мартинюк П. М., Ткачук С. А.	Визначення критичної поверхні ковзання в ґрунтовому схилі за допомогою методів оптимізації .....76



Брушковський О. Л., Дубчак І. В.	До визначення полів напружень і деформацій у елементах силових конструкцій на основі теоретико- експериментального методу .....85
Власюк А. П., Гульчук В. А.	Побудова монотонних різницевих схем для нелінійного багатовимірного рівняння параболічного типу та нелінійних систем рівнянь параболічного типу .....92
Степанченко О. М., Мартинюк А. О.	Розробка програмного комплексу для пошуку оптимальної позиції та маневра військ .....100
Степанченко О. М., Карачанський А. О.	Розробка графічного редактора для вирішення тактичних завдань.....106
Турбал Ю. В., Радовенюк О. В., Турбал М. Ю.	Граничні властивості орбіт дискретних динамічних систем, що визначаються класом квадратичних операторів.....112
Гладун Л. В.	Оптимізаційний аналіз прямих методів розв'язання систем лінійних алгебраїчних рівнянь.....118
Остапчук О. П., Гриб О. П.	Числове дослідження одновимірної задачі масоперенесення сольових розчинів за нелінійної абсорбції .....125
Герус В. А., Сідлярук І.	Дослідження роботи кліткових автоматів .....132
Степанченко О. М., Харів Н. О., Чугай М. І.	Розробка програмного комплексу для оцінювання ефективності розміщення контенту на web-сайтах .....140
Степанченко О. М., Харів Н. О., Личак Ю. В.	Оптимізація графіки для роботи у середовищі ADOBE FLASH.....147
Зубик Я. Я., Чижевський В. С.	Методи пошуку оптимального маршруту .....154

Наукове видання

**ВІСНИК**  
**НАВЧАЛЬНО-НАУКОВОГО ІНСТИТУТУ АВТОМАТИКИ,**  
**КІБЕРНЕТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ**  
**НУВГП**

**Випуск 1**

*Технічний редактор*

*Харів Н. О.*

*Друкується в авторській редакції*

Підписано до друку 29.08.2014 р. Формат 60x84 1/16.  
Папір друкарський № 1. Гарнітура Times. Друк різнографічний.  
Ум.-друк. арк. 9,4. Обл.-вид. арк. 9,8.  
Тираж 150 прим. Зам. № 4036.

*Видавець і виготовлювач*  
*Редакційно-видавничий відділ*  
*Національного університету*  
*водного господарства та природокористування*  
*33028, Рівне, вул. Соборна, 11.*

*Свідоцтво про внесення суб'єкта видавничої справи до державного*  
*реєстру видавців, виготівників і розповсюджувачів видавничої*  
*продукції РВ № 31 від 26.04.2005 р.*