

4. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ГАЛУЗІ

УДК 004.[272.2+942]

АЛГОРИТМІЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ДОСЛІДЖЕННЯ НЕІЗОТЕРМІЧНОГО ВОЛОГОПЕРЕНОСЕННЯ У СЕРЕДОВИЩАХ ІЗ ФРАКТАЛЬНОЮ СТРУКТУРОЮ

Я.І. Соколовський¹, В.І. Ярқун², М.В. Левкович³

Сформовано вимоги і вибрано інтегрований підхід до створення алгоритмічного забезпечення чисельного моделювання неізотермічного вологоперенесення у середовищах із фрактальною структурою, який охоплює розроблення паралельного алгоритму розв'язання систем диференціальних рівнянь у частинних похідних. Запропоновано синхронізацію взаємодії потоків для паралельних обмінів між часовими шарами. Розроблено програмне забезпечення та створено інтерфейс користувача для візуалізації результатів дослідження.

Ключові слова: паралельні обчислення, потоки, диференціальні рівняння дробового порядку, вологоперенесення.

Вступ. Важливим є дослідження одновимірних та двовимірних математичних моделей теплообмінних процесів, для опису яких використовували диференціальні рівняння у частинних похідних із дробовим порядком за часом та просторовими координатами. Пошук ефективного рішення щодо розроблення програмних засобів для реалізації таких математичних моделей зумовлює впровадження паралельних обчислень. Паралельне програмування дає змогу використовувати для обчислень не один процесор, а кілька доступних процесорів для того, щоб пришвидшити процес обчислення. Отже, використання паралельних обчислень дає змогу розглядати складніші математичні моделі дослідження неізотермічного волого-перенесення у середовищах із фрактальною структурою.

Аналіз сучасного стану досліджень. Сучасні дослідження багатьох природних процесів у всіх галузях науки – фізиці, біофізиці, механіці, економіці, медицині, гірничій справі – дедалі частіше зводиться до обчислення складних задач та створення програмного та алгоритмічного забезпечення, реалізація якого потребує значних обчислювальних ресурсів. Опис алгоритмічного забезпечення процесів, що описуються системами диференціальних рівнянь у частинних похідних дробового порядку висвітлено у працях [1, 2, 5, 9], а з використанням розпаралелювання процесу – у [5, 7, 8].

Останнім часом багато досліджень проводять з використанням кластерних систем. У праці [7] розглянуто швидкий алгоритм вирішення систем рівнянь агрегаційно-фрагментаційної кінетики, запропоновано паралельну реалізацію алгоритму, що ґрунтується на використанні паралельних алгоритмів побудови малорангових матричних апроксимацій, паралельних алгоритмів множення малорангових матриць на вектор. Варто звернути увагу, що незначна кількість задач

реалізується багатопотоковим розпаралеленням. У праці [8] розглянуто багатопотокову реалізацію мовою Java чотирьохточкового блокового однокрокового методу інтегрування звичайних диференціальних рівнянь, де запропоновано алгоритм реалізації методу за допомогою чотирьох потоків. Як механізм обміну даними між потоками використано блокувальні черги класу `LinkedBlockingQueue`. Наведено приклади задач для оцінювання правильності і точності методу. Нові паралельні алгоритми дають змогу пришвидшити розрахунок у десятки разів і перейти до дослідження складних систем, для дослідження яких потрібно розглядати системи із значною кількістю диференціальних рівнянь [7-8]. У багатьох дослідженнях обговорюють умови, за яких багатопотокова реалізація цього методу стає ефективною.

Постановка задачі та її реалізація. Математична модель процесу тепло-воловоперенесення у середовищах із фрактальною структурою описується системою диференціальних рівнянь у частинних похідних із дробовим порядком за часом τ та просторовими координатами x_1 та x_2

$$c\rho \frac{\partial^\alpha T(\tau, x_1, x_2)}{\partial \tau^\alpha} = \lambda_1 \frac{\partial^\beta T(\tau, x_1, x_2)}{\partial x_1^\beta} + \lambda_2 \frac{\partial^\beta T(\tau, x_1, x_2)}{\partial x_2^\beta} + \varepsilon \rho_0 r \frac{\partial^\alpha U(\tau, x_1, x_2)}{\partial \tau^\alpha}, \quad (1)$$

$$\frac{\partial^\alpha U(\tau, x_1, x_2)}{\partial \tau^\alpha} = a_1 \frac{\partial^\beta U(\tau, x_1, x_2)}{\partial x_1^\beta} + a_2 \frac{\partial^\beta U(\tau, x_1, x_2)}{\partial x_2^\beta} + a_1 \delta \frac{\partial^\beta T(\tau, x_1, x_2)}{\partial x_1^\beta} + a_2 \delta \frac{\partial^\beta T(\tau, x_1, x_2)}{\partial x_2^\beta}, \quad (2)$$

з початковими умовами:

$$T|_{\tau=0} = T_0(x_1, x_2), \quad (3)$$

$$U|_{\tau=0} = U_0(x_1, x_2) \quad (4)$$

та граничними умовами третього роду:

$$\lambda_i \left. \frac{\partial^\beta T}{\partial x_i^\beta} \right|_{x_i=l_i} + \rho_0(1-\varepsilon)\beta_i (U|_{x_i=l_i} - U_{pi}) = \alpha_i (T|_{x_i=l_i} - t_{ci}), \quad (5)$$

$$a_i \delta \left. \frac{\partial^\beta T}{\partial x_i^\beta} \right|_{x_i=l_i} + a_i \left. \frac{\partial^\beta U}{\partial x_i^\beta} \right|_{x_i=l_i} = \beta_i (U_{pi} - U|_{x_i=l_i}), \quad (6)$$

$$\lambda_i \left. \frac{\partial^\beta T}{\partial x_i^\beta} \right|_{x_i=0} + \rho_0(1-\varepsilon)\beta_i^* (U|_{x_i=0} - U_{pi}^*) = \alpha_i^* (T|_{x_i=0} - t_{ci}^*), \quad (7)$$

$$a_i \delta \left. \frac{\partial^\beta T}{\partial x_i^\beta} \right|_{x_i=0} + a_i \left. \frac{\partial^\beta U}{\partial x_i^\beta} \right|_{x_i=0} = \beta_i^* (U_{pi}^* - U|_{x_i=0}), \quad (8)$$

де: $(\tau, x_1, x_2) \in D, D = [0, t] \times [0, l_1] \times [0, l_2]$; T, U – шукані функції, де T – температура, U – вологовміст; c – питома теплоємність; ρ – густина; ρ_0 – базисна густина; ε – коефіцієнт фазового переходу; r – питома теплота пароутворення; $\lambda_i (i=1,2)$ – коефіцієнти теплопровідності; $a_i (i=1,2)$ – коефіцієнти вологопровідності; δ – термоградієнтний коефіцієнт; $t_{ci}, t_{ci}^* (i=1,2)$ – значення температури середовища; $U_{pi}, U_{pi}^* (i=1,2)$ – значення відносної вологості зовнішнього середовища; $\alpha_i, \alpha_i^* (i=1,2)$ – коефіцієнти теплообміну; $\beta_i, \beta_i^* (i=1,2)$ – коефіцієнти

¹ проф. Я.І. Соколовський, д-р техн. наук – НЛТУ України, м. Львів;

² асист. В.І. Ярқун – НЛТУ України, м. Львів;

³ аспір. М.В. Левкович – НЛТУ України, м. Львів

вологообміну; l_i ($i=1,2$) – геометричні розміри; α – дробовий порядок похідної за часом ($0 < \alpha \leq 1$); β, γ – дробові показники похідної за просторовими координатами ($1 < \beta \leq 2$), ($0 < \gamma \leq 1$).

Чисельний метод розв'язання задачі. В області D введемо просторово-часову сітку:

$$\begin{aligned} \mathcal{D}_{\Delta\tau, h_1, h_2} &= \{(\tau^k, x_{1(n)}, x_{2(m)}) : x_{1(n)} = (n-1)h_1, x_{2(m)} = \\ &= (m-1)h_2, \tau^k = k\Delta\tau, n = 1, \dots, N; h_1 = \frac{l_1}{N-1}; m = 1, \dots, M; \\ h_2 &= \frac{l_2}{M-1}; k = 0, 1, \dots, K; \Delta\tau = \frac{t}{K}\}. \end{aligned}$$

Використавши формулу Рімана-Ліувілля [2]

$$\left. \frac{\partial^\alpha f(\tau)}{\partial \tau^\alpha} \right|_{\tau^k} = \frac{1}{\Gamma(1-\alpha)} \left(\frac{f(\tau^k)}{(\tau^{k+1} - \tau^k)^\alpha} + \int_{\tau^k}^{\tau^{k+1}} \frac{f'(\xi)}{(\tau^{k+1} - \xi)^\alpha} d\xi \right)$$

різницеву апроксимацію дробової похідної α ($0 < \alpha \leq 1$) на відрізку $[\tau^k, \tau^{k+1}]$ можна записати так:

$$\left. \frac{\partial^\alpha u}{\partial \tau^\alpha} \right|_{\tau^k} \approx \frac{u^{k+1} - \alpha u^k}{\Gamma(2-\alpha)\Delta\tau^\alpha}, \quad \Delta\tau = \tau^{k+1} - \tau^k, \quad (9)$$

де $\Gamma(\alpha)$ – гамма-функція.

Для визначення дробової похідної β ($1 < \beta \leq 2$) за координатами x_1, x_2 скористаємось формулою Грюнвальда-Летнікова [3]

$$\frac{\partial^\beta f(\tau)}{\partial x_i^\beta} = \lim_{h_i \rightarrow 0} \frac{1}{h_i^\beta} \sum_{j=0}^{\lfloor \frac{\tau}{h_i} \rfloor} (-1)^j \frac{\Gamma(\beta+1)}{\Gamma(j+1)\Gamma(\beta-j+1)} f(x_i - jh_i),$$

де: $h_i = x_{in+1} - x_{in}$; $\lfloor x_i \rfloor$ – ціла частина x_i ; $i = 1, 2$.

Тоді різницева апроксимація дробової похідної β за координатами x_1 і x_2 матиме такий вигляд:

$$\left. \frac{\partial^\beta u}{\partial x_1^\beta} \right|_{x_{1(n)}} \approx \frac{1}{h_1^\beta} \sum_{j=0}^n q_j u_{n-j+1}, \quad \left. \frac{\partial^\beta u}{\partial x_2^\beta} \right|_{x_{2(m)}} \approx \frac{1}{h_2^\beta} \sum_{j=0}^m q_j u_{m-j+1}, \quad (10)$$

де $q_0 = 1, q_j = \frac{(-1)^j \beta(\beta-1)\dots(\beta-j+1)}{j!}$.

Враховуючи (9), (10), отримаємо неявну різницеву схему для чисельної реалізації системи диференціальних рівнянь (1), (2):

$$c\rho \frac{T_{n,m}^{k+1} - \alpha T_{n,m}^k}{\Gamma(2-\alpha)\Delta\tau^\alpha} = \frac{\lambda_1}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^{k+1} + \frac{\lambda_2}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^{k+1} + \varepsilon\rho_0 r \frac{U_{n,m}^{k+1} - \alpha U_{n,m}^k}{\Gamma(2-\alpha)\Delta\tau^\alpha}, \quad (11)$$

$$\frac{U_{n,m}^{k+1} - \alpha U_{n,m}^k}{\Gamma(2-\alpha)\Delta\tau^\alpha} = \frac{a_1}{h_1^\beta} \sum_{j=0}^n q_j U_{n-j+1,m}^{k+1} + \frac{a_2}{h_2^\beta} \sum_{j=0}^m q_j U_{n,m-j+1}^{k+1} + \frac{a_1\delta}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^{k+1} + \frac{a_2\delta}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^{k+1}. \quad (12)$$

Явна різницева схема буде така:

$$c\rho \frac{T_{n,m}^{k+1} - \alpha T_{n,m}^k}{\Gamma(2-\alpha)\Delta\tau^\alpha} = \frac{\lambda_1}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^k + \frac{\lambda_2}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^k + \varepsilon\rho_0 r \frac{U_{n,m}^{k+1} - \alpha U_{n,m}^k}{\Gamma(2-\alpha)\Delta\tau^\alpha}, \quad (13)$$

$$\frac{U_{n,m}^{k+1} - \alpha U_{n,m}^k}{\Gamma(2-\alpha)\Delta\tau^\alpha} = \frac{a_1}{h_1^\beta} \sum_{j=0}^n q_j U_{n-j+1,m}^k + \frac{a_2}{h_2^\beta} \sum_{j=0}^m q_j U_{n,m-j+1}^k + \frac{a_1\delta}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^k + \frac{a_2\delta}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^k. \quad (14)$$

Початкові умови:

$$T_{n,m}^0 = T_0(x_{1(n)}, x_{2(m)}), \quad U_{n,m}^0 = U_0(x_{1(n)}, x_{2(m)}). \quad (15)$$

Скінченно-різницевий аналог граничних умов має вигляд:

$$\begin{aligned} \lambda_i \frac{T_R^{k+1} - \gamma T_{R-1}^{k+1}}{\Gamma(2-\gamma)h_i^\gamma} + \rho_0(1-\varepsilon)\beta_i(U_R^{k+1} - U_{pi}) &= \alpha_i(T_R^{k+1} - t_{ci}), \\ a_i\delta \frac{T_R^{k+1} - \gamma T_{R-1}^{k+1}}{\Gamma(2-\gamma)h_i^\gamma} + a_i \frac{U_R^{k+1} - \gamma U_{R-1}^{k+1}}{\Gamma(2-\gamma)h_i^\gamma} &= \beta_i(U_{pi} - U_R^{k+1}), \\ \lambda_i \frac{T_2^{k+1} - \gamma T_1^{k+1}}{\Gamma(2-\gamma)h_i^\gamma} + \rho_0(1-\varepsilon)\beta_i^*(U_1^{k+1} - U_{pi}^*) &= \alpha_i^*(T_1^{k+1} - t_{ci}^*), \\ a_i\delta \frac{T_2^{k+1} - \gamma T_1^{k+1}}{\Gamma(2-\gamma)h_i^\gamma} + a_i \frac{U_2^{k+1} - \gamma U_1^{k+1}}{\Gamma(2-\gamma)h_i^\gamma} &= \beta_i^*(U_{pi}^* - U_1^{k+1}), \end{aligned} \quad (16)$$

де $i = 1, 2$ (якщо $i = 1$, то $R = N$ та якщо $i = 2$, то $R = M$).

Використаємо метод предиктор-коректор [13] для знаходження чисельних розв'язків отриманої системи різницевих рівнянь. У ролі предиктора використаємо неявну різницеву схему (11), (12), а у ролі коректора – явну різницеву схему (13), (14).

На першому півкроці інтервалу $\Delta\tau/2$ запишемо неявну різницеву схему, у якій врахуємо тільки похідну дробового порядку β за координатою x_1 :

$$c\rho \frac{T_{n,m}^{k+1/4} - \alpha T_{n,m}^k}{\Gamma(2-\alpha)(\Delta\tau/2)^\alpha} = \frac{\lambda_1}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^{k+1/4} + \varepsilon\rho_0 r \frac{U_{n,m}^{k+1/4} - \alpha U_{n,m}^k}{\Gamma(2-\alpha)(\Delta\tau/2)^\alpha}, \quad (17)$$

$$\frac{U_{n,m}^{k+1/4} - \alpha U_{n,m}^k}{\Gamma(2-\alpha)(\Delta\tau/2)^\alpha} = \frac{a_1}{h_1^\beta} \sum_{j=0}^n q_j U_{n-j+1,m}^{k+1/4} + \frac{a_1\delta}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^{k+1/4}. \quad (18)$$

Рівняння (17) для зручності перепишемо у вигляді:

$$U_{n,m}^{k+1/4} = -A_1 q_0 T_{n+1,m}^{k+1/4} + \left(\frac{c\rho}{\varepsilon\rho_0 r} - A_1 q_1 \right) T_{n,m}^{k+1/4} - A_1 \sum_{j=2}^n q_j T_{n-j+1,m}^{k+1/4} + \alpha \left(U_{n,m}^k - \frac{c\rho}{\varepsilon\rho_0 r} T_{n,m}^k \right), \quad (19)$$

де $A_1 = \frac{\lambda_1 \Gamma(2-\alpha)(\Delta\tau/2)^\alpha}{\varepsilon\rho_0 r h_1^\beta}$.

Граничні умови (16) відносно координати x_1 , що відповідають рівнянню (17), запишемо у такому вигляді:

$$U_{N,m}^{k+1/4} = \left(\frac{\alpha_1}{\rho_0(1-\varepsilon)\beta_1} - B_1 \right) T_{N,m}^{k+1/4} + B_1 \gamma T_{N-1,m}^{k+1/4} + U_{pl} - \frac{\alpha_1 t_{cl}}{\rho_0(1-\varepsilon)\beta_1}, \quad (20)$$

$$U_{1,m}^{k+1/4} = \left(\frac{\alpha_1^*}{\rho_0(1-\varepsilon)\beta_1^*} + B_1^* \gamma \right) T_{1,m}^{k+1/4} - B_1^* T_{2,m}^{k+1/4} + U_{pl}^* - \frac{\alpha_1^* t_{cl}^*}{\rho_0(1-\varepsilon)\beta_1^*}, \quad (21)$$

$$\text{де: } B_1 = \frac{\lambda_1}{\rho_0(1-\varepsilon)\beta_1\Gamma(2-\gamma)h_1^\gamma}, \quad B_1^* = \frac{\lambda_1}{\rho_0(1-\varepsilon)\beta_1^*\Gamma(2-\gamma)h_1^\gamma}.$$

Запишемо в матричному вигляді рівняння (19)-(21):

$$U_n^{k+1/4} = A T_n^{k+1/4} + \alpha U_n^k - \frac{\alpha c \rho}{\varepsilon \rho_0 r} T_n^k + \Psi_1, \quad (22)$$

$$\text{де: } U_n^{k+1/4} = \left[U_{1,m}^{k+1/4}, U_{2,m}^{k+1/4}, \dots, U_{N-1,m}^{k+1/4}, U_{N,m}^{k+1/4} \right]^T, \quad T_n^{k+1/4} = \left[T_{1,m}^{k+1/4}, T_{2,m}^{k+1/4}, \dots, T_{N-1,m}^{k+1/4}, T_{N,m}^{k+1/4} \right]^T,$$

$$U_n^k = \left[0, U_{2,m}^k, \dots, U_{N-1,m}^k, 0 \right]^T, \quad T_n^k = \left[0, T_{2,m}^k, \dots, T_{N-1,m}^k, 0 \right]^T,$$

$$\Psi_1 = \left[U_{pl}^* - \frac{\alpha_1^* t_{cl}^*}{\rho_0(1-\varepsilon)\beta_1^*}, 0, \dots, 0, U_{pl} - \frac{\alpha_1 t_{cl}}{\rho_0(1-\varepsilon)\beta_1} \right]^T.$$

Компоненти a_{ij} , $i, j = \overline{1, N}$, матриці A визначаються за виразами:

$$a_{ij} = \begin{cases} 0, & j \geq i + 2; \\ \frac{\alpha_1}{\rho_0(1-\varepsilon)\beta_1} - B_1, & i = j = N; \\ 0, & i = N, 1 \leq j \leq N - 2; \\ -B_1^*, & i = 1, j = 2; \\ \frac{c\rho}{\varepsilon\rho_0 r} - A_1 q_1, & i = j \neq 1 \neq N; \\ B_1 \gamma, & i = N, j = N - 1; \\ \frac{\alpha_1^*}{\rho_0(1-\varepsilon)\beta_1^*} + B_1^* \gamma, & i = j = 1; \\ -A_1 q_{i-j+1}, & \text{інше.} \end{cases}$$

Аналогічно запишемо у матричному вигляді рівняння (18) та граничні умови (16), що відповідають йому:

$$B T_n^{k+1/4} + C U_n^{k+1/4} + \Psi_2 + \alpha U_n^k = 0, \quad C = (c_{ij}), \quad B = (b_{ij}), \quad i, j = \overline{1, N}, \quad (23)$$

$$c_{ij} = \begin{cases} 0, & j \geq i + 2; \\ 0, & i = N, 1 \leq j \leq N - 2; \\ (Z q_1 - 1), & i = j \neq 1 \neq N; \\ (a_1 \gamma - \beta_1^* \Gamma(2-\gamma) h_1^\gamma), & i = j = 1; \\ -(a_1 + \beta_1 \Gamma(2-\gamma) h_1^\gamma), & i = j = N; \\ -a_1, & i = 1, j = 2; \\ a_1 \gamma, & i = N, j = N - 1; \\ Z q_{i-j+1}, & \text{інше.} \end{cases} \quad b_{ij} = \begin{cases} 0, & j \geq i + 2; \\ 0, & i = N, 1 \leq j \leq N - 2; \\ Z q_1 \delta, & i = j \neq 1 \neq N; \\ a_1 \delta \gamma, & i = j = 1; i = N, j = N - 1; \\ -a_1 \delta, & i = j = N; i = 1, j = 2; \\ Z q_{i-j+1} \delta, & \text{інше.} \end{cases}$$

$$Z_1 = \frac{a_1 \Gamma(2-\alpha) \left(\frac{\Delta \tau}{2} \right)^\alpha}{h_1^\beta}, \quad \Psi_2 = \left[\beta_1^* \Gamma(2-\gamma) h_1^\gamma U_{pl}^*, 0, \dots, 0, \beta_1 \Gamma(2-\gamma) h_1^\gamma U_{pl} \right]^T.$$

Підставляємо (22) у (23) і отримаємо систему рівнянь, яку розв'язуємо відносно функції T

$$(B + CA) T_n^{k+1/4} - \frac{\alpha c \rho}{\varepsilon \rho_0 r} C T_n^k + (\alpha C + \alpha) U_n^k + \Psi_1 + \Psi_2 = 0. \quad (24)$$

Знайшовши із (24) множину розв'язків $T_{1,m}^{k+1/4}, T_{2,m}^{k+1/4}, \dots, T_{N-1,m}^{k+1/4}, T_{N,m}^{k+1/4}, (k = 0, 1, \dots, K - 1)$, шукаємо із (22) множину розв'язків $U_{1,m}^{k+1/4}, U_{2,m}^{k+1/4}, \dots, U_{N-1,m}^{k+1/4}, U_{N,m}^{k+1/4}, (k = 0, 1, \dots, K - 1)$.

На другому півкроці інтервалу $\Delta \tau / 2$ запишемо неявну різницеву схему, у якій врахуємо тільки похідну дробового порядку β за просторовою координатою x_2 :

$$c \rho \frac{T_{n,m}^{k+1/2} - \alpha T_{n,m}^{k+1/4}}{\Gamma(2-\alpha) \left(\frac{\Delta \tau}{2} \right)^\alpha} = \frac{\lambda_2}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^{k+1/2} + \varepsilon \rho_0 r \frac{U_{n,m}^{k+1/2} - \alpha U_{n,m}^{k+1/4}}{\Gamma(2-\alpha) \left(\frac{\Delta \tau}{2} \right)^\alpha}, \quad (25)$$

$$\frac{U_{n,m}^{k+1/2} - \alpha U_{n,m}^{k+1/4}}{\Gamma(2-\alpha) \left(\frac{\Delta \tau}{2} \right)^\alpha} = \frac{a_2}{h_2^\beta} \sum_{j=0}^m q_j U_{n,m-j+1}^{k+1/2} + \frac{a_2 \delta}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^{k+1/2}. \quad (26)$$

Аналогічно як і на першому півкроці, на другому півкроці отримаємо дві системи. Внаслідок їх розв'язання отримаємо множину розв'язку відносно функції $T - T_{n,1}^{k+1/2}, T_{n,2}^{k+1/2}, \dots, T_{n,m-1}^{k+1/2}, T_{n,m}^{k+1/2}, (k = 0, 1, \dots, K - 1)$, та множину розв'язку відносно функції $U - U_{n,1}^{k+1/2}, U_{n,2}^{k+1/2}, \dots, U_{n,m-1}^{k+1/2}, U_{n,m}^{k+1/2}, (k = 0, 1, \dots, K - 1)$.

Для знаходження розв'язків на усьому інтервалі $\Delta \tau$ використаємо коректор, який реалізується на явній різницевій схемі:

$$c \rho \frac{T_{n,m}^{k+1} - \alpha T_{n,m}^k}{\Gamma(2-\alpha) \Delta \tau^\alpha} = \frac{\lambda_1}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^{k+1/2} + \frac{\lambda_2}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^{k+1/2} + \varepsilon \rho_0 r \frac{U_{n,m}^{k+1} - \alpha U_{n,m}^k}{\Gamma(2-\alpha) \Delta \tau^\alpha}, \quad (27)$$

$$\frac{U_{n,m}^{k+1} - \alpha U_{n,m}^k}{\Gamma(2-\alpha) \Delta \tau^\alpha} = \frac{a_1}{h_1^\beta} \sum_{j=0}^n q_j U_{n-j+1,m}^{k+1/2} + \frac{a_2}{h_2^\beta} \sum_{j=0}^m q_j U_{n,m-j+1}^{k+1/2} + \frac{a_1 \delta}{h_1^\beta} \sum_{j=0}^n q_j T_{n-j+1,m}^{k+1/2} + \frac{a_2 \delta}{h_2^\beta} \sum_{j=0}^m q_j T_{n,m-j+1}^{k+1/2}. \quad (28)$$

Отже, із (28) знаходимо множину розв'язку $\{U_{n,m}^{k+1} : k = \overline{0, K-1}; n = \overline{1, N}; m = \overline{1, M}\}$, а із (27) отримаємо множину розв'язку $\{T_{n,m}^{k+1} : k = \overline{0, K-1}; n = \overline{1, N}; m = \overline{1, M}\}$.

Алгоритмічні та програмні особливості. Під час написання паралельної програми потрібно вирішувати, скільки потоків і якого типу потрібно використовувати, і як вони повинні взаємодіяти. Ці рішення залежать як від конкретного програмного, так і від апаратного забезпечення. У будь-якому випадку, важливо є правильна синхронізація взаємодії потоків [6].

Наведемо формальний опис паралельного алгоритму, базованого на описаній чисельній схемі (11)-(28). Нехай є p обчислювальних модулів (процесорів, ядер або потоків) однаковою продуктивності. У такому разі потрібно розбити

розрахункову область на p однакових підобластей. Зокрема, кожен обчислювач (потік) буде виконувати операції в межах доступної йому підобласті. Оскільки алгоритм задачі побудований на базі схеми предиктор-коректор, то постає потреба двічі – на першому та другому півкроках $\Delta\tau/2$ виконувати обчислення за неявною схемою за різними просторовими координатами і коректувати їх явною.

Варто зазначити, що на відміну від класичних рівнянь тепло- і вологоперенесення, у рівняннях з частинними похідними із дробовим порядком за часом τ та просторовими координатами x та y потрібно зберігати значення невідомих на всіх тимчасових шарах. Кожен обчислювач на кожному часовому шарі повинен мати допоміжні змінні для зберігання значень, переданих з попереднього часового шару. Для першого часового шару використаємо значення початкових умов. Алгоритм задачі можна описати за допомогою таких пунктів [5]:

1. Виходячи із початкових умов і параметрів розбиття сітки, всі потоки паралельно розраховують коефіцієнти для першого півкроку $\Delta\tau/2$.
2. Для обчислення на наступному часовому кроці виконуємо такі настанови:
 - a) обчислюються нові граничні коефіцієнти враховуючи значення з попереднього шару;
 - b) виконуємо обчислення на першому півкроці $\Delta\tau/2$;
 - c) кожний потік використовує внутрішній ітераційний процес;
 - d) потоки, що відповідають за розрахунок сусідніх підобластей, обмінюються двома прикордонними значеннями. При цьому враховуються значення з попереднього тимчасового шару, а на наступних ітераційних кроках – значення з попередньої ітерації;
 - e) кожен потік паралельно розв'язує свою систему методом прогонки;
 - f) завершуються обчислення на першому півкроці $\Delta\tau/2$;
 - g) якщо номер ітерації менше p , повтор з кроку b), інакше перехід на новий часовий шар і повтор з кроку 2.

3. Виходячи із початкових умов і параметрів розбиття сітки, всі потоки паралельно розраховують коефіцієнти для другого півкроку $\Delta\tau/2$ і незмінні складові прогоночних коефіцієнтів.

4. Виконуємо пункти 2(a) – 2(g) для другого півкроку.

5. Для завершення розв'язку на усьому інтервалі виконуємо коректування результатів за допомогою співвідношення коректор.

На рис. 1 подано схему реалізації описаного алгоритму. Враховано розподілення обчислень на потоках. Кількість потоків визначається залежно від розміру сітки розбиття області, яку будемо досліджувати. На базі запропонованого алгоритму створено програмне забезпечення чисельного моделювання неізотермічного вологоперенесення у середовищах із фрактальною структурою, яке подає результат в табличному і графічному вигляді.

Проектування програмної системи. Реалізація запропонованого алгоритму з використанням ООП полягає у розробленні класів і відношень між ними. Створено класи, що реалізують: характеристики деревини (геометричні, тепло-механічні); розбиття області на елементи; керування обчисленнями (послідовними, паралельними, розв'язування СЛАР та ін.); представлення результатів, інтерфейс користувача. Важливі особливості проектного програмного забезпечення відображено на рис. 2 у графічній нотатції UML.

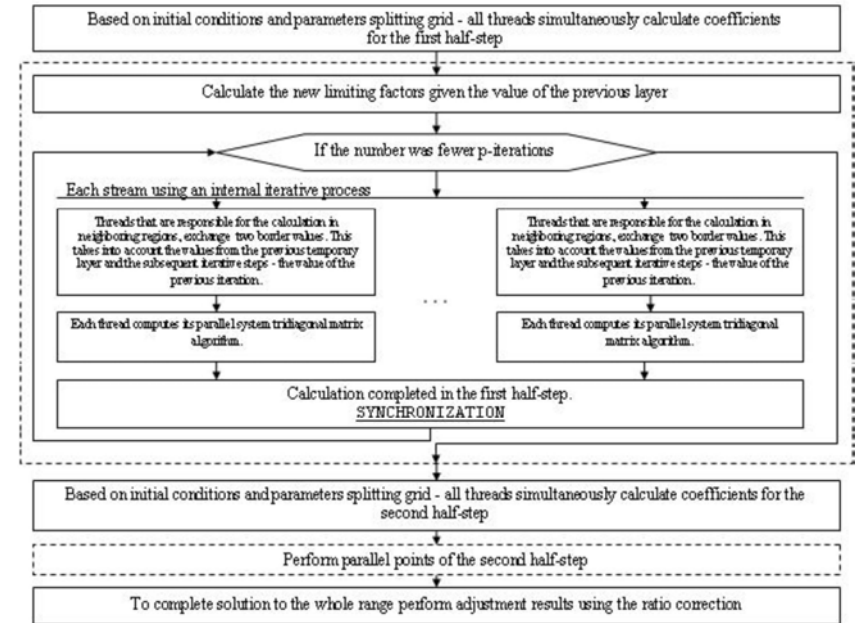


Рис. 1. Схема алгоритму програмного забезпечення для чисельного моделювання

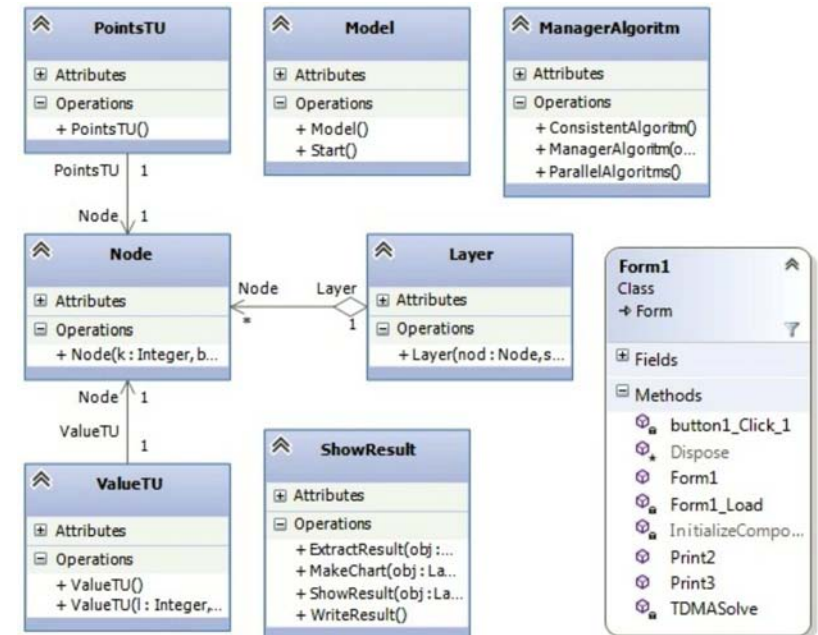


Рис. 2. UML діаграма класів і відношення між ними

Для реалізації запропонованого алгоритму було спроектовано спеціальний клас ManagerAlgorithm. Цей клас містить методи ConsistentAlgorithm для послідовного виконання обчислень, ParallelAlgorithms для паралельного виконання обчислень, методи, які дають змогу підраховувати час виконання, а також конструктори для створення об'єктів цього класу. Клас Model призначений для керування процесом запуску моделювання, відповідно у ньому міститься метод Start. Запроектвані методи ExtractResult, ShowResult, MakeChart поєднанні у класі ShowResult та здійснюють представлення результатів з виведенням у файл, табличному та графічному вигляді. Між класами Layer та Node реалізовано відношення агрегації, що є слабким зв'язком, тобто в цьому випадку об'єкти Layer та Node будуть рівноправними. Кратність цього зв'язку 1 до багатьох, а це свідчить про те, що один шар буде містити багато вузлів. У конструктор Layer передається посилання на вже існуючий об'єкт Node, що збільшує гнучкість програми. Водночас, клас Node має відношення асоціації з класами PointsTU та ValueTU – це відношення, за якого об'єкти класу одного типу пов'язані з об'єктами іншого типу. На UML діаграмі класів асоціацію позначено у вигляді звичайної стрілки. У цьому випадку запроєктовано зв'язок один до одного – одному вузлу буде відповідати одна точка та один набір характеристик. Ще одним класом програми є клас Form1, який призначений для діалогу з користувачем та містить елементи форми та деякі методи.

На рис. 3 подано інтерфейс користувача програми, а саме форму для заповнення вхідних параметрів. Щоб здійснити моделювання, спочатку потрібно внести потрібні вхідні параметри, такі як: фізичні характеристики досліджуваного взірця, температуру, вологість взірця та зовнішнього середовища, кількість вузлів, час моделювання та ін.

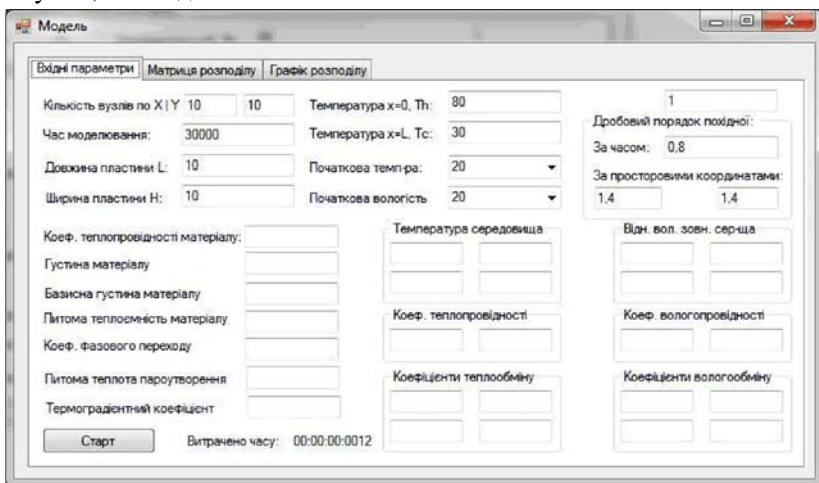


Рис. 3. Діалогове вікно програми

Задавши всі вхідні параметри, можна приступити до процесу моделювання. Зважаючи на те, що реалізація алгоритму є трудомістким процесом і враховуючи розмірність сітки розбиття та час моделювання, потрібно очікувати на три-

вале моделювання. На час моделювання також впливають і технічні характеристики апаратного забезпечення, на якому буде виконуватися програма. Дані результату моделювання в табличному представленні можна спостерігати на рис. 4.

80	80	80	80	80	80	80
67,4	67,4	67,4	67,4	67,4	67,4	67,4
56,56	56,56	56,56	56,56	56,56	56,56	56,56
49,01	49,01	49,01	49,01	49,01	49,01	49,01
45,88	45,88	45,88	45,88	45,88	45,88	45,88
47,8	47,8	47,8	47,8	47,8	47,8	47,8
54,93	54,93	54,93	54,93	54,93	54,93	54,93
66,8	66,8	66,8	66,8	66,8	66,8	66,8
82,34	82,34	82,34	82,34	82,34	82,34	82,34
99,91	99,91	99,91	99,91	99,91	99,91	99,91

Рис. 4. Табличне представлення результатів моделювання

На основі табличних даних результатів моделювання реалізовано методи представлення вологи та температури у графічному вигляді.

Висновки. У нотаціях об'єктно-орієнтованого підходу та паралельного програмування запропоновано алгоритм для чисельного моделювання неізотермічного вологоперенесення у середовищах із фрактальною структурою.

Розроблене багатопотокове паралельне програмне забезпечення може бути ефективним для розв'язку прикладних задач на багатопроцесорних системах і процесорах, що використовують технологію гіперпоточності.

Створені класи у реалізованій програмній системі забезпечують не тільки масштабування, а й легке розширення паралельної реалізації, що дасть змогу реалізовувати схожі складніші моделі.

Література

1. Соколовський Я.І. Числовий метод дослідження теплообміну на підставі похідних дробового порядку / Я.І. Соколовський, М.В. Москвітін // Технічні вісті : зб. наук. праць. – 2015. – Вип. 1(41), 2(42). – С. 12-15.
2. Учайкин В.В. Метод дробных производных / В.В. Учайкин. – Ульяновск : Изд-во "Артишок", 2008. – 512 с.
3. Podlubny I. Fractional Differential Equations / I. Podlubny. 198 of Mathematics in Science and Engineering, Academic Press, San Diego, Calif, USA, 1999. – 340 s.
4. Васильев В.В. Дробное исчисление и аппроксимационные методы в моделировании динамических систем : науч. изд. / В.В. Васильев, Л.А. Симак. – К. : Изд-во НАН Украины, 2008. – 256 с.
5. Sokolovskyy Y. Mathematical modeling of heat and moisture transfer and rheological behavior in materials with fractal structure using the parallelization of predictor-corrector numerical method / Y. Sokolovskyy, V. Shymanskyi, M. Levkovich, V. Yarkun // 2016 IEEE First International Conference on Data Stream Mining & Processing (DSMP), Lviv, 2016. – Pp. 108-111.
6. Грегори Р. Эндриос Основы многопоточного, параллельного и распределенного программирования / Р. Грегори. – М. : Изд. дом "Вильямс". 2003. – 512 с.
7. Матвеев С.А. Параллельная реализация быстрого метода решения уравнений агрегационно-фрагментационной кинетики типа уравнений Смолуховского / С.А. Матвеев // Вычислительная математика и программирование. – 2015. – Т. 16. – Вип. 3. – С. 360-368.

8. Кудерметов Р.К. Многопоточная реализация четырехточечного блочного одношагового метода решения дифференциальных уравнений / Р.К. Кудерметов // Электротехнические и компьютерные системы : сб. науч. тр. – 2015. – № 17. – С. 110-116.

9. Соколовський Я.І. Чисельний метод дослідження неізотермічного вологоперенесення у середовищах з фрактальною структурою / Я.І. Соколовський, М.В. Левкович // Вісник Національного університету "Львівська політехніка". – Сер.: Комп'ютерні системи проектування. Теорія і практика. – Львів : Вид-во НУ "Львівська політехніка". – 2016. – № 844. – С. 124-132.

Надійшла до редакції 7.10.2016 р.

Соколовский Я.И., Яркун В.И., Левкович М.В. Алгоритмическое и программное обеспечение для исследования неізотермического влагопереноса в средах с фрактальной структурой

Сформированы требования и выбран интегрированный подход к созданию алгоритмического обеспечения численного моделирования неізотермического влагопереноса в средах с фрактальной структурой, который охватывает разработку параллельного алгоритма решения систем дифференциальных уравнений в частных производных. Предложена синхронизация взаимодействия потоков для параллельных обменов между временными шарами. Разработано программное обеспечение и создан интерфейс для визуализации результатов исследования.

Ключевые слова: параллельные вычисления, потоки, дифференциальные уравнения дробного порядка, влагоперенос.

Sokolovskyy Ya.I., Yarkun V.I., Levkovych M.V. Algorithms And Software for the Research of Unisothermal Moisture Transfer in the Environment with Fractal Structure

The requirements are specified and the integrated approach to creating numerical simulation algorithm software for unisothermal moisture transfer in environments with fractal structure is selected, which includes the development of parallel algorithm for solving systems of differential equations in partial derivatives. Synchronization streams for parallel interaction exchanges between temporal layers are proposed. Software is developed and user interface to visualize the results of the research is created.

Keywords: parallel computing, flow, differential equations of fractional order, moisture transfer.

УДК 539.3

КОНЦЕНТРАЦІЯ НАПРУЖЕНЬ БІЛЯ ЕЛІПСОЇДАЛЬНОГО ВКЛЮЧЕННЯ У ТІЛІ ВНАСЛІДОК ПОПЕРЕДНЬОГО ПЛАСТИЧНОГО ДЕФОРМУВАННЯ

М.М. Стадник¹, І.Я. Горбачевський²

Досліджено напруження, що зберігаються у неоднорідному пружному тілі з еліпсоїдальним та сфероїдальним нежорсткими чужорідними включеннями після зняття зовнішніх сил, що призвели до попереднього пластичного деформування, у місцях їх найбільшої концентрації. Задачу розв'язано наближено з урахуванням тонкості включення. За тестову взято задачу для пружного сталюого тіла з нікелевим та мідним сфероїдальним включеннями, коли попередньо композитові надали 8 % залишкової деформації. Показано, що величина залишкових напружень є істотною, і може позначитися на майбутній експлуатації такого композита.

Ключові слова: включення, концентрація напружень, залишкові деформації.

¹ проф. М.М. Стадник, д-р техн. наук – НЛТУ України, м. Львів;
² доц. І.Я. Горбачевський, канд. техн. наук – НЛТУ України, м. Львів

Модельне представлення задачі. Розглянемо безмежне тривимірне тіло із пружним ($0 \leq E_1 / E \leq 1$, E_1, E – модулі пружності включення і матриці, відповідно) включенням, обмеженим деякою гладкою поверхнею Σ , симетричною відносно серединної площини S .

Систему прямокутних декартових координат $Oxyz$ виберемо так, щоб область S лежала в площині xOy , а поверхня Σ описувалася функцією $|z| = h(x, y)$, $(x, y) \in S$. При цьому вважаємо, що висота включення $2h(x, y)$ і радіуси кривини $\rho(x, y, 0)$ контурів його поперечних перерізів площинами, які проходять через вісь Oz , є малими, порівняно з іншими лінійними розмірами включення. Нехай тіло із включенням спочатку піддають дії зовнішніх рівномірно розподілених навантажень, що паралельні до осі Oz і симетричні відносно площини $z = 0$, поза границю пружності, а після тіло повністю звільняють від дії зусиль. Оскільки в загальному випадку механічні характеристики матриці і включення різні, у тілі виникають внутрішні залишкові напруження. Мета цього дослідження – визначити концентрацію залишкових нормальних напружень $\sigma_{zz}^{(y)}$ у тілі (матриці) при $z = 0$ в околі контуру L включення, що обмежує область S .

У праці [1] запропоновано моделювати задачу як суперпозицію двох задач: задачі (1) для тіла з порожниною, яка має конфігурацію включення Σ , до берегів якої прикладені невідомі контактні напруження $\sigma_{zz}^{(k)}$; та задачі (2) для тіла з включенням, на яке діють ті ж самі контактні напруження. За припущення, що при розвантаженні виконується закон Гука, і розриви у тілі відсутні, для визначення $\sigma_{zz}^{(k)}$ запропоновано таку умову, що зв'язує деформації у тілі та у включенні

$$u_z^{(1)}(x, y) + u_z^{(2)}(x, y) = (e - e_1)h(x, y), \quad (x, y) \in S, \quad (1)$$

де: e і e_1 – середні залишкові пластичні деформації матриці й включення; $u_z^{(1)}(x, y)$ – компонента пружних зміщень точок поверхні порожнини в матриці, що має конфігурацію включення Σ , до берегів якої прикладені нормальні напруження $\sigma_{zz}^{(k)}$; $u_z^{(2)}(x, y)$ – компонента пружних зміщень точок поверхні включення за дії цих же напружень. Враховуючи тонкість включення і симетричність напружено-деформованого стану відносно площини $z = 0$, для задачі (2) зміщення $u_z^{(2)}(x, y)$ можна обчислювати так:

$$u_z^{(2)}(x, y) = \frac{\sigma_{zz}^{(k)}}{2G_1 d_5} h(x, y), \quad (2)$$

де: G_1 – модуль зсуву; μ_1 – коефіцієнт Пуассона матеріалу включення. Тут і далі для стислості введені позначення: $d_1 = 1 - \mu$, $d_3 = 1 - 2\mu$, $d_4 = 1 - \mu_1$, $d_5 = 1 + \mu_1$, $d_6 = 1 - 2\mu_1$.

Для наближеного визначення зміщень $u_z^{(1)}(x, y)$, $(x, y) \in S$ задачу (1) моделюємо задачею математичної теорії тріщин. Відтак, граничні умови зносимо на серединну площину xOy і приходимо до крайової задачі для півпростору $z \geq 0$: