КІБЕРНЕТИКА та КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

УДК 519.85 DOI:10.34229/2707-451X.25.1.3

В.О. ЖИДКОВ, П.І. СТЕЦЮК, О.М. ХОМ'ЯК

МЕТОД НАЙМЕНШИХ КВАДРАТІВ ТА МЕТОД НАЙМЕНШИХ МОДУЛІВ ДЛЯ ПОШУКУ ДЕФЕКТІВ У РЕГУЛЯРНИХ ЗОБРАЖЕННЯХ

Описано метод найменших квадратів (МНК) та метод найменших модулів (МНМ) для пошуку дефектів у регулярних зображеннях (регулярна та базисна регулярна структури). Для МНК сформульовано задачі безумовної мінімізації опуклих квадратичних функцій, які дозволяють знайти найкращі параметри регулярних структур за критерієм найменших квадратів, а для МНМ – задачі безумовної мінімізації опуклих кусочно-лінійних функцій для пошуку найкращих параметрів за критерієм найменших модулів. Наведено коди octave-функцій для обчислення градієнтів гладких функцій та субградієнтів негладких функцій за допомогою матрично-векторних операцій. Описано використання r-алгоритму для розв'язання за допомогою МНК та МНМ тестових задач для регулярних зображень розмірами 400×600 та 1000×1500 пікселів.

Ключові слова: *регулярна 3D-структура, метод найменших квадратів, метод найменших модулів, r-алгоритм.*

Вступ. Регулярні зображення з областями дефектів є характерними при неруйнуючому контролі якості тонкостінних багатошарових композиційних матеріалів [1]. Щоб знизити вплив людського фактору потрібно автоматизувати процес визначення місць розташування дефектів у відповідальних елементах конструкцій. У роботі розглянемо оптимізаційні задачі для пошуку дефектів у регулярних зображеннях, або регулярних 3D-структурах [2–4] і методи їх розв'язання.

3D-структурою будемо називати трійку $\{A, u, v\}$, де A – $m \times n$-матриця, $A = \{a_{ij}\}_{i=1,\dots,m}^{j=1,\dots,n} \in \mathbb{R}^{m \times n}$, u, v – вектори, $u \in \mathbb{R}^m$, $v \in \mathbb{R}^n$. 3D-структура $\{A^*, u^*, v^*\}$ – регулярна, якщо $a_{ij}^* = u_i^* + v_j^*$, $i = 1, \dots, m$, $j = 1, \dots, n$. Вектори u^*, v^* – параметри 3D-структури $\{A^*, u^*, v^*\}$. Регулярну 3D-структуру $\{A^*, u^*, v^*\}$ будемо називати базисною, якщо $\sum_{j=1}^{j=m} u_j^* = \sum_{i=1}^{i=n} v_i^*$.

Далі розглянемо задачі апроксимації коефіцієнтів довільної $m \times n$-матриці A за допомогою коефіцієнтів матриці A^* для регулярної 3D-структури за критерієм найменших квадратів та критерієм найменших модулів. Результуюча матриця $\bar{A} = A - A^*$ визначає області дефектів регулярного зображення.

Матеріал статті викладений у такому порядку.

У розділі 1 описано чотири оптимізаційні задачі для знаходження найкращих параметрів регулярних структур та наведено їх властивості. Перша та друга задачі відповідають знаходженню найкращих за критерієм найменших квадратів регулярної та базисної регулярної структур, а третя та четверта – найкращих за критерієм найменших модулів регулярної та базисної регулярної структур. У розділі 2 описано способи обчислення градієнтів гладких функцій та субградієнтів негладких функцій для всіх чотирьох задач. У розділі 3 наведено опис результатів застосування r-алгоритму для розв'язання задач з матрицями розмірів 400×600 та 1000×1500.

1. Метод найменших квадратів (МНК) та метод найменших модулів (МНМ) для пошуку дефектів у регулярних 3D-структурах

Нехай A – задана $m\times n$-матриця, e_m – одиничний m-вектор, e_n – одиничний n-вектор. Якщо $\{A^*, x^*, y^*\}$ – регулярна або базисна регулярна 3D-структура, то $m\times n$-матриця A^* має вигляд $A^* = x^* e_n^T + e_m (y^*)^T$, де параметри $x^* \in \mathbb{R}^m$, $y^* \in \mathbb{R}^n$. Будемо розглядати оптимізаційні задачі для знаходження параметрів регулярної та базисної регулярної структури, щоб коефіцієнти довільної $m\times n$-матриці A найменше відрізнялися від коефіцієнтів матриці A^* за критерієм найменших квадратів та критерієм найменших модулів. Першому критерію буде відповідати МНК, а другому – МНМ.

Задача 1. МНК для регулярної 3D-структури $\{A^*, x^*, y^*\}$. Потрібно знайти такі $x^* \in R^m$ і $y^* \in R^n$, щоб коефіцієнти $m\times n$-матриці $A^* = x^* e_n^T + e_m (y^*)^T$ мінімально відхилялись від коефіцієнтів матриці A за критерієм найменших квадратів.

Задача 1 еквівалентна оптимізаційній задачі: знайти

$$(x^*, y^*) = \underset{x\in R^m, y\in R^n}{\arg\min} \left\{ f_1(x,y) = \sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij} - x_i - y_j)^2 \right\}, \tag{1}$$

яка є задачею безумовної мінімізації опуклої квадратичної функції від $m+n$ змінних. Оптимальне значення функції $f_1^* = f_1(x^*, y^*)$ може досягатися при різних значеннях параметрів (x^*, y^*), тому що задача (1) має безліч розв'язків [2]. Так, якщо (x^*, y^*) – розв'язок задачі (1), то $(x^* + te_m, y^* - te_n)$ при довільному $t \neq 0$ також буде розв'язком задачі (1).

Задача 2. МНК для базисної регулярної 3D-структури $\{A^{**}, x^{**}, y^{**}\}$**.** Потрібно знайти такі $x^{**} \in R^m$ і $y^{**} \in R^n$, щоб коефіцієнти $m\times n$-матриці $A^{**} = x^{**} e_n^T + e_m (y^{**})^T$ мінімально відхилялись від коефіцієнтів матриці A за критерієм найменших квадратів.

Задача 2 еквівалентна оптимізаційній задачі: знайти

$$(x^{**}, y^{**}) = \underset{x\in R^n, y\in R^n}{\arg\min} \left\{ f_2(x,y) = \sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij} - x_i - y_j)^2 + \left(\sum_{i=1}^{n} x_i - \sum_{j=1}^{m} y_j \right)^2 \right\}, \tag{2}$$

яка є задачею безумовної мінімізації опуклої квадратичної функції від $m+n$ змінних. Оптимальне значення функції $f_2^* = f_2(x^{**}, y^{**})$ досягається в єдиній точці (x^{**}, y^{**}), яку для довільної точки (x^*, y^*) – розв'язку задачі (1), можна розрахувати за формулами

$$x^{**} = x^* - \frac{e_m^T x^* - e_n^T y^*}{n+m} e_m, \quad y^{**} = y^* + \frac{e_m^T x^* - e_n^T y^*}{n+m} e_n.$$

При цьому виконується рівність $f_2^* = f_1^*$.

Задача 3. МНМ для регулярної 3D-структури $\{A^*, x^*, y^*\}$. Потрібно знайти такі $x^* \in R^m$ і $y^* \in R^n$, щоб коефіцієнти $m\times n$-матриці $A^* = x^* e_n^T + e_m (y^*)^T$ мінімально відхилялись від коефіцієнтів матриці A за критерієм найменших модулів.

Задача 3 еквівалентна оптимізаційній задачі: знайти

$$(x^*, y^*) = \underset{x\in R^m, y\in R^n}{\arg\min} \left\{ f_3(x,y) = \sum_{i=1}^{m}\sum_{j=1}^{n} \left| a_{ij} - x_i - y_j \right| \right\}, \qquad (3)$$

яка є задачею безумовної мінімізації опуклої кусково-лінійної функції від $m+n$ змінних. Оптимальне значення функції $f_3^* = f_3(x^*, y^*)$ може досягатися при різних значеннях параметрів (x^*, y^*). Задача (3) має безліч розв'язків, як і задача (1).

Задача 4. МНМ для базисної регулярної 3D-структури $\{A^{**}, x^{**}, y^{**}\}$. Потрібно знайти такі $x^{**} \in R^m$ і $y^{**} \in R^n$, щоб коефіцієнти $m\times n$-матриці $A^{**} = x^{**}e_n^T + e_m(y^{**})^T$ мінімально відхилялись від коефіцієнтів матриці A за критерієм найменших модулів.

Задача 4 еквівалентна оптимізаційній задачі: знайти

$$(x^*, y^*) = \underset{x\in R^m, y\in R^n}{\arg\min} \left\{ f_4(x,y) = \sum_{i=1}^{m}\sum_{j=1}^{n} \left| a_{ij} - x_i - y_j \right| + \left| \sum_{i=1}^{m} x_i - \sum_{j=1}^{n} y_j \right| \right\}, \qquad (4)$$

яка є задачею безумовної мінімізації опуклої кусково-лінійної функції від $m+n$ змінних. Оптимальне значення функції $f_4^* = f_4(x^{**}, y^{**})$ досягається в єдиній точці (x^{**}, y^{**}). При цьому виконується рівність $f_4^* = f_3^*$.

Для розв'язання задач 1 – 4 можна використовувати солвери з NEOS-сервера [5]. Так, наприклад, для задач 1 та 2 можна використовувати солвери для розв'язання задач нелінійного програмування SNOPT [6], IPOPT [7] та інші. Задачі (3) та (4) можуть бути зведені до задач лінійного програмування і для їх розв'язання можна використати солвери Gurobi [8], CPLEX [9] та інші.

Однак, задачі (3) та (4) – це задачі безумовної мінімізації опуклих кусково-лінійних функцій і для їх розв'язання можна використовувати субградієнтні методи мінімізації негладких функцій. Ці методи також можна використати і для задач (1) та (2) – задач безумовної мінімізації опуклих квадратичних функцій. Для використання субградієнтних методів потрібні оракули, які обчислюють значення функції та субградієнта у точках ітераційного процесу.

2. Обчислення градієнтів та субградієнтів для задач (1)–(4)

Для обчислення значень функцій $f_1(x,y)$, $f_2(x,y)$ та їх градієнтів реалізовано octave-функції `fg1(x)`, `fg2(x)`, а для обчислення значень функцій $f_3(x,y)$, $f_4(x,y)$ та їх субградієнтів – octave-функції `fg3(x)`, `fg4(x)`. Обчислення значень функцій та їх (суб)градієнтів проводилися мовою Octave [10] з використанням матрично-векторних операцій. Задана $m\times n$-матриця A передається в octave-функції через глобальну змінну **`A`**.

Значення функції $f_1(x,y)$ та її градієнта $g_{f_1}(x,y)$ обчислюються за формулами

$$f_1(x,y) = \sum_{i=1}^{m}\sum_{j=1}^{n} (a_{ij} - x_i - y_j)^2, \quad g_{f_1}(x,y) = \begin{cases} -2\sum_{j=1}^{n}(a_{ij} - x_i - y_j), & i = 1,\ldots,m, \\ -2\sum_{i=1}^{m}(a_{ij} - x_i - y_j), & j = 1,\ldots,n. \end{cases} \qquad (5)$$

Формули (5) поставлені в основу octave-функції `fg1(x)`, код якої наведено далі:

```
function [f,g] = fg1(x)
global A # A - задана m*n матриця
n=columns(A); m=rows(A);
A1 = A - x(1:m,1)*ones(1,n)-ones(m,1)*x((m+1):(m+n),1)';
f=sum(sum(A1.*A1));
g=[-2*sum(A1')'; -2*sum(A1)'];
endfunction
```

Значення функції $f_2(x,y)$ та її градієнта $g_{f_2}(x,y)$ обчислюються за формулами

$$f_2(x,y)=f_1(x,y)+\left(\sum_{i=1}^{n}x_i-\sum_{j=1}^{m}y_j\right)^2,\quad g_{f_2}(x,y)=g_{f_1}(x,y)+\begin{cases}2\left(\sum_{i=1}^{n}x_i-\sum_{j=1}^{m}y_j\right), & i=1,\dots,m,\\ -2\left(\sum_{i=1}^{n}x_i-\sum_{j=1}^{m}y_j\right), & j=1,\dots,n.\end{cases} \tag{6}$$

Формули (6) поставлені в основу octave-функції `fg2(x)`, код якої наведено далі:

```
function [f,g] = fg2(x)
global A # A - задана m*n матриця
n=columns(A); m=rows(A);
A1 = A - x(1:m,1)*ones(1,n)-ones(m,1)*x((m+1):(m+n),1)';
f=sum(sum(A1.*A1));
g=[-2*sum(A1')'; -2*sum(A1)'];
temp=sum(x(1:m,1))-sum(x((m+1):(m+n),1));
f=f+temp*temp;
g=g+2*temp*[ones(m,1); -ones(n,1)];
endfunction
```

Значення функції $f_3(x,y)$ та її градієнта $g_{f_3}(x,y)$ обчислюються за формулами

$$f_3(x,y)=\sum_{i=1}^{m}\sum_{j=1}^{n}\left|a_{ij}-x_i-y_j\right|,\quad g_{f_3}(x,y)=\begin{cases}-\sum_{j=1}^{n}sign\left(a_{ij}-x_i-y_j\right), & i=1,\dots,m,\\ -\sum_{i=1}^{m}sign\left(a_{ij}-x_i-y_j\right), & j=1,\dots,n.\end{cases} \tag{7}$$

Формули (7) поставлені в основу octave-функції `fg3(x)`, код якої наведено далі:

```
function [f,g] = fg3(x)
global A # A - задана m*n матриця
n=columns(A); m=rows(A);
A1 = A - x(1:m,1)*ones(1,n)-ones(m,1)*x((m+1):(m+n),1)';
f=sum(sum(abs(A1)));
g=[-sum(sign(A1'))'; -sum(sign(A1))'];
endfunction
```

Значення функції $f_4(x,y)$ та її градієнта $g_{f_4}(x,y)$ обчислюються за формулами

$$f_4(x,y)=f_3(x,y)+\left|\sum_{i=1}^{m}x_i-\sum_{j=1}^{n}y_j\right|,\quad g_{f_4}(x,y)=g_{f_3}(x,y)+\begin{cases} sign\left(\sum_{i=1}^{m}x_i-\sum_{j=1}^{n}y_j\right), & i=1,\ldots,m,\\ -sign\left(\sum_{i=1}^{m}x_i-\sum_{j=1}^{n}y_j\right), & j=1,\ldots,n.\end{cases} \tag{8}$$

Формули (8) поставлені в основу octave-функції `fg4(x)`, код якої наведено далі:

```
function [f,g] = fg4(x)
global A # A - задана m*n матриця
n=columns(A); m=rows(A);
A1 = A - x(1:m,1)*ones(1,n)-ones(m,1)*x((m+1):(m+n),1)';
f=sum(sum(abs(A1)));
g=[-sum(sign(A1'))'; -sum(sign(A1))'];
temp=sum(x(1:m,1))-sum(x((m+1):(m+n),1));
f=f+abs(temp);
g=g+sign(temp)*[ones(m,1); -ones(n,1)];
endfunction
```

Далі наведемо результати обчислювальних експериментів щодо застосування r-алгоритму Шора [11, 12] для знаходження параметрів регулярних зображень невеликих розмірів – 400 пікселів по вертикалі та 600 пікселів по горизонталі, та середніх розмірів – 1000 та 1500 пікселів. При цьому розміри задач (1)–(4) для невеликих зображень будуть визначатися розмірами $n=600$ і $m=400$, а для середніх зображень – розмірами $n=1500$ і $m=1000$. Обчислення проводились на комп'ютері з процесором Intel Core i5-9400f з 2.9 GHz, використовуючи GNU Octave версії 5.1.0.

Мета обчислювальних експериментів – це оцінка часу розв'язання задач (1) – (4) вказаних розмірів на сучасних ПЕОМ, якщо для реалізації МНК (задачі 1 та 2) та МНМ (задачі 3 та 4) використовується одна і та ж програмна реалізація r-алгоритму, а саме програма **ralgb5a** [13]. При розрахунках вибиралась одна і та ж стартова точка $x_0=(0,0,\ldots,0)^T$, використовувались такі параметри r-алгоритму: $\alpha=2$, $h_0=1$, $q_1=0.9$ (МНК) і $q_1=0.95$ (МНМ), і такі параметри зупинки: $\varepsilon_x=10^{-6}$, $\varepsilon_g=10^{-12}$, **maxitn = 1000**, друк інформації про хід ітераційного процесу здійснювався через кожних **intp = 100** ітерацій. Зауважимо, що програма **ralgb5a** використовує ще два параметри r-алгоритму, значення яких дорівнюють $q_2=1.1$, $n_h=3$ та не підлягають зміні.

3. *r*-алгоритм для розв'язання тестових задач (1)–(4)

Перший експеримент пов'язаний з відновленням за допомогою МНК (задачі 1 та 2) параметрів регулярної та базисної регулярної структур без дефектів для розмірів $n=600$ і $m=400$, $n=1500$ і $m=1000$. Для цього п'ять різних варіантів заданої $m\times n$-матриці A розраховувались за формулою $A=xe_n^T+e_my^T$, де компоненти векторів $x\in\mathbb{R}^m$, $y\in\mathbb{R}^n$ генерувалися датчиком випадкових чисел **rand**("seed",2025) в діапазоні від 0 до 265. Для обчислення значень функцій $f_1(x,y)$, $f_2(x,y)$ та їх градієнтів використовуються octave-функції `fg1(x)` **та** `fg2(x)`.

Перший експеримент реалізує такий octave-код.

```
clear;                # очищуємо пам'ять для першого тесту
global A              # A - задана m*n матриця
n = 600, m = 400,     # n = 1500, m = 1000,
# Встановлюємо параметри r-алгоритму
alpha = 2, h0 = 1.0,  q1 = 0.9,
epsx = 1.e-6, epsg = 1.e-12, maxitn = 1000, intp=100,
# Запускаємо на розрахунок по п'ять тестових задач (1) та (2)
scale = 256; rand("seed",2025); tab12 = []; ntest = 5;
for ii=1:ntest
   # Генеруємо матрицю A та параметри базисної регулярної 3D-структури
   x = scale*rand(m,1);  y = scale*rand(n,1);
   A = x*ones(1,n)+ones(m,1)*(y)';
   temp = (sum(x)-sum(y))/(n+m);
   xb = x - temp*ones(m,1);  yb = y + temp*ones(n,1);
   # Розв'язуємо задачу (1)
   x0=zeros(m+n,1); tstart=time();
   [xr1,fr1,itn1,nfg1,istop1] =
ralgb5a(@fg1,x0,alpha,h0,q1,epsg,epsx,maxitn,intp);
   temp=(sum(xr1(1:m))-sum(xr1(m+1:m+n)))/(n+m);
   xr1 = xr1 - temp*[ones(m,1);ones(n,1)];
   itn1, nfg1, istop1, fr1, dx1=norm(xr1-[xb; yb]),
   time1=time()-tstart,
   # Розв'язуємо задачу (2)
   x0=zeros(m+n,1); tstart=time();
   [xr2,fr2,itn2,nfg2,istop2] =
ralgb5a(@fg2,x0,alpha,h0,q1,epsg,epsx,maxitn,intp);
   itn2, nfg2, istop2, fr2, dx2=norm(xr2-[xb; yb]),
   time2=time()-tstart,
   tab12 = [tab12; ii itn1 nfg1 time1 dx1 fr1 itn2 nfg2 time2 dx2 fr2];
endfor
tab12,  # друкуємо результати розрахунку для першого тесту
```

Результати першого експерименту для $n = 600$ і $m = 400$ наведені в табл. 1, а для $n = 1500$ і $m = 1000$ – в табл. 2, де **itn** – кількість ітерацій *r*-алгоритму, **nfg** – кількість обчислень значення функції та її градієнта за формулами (5) та (6), **time** – час виконання програми в секундах, **delta** – норма відхилення знайденого наближення до точки мінімуму від точки мінімуму, **fr** – оптимальне значення функції.

ТАБЛИЦЯ 1. Затрати *r*-алгоритму для МНК ($m = 400$ і $n = 600$, $q_1 = 0.9$)

№ п/п	Задача (1)					Задача (2)				
	itn	nfg	time	delta	fr	itn	nfg	time	delta	fr
1	291	579	4.28	1.5e-06	1.4e-09	295	586	4.54	1.6e-07	1.5e-11
2	306	606	4.73	4.6e-07	9.7e-11	286	573	4.57	7.9e-08	3.5e-12
3	308	612	4.58	2.3e-07	2.9e-11	276	549	4.12	2.1e-07	2.3e-11
4	298	593	4.42	1.6e-07	1.6e-11	291	583	4.32	3.2e-07	4.1e-11
5	299	595	4.43	2.1e-07	1.9e-11	284	562	4.28	1.0e-07	5.5e-12

ТАБЛИЦЯ 2. Затрати r-алгоритму для МНК ($m = 1000$ і $n = 1500$, $q_1 = 0.9$)

№ п/п	Задача (1)					Задача (2)				
	itn	nfg	time	delta	fr	itn	nfg	time	delta	fr
1	298	604	30.12	1.3e-07	1.9e-11	303	615	30.64	3.5e-07	1.8e-10
2	301	615	30.49	3.2e-07	1.4e-10	304	614	30.61	2.5e-07	6.8e-11
3	275	557	27.76	2.2e-07	6.3e-11	294	595	29.70	3.8e-07	1.7e-10
4	306	616	30.60	2.7e-07	8.6e-11	325	657	32.87	2.4e-07	6.5e-11
5	304	612	30.40	1.7e-07	3.3e-11	292	592	29.52	3.2e-07	1.2e-10

З колонки **delta** табл. 1 та 2 видно, що знайдені параметри регулярних 3D-структур відхиляються від точних на величину порядку 10^{-7} за евклідовою нормою. Це означає, що наближення до точок мінімуму знайдено досить точно, про що сигналізують і знайдені з точністю $10^{-12} \div 10^{-10}$ мінімальні значення функцій $f_1(x, y)$, $f_2(x, y)$ (див. колонку **fr**). Для вирішення задач 1 та 2 за допомогою МНК на базі r-алгоритму достатньо менше 5 секунд, якщо $n = 600$ і $m = 400$, та близько 30 секунд, якщо $n = 1500$ і $m = 1000$. Це означає, що розроблені програми можна використовувати у діалоговому режимі для знаходження дефектів у регулярних 3D-структурах розмірів $n = 600 \div 1500$ і $m = 400 \div 1000$, тому що для цього потрібно від 5 до 30 секунд на сучасних персональних ЕОМ з використанням GNU Octave версії 5.1.0.

Другий експеримент пов'язаний з відновленням за допомогою МНМ параметрів базисної регулярної структури з дефектами у невеликій області (441=21*21 піксель). Він побудований за аналогією з першим експериментом, де коефіцієнти $m \times n$ матриці A для регулярної структури корегувалися за формулою $A = A + 0.2 \times A(190:210, 290:310)$. Для обчислення значень функцій $f_3(x, y)$, $f_4(x, y)$ та їх субградієнтів використовуються octave-функції `fg3(x)` **та** `fg4(x)`.

Другий експеримент реалізує такий octave-код.

```
clear;                  # очищуємо пам'ять для другого тесту
global A                # A - задана m*n матриця
n = 600, m = 400,       # n = 1500, m = 1000,
# Встановлюємо параметри r-алгоритму
alpha = 2, h0 = 1.0,  q1 = 0.95,
epsx = 1.e-6, epsg = 1.e-12, maxitn = 1000, intp=100,
# Запускаємо на розрахунок по п'ять тестових задач (3) та (4)
scale = 256; rand("seed",2025); tab34 = []; ntest = 5;
for ii=1:ntest
   # Генеруємо матрицю A та параметри базисної регулярної 3D-структури
   x = scale*rand(m,1);  y = scale*rand(n,1);
   A = x*ones(1,n)+ones(m,1)*(y)';
   temp = (sum(x)-sum(y))/(n+m);
   xb = x - temp*ones(m,1);  yb = y + temp*ones(n,1);
   i1=10; # додаємо 21*21 дефект до матриці A
   A(m/2-i1:m/2+i1,n/2-i1:n/2+i1)=1.2*A(m/2-i1:m/2+i1,n/2-i1:n/2+i1);
   # Розв'язуємо задачу (3)
   x0=zeros(m+n,1); tstart=time();
   [xr3,fr3,itn3,nfg3,istop3] =
ralgb5a(@fg3,x0,alpha,h0,q1,epsg,epsx,maxitn,intp);
   temp=(sum(xr3(1:m))-sum(xr3(m+1:m+n)))/(n+m);
   xr3 = xr3 - temp*[ones(m,1);ones(n,1)];
   itn3, nfg3, istop3, fr3, dx3=norm(xr3-[xb; yb]),
   time3=time()-tstart,
```

```
   # Розв'язуємо задачу (4)
   x0=zeros(m+n,1); tstart=time();
   [xr4,fr4,itn4,nfg4,istop4] =
ralgb5a(@fg4,x0,alpha,h0,q1,epsg,epsx,maxitn,intp);
   itn4, nfg4, istop4, fr4, dx4=norm(xr4-[xb; yb]),
   time4=time()-tstart,
   tab34 = [tab34; ii itn3 nfg3 time3 dx3 fr3 itn4 nfg4 time4 dx4 fr4];
endfor
tab34,  # друкуємо результати розрахунку для другого тесту
```

Результати другого експерименту для $n=600$ і $m=400$ наведені в табл. 3, а для $n=1500$ і $m=1000$ – в табл. 4, де **itn** – кількість ітерацій r-алгоритму, **nfg** – кількість обчислень значення функції та її субградієнта за формулами (7) та (8), **time, delta** та **fr** – такі ж самі позначення, які були використані для таблиць 1 та 2.

ТАБЛИЦЯ 3. Затрати r-алгоритму для МНМ ($m=400$ і $n=600$, $q_1=0.95$)

№ п/п	Задача (3)					Задача (4)				
	itn	nfg	time	delta	fr	itn	nfg	time	delta	fr
1	374	550	5.56	6.8e-07	23377.4	374	550	5.60	7.2e-07	23377.4
2	377	555	5.56	7.5e-07	24339.1	378	556	5.60	7.5e-07	24339.1
3	375	551	5.53	7.5e-07	20420.9	376	552	5.63	7.2e-07	20420.9
4	376	552	5.53	7.7e-07	21921.2	376	552	5.59	7.4e-07	21921.2
5	374	548	5.51	8.2e-07	21439.4	375	549	5.58	7.5e-07	21439.4

ТАБЛИЦЯ 4. Затрати r-алгоритму для МНМ ($m=1000$ і $n=1500$, $q_1=0.95$)

№ п/п	Задача (3)					Задача (4)				
	itn	nfg	time	delta	fr	itn	nfg	Time	delta	fr
1	386	577	38.42	8.3e-07	24611.2	385	576	38.36	8.4e-07	24611.2
2	386	576	38.28	8.2e-07	24757.9	386	576	38.33	8.5e-07	24757.9
3	386	577	38.62	7.9e-07	21368.0	386	577	38.79	8.1e-07	21368.0
4	386	576	38.73	8.1e-07	22587.6	386	576	38.38	8.0e-07	22587.6
5	386	576	38.35	8.1e-07	23040.4	386	576	38.41	8.2e-07	23040.4

Знайдені параметри регулярних 3D-структур відхиляються від точних менше ніж на 10^{-6} за евклідовою нормою (див. колонку **delta** в табл. 3 та 4), що означає, що наближення до точок мінімуму знайдено досить точно. Для всіх десяти розрахунків знайдені мінімальні значення функцій $f_3(x,y)$ та $f_4(x,y)$ (див. колонку **fr**) співпадають з точністю до всіх шести значущих цифр. Для розв'язання задач 1 та 2 за допомогою МНМ на базі r-алгоритму достатньо менше 6 секунд, якщо $n=600$ і $m=400$, та менше 40 секунд, якщо $n=1500$ і $m=1000$. Це означає, що на сучасних персональних ЕОМ розроблені програми можна використовувати в діалоговому режимі для аналізу дефектів у регулярних зображеннях невеликих розмірів (6 секунд, GNU Octave версії 5.1.0) та середніх розмірів (40 секунд).

Висновки. Регулярні зображення з областями дефектів характерні при неруйнівному контролі якості багатошарових тонкостінних композиційних матеріалів за допомогою методів лазерної інтерферометрії, таких як метод голографічної інтерферометрії, метод спекл-інтерферометрії і метод ширографії [1]. Розроблені на основі r-алгоритму методи та їх програмні реалізації дозволяють автоматизувати процес визначення місця розташування дефектів у відповідальних елементах конструкцій і знизити вплив людського фактору при неруйнівному контролі якості. При цьому для зображень із $400 \div 1000$ пікселями по вертикалі та $600 \div 1500$ пікселями по горизонталі оптимізаційні задачі (1)–(4) можна вирішувати у діалоговому режимі, тому що для цього потрібно від 5 до 40 секунд на сучасних персональних ЕОМ з використанням GNU Octave версії 5.1.0.

Авторські внески. Жидков В.О. – методологія, дослідження, програмне забезпечення, написання, редагування; Стецюк П.І. – концептуалізація, керівництво; Хом'як О.М. – узагальнення, написання, оригінальна чернетка.

Список літератури

1. Lobanov L.M., Pivtorak V.A., Kyjanets I.V., Savitsky V.V., Tkachuk G.I. Express control of quality and stressed state of welded structures using method of electron shearography and speckle-interferometry. *The Paton Welding Journal*. August, 2005. P. 35–40.
2. Стецюк П.И., Савицкий В.В. О поиске дефектов в регулярных 3D-структурах. *Проблемы управления и информатики*. 2018. 2. С. 33–48.
3. Stetsyuk P.I., Savitsky V.V., Zhydkov V.O. Optimization Problems for Regular Image Reconstruction. Book of Absracts of International Conference on Optimization and Equilibrium Problems, Dresden, July 31 – August 2, 2019. P. 45–46.
4. Стецюк П., Савицький В., Жидков В. Пошук дефектів у регулярних 3D-структурах. Методи негладкої оптимізації в прикладних задачах / ред. П. Стецюк, М. Григорак. Київ: ЛАЗУРИТ ПОЛІГРАФ, 2023. С. 230–257.
5. NEOS Solver. https://neos-server.org/ (звернення: 17.02.2025)
6. SNOPT (Sparse Nonlinear OPTimizer) https://ccom.ucsd.edu/~optimizers/solvers/snopt/ (звернення: 17.02.2025)
7. Ipopt Documentation. https://coin-or.github.io/Ipopt/ (звернення: 17.02.2025)
8. Gurobi Optimization, Inc., Gurobi Optimizer Reference Manual, 2014. http://www.gurobi.com/ (звернення: 17.02.2025)
9. CPLEX Optimizer. High-performance mathematical programming solver for linear programming, mixed-integer programming and quadratic programming. https://www.ibm.com/analytics/cplex-optimizer (звернення: 17.02.2025)
10. GNU Octave https://octave.org/ (звернення: 17.02.2025)
11. Шор Н.З. Методы минимизации недифференцируемых функций и их приложения. Киев: Наукова думка, 1979. 200 с.
12. Shor N.Z. Nondifferentiable Optimization and Polynomial Problems. Amsterdam, Kluwer, 1998, 165 p.
13. Стецюк П.І. Комп'ютерна програма «Octave-програма ralgb5a: $r(\alpha)$-алгоритм з адаптивним кроком». Свідоцтво про реєстрацію авторського права на твір № 85010. Україна. Міністерство економічного розвитку і торгівлі. Державний департамент інтелектуальної власності. Дата реєстрації 29.01.2019.

Одержано 24.02.2025

Жидков Володимир Олександрович,
аспірант
Інституту кібернетики імені В.М. Глушкова НАН України, Київ,
lynx233@yahoo.com
https://orcid.org/0000-0003-0553-6384

Стецюк Петро Іванович,
доктор фізико-математичних наук, завідувач відділу методів негладкої оптимізації
Інституту кібернетики імені В.М. Глушкова НАН України, Київ,
stetsyukp@gmail.com
https://orcid.org/0000-0003-4036-2543

Хом'як Ольга Миколаївна,
кандидат фізико-математичних наук, старший науковий співробітник відділу математичних методів теорії надійності складних систем Інституту кібернетики імені В.М. Глушкова НАН України, Київ.
khomiak.olha@gmail.com
https://orcid.org/0000-0002-5384-9070

УДК 519.85

В.О. Жидков, П.І. Стецюк, О.М. Хом'як *

Метод найменших квадратів та метод найменших модулів для пошуку дефектів у регулярних зображеннях

Інститут кібернетики імені В.М. Глушкова НАН України, Київ
* *Листування:* *khomiak.olha@gmail.com*

Для пошуку дефектів у регулярних зображеннях (регулярних 3D-структурах, де (i, j)-коефіцієнт матриці дорівнює сумі i-го коефіцієнта рядка та j-го коефіцієнта стовпця) описано метод найменших квадратів (МНК) та метод найменших модулів (МНМ). Їм відповідають оптимізаційні задачі апроксимації коефіцієнтів заданої матриці за допомогою коефіцієнтів матриці для регулярної 3D-структури за критерієм найменших квадратів та критерієм найменших модулів. Різниця між коефіцієнтами заданої матриці та коефіцієнтами знайденої матриці визначає області дефектів для регулярної 3D-структури.

Сформульовано чотири оптимізаційні задачі для знаходження параметрів регулярних структур та наведено їх властивості. Перша та друга задачі відповідають знаходженню найкращих за критерієм найменших квадратів регулярної та базисної регулярної структур, а третя та четверта – найкращих за критерієм найменших модулів регулярної та базисної регулярної структур. Описано способи обчислення градієнтів гладких функцій (МНК) та субградієнтів негладких функцій (МНМ) для всіх чотирьох задач. Наведено коди octave-функцій, де обчислення значень функції та її (суб)градієнтів реалізовані за допомогою матрично-векторних операцій.

Наведено результати застосування r-алгоритму для оцінки часу розв'язання на сучасних ПЕОМ тестових задач для регулярних зображень невеликих розмірів – 400 пікселів по вертикалі та 600 пікселів по горизонталі, та середніх розмірів – 1000 та 1500 пікселів. Перший експеримент пов'язаний з відновленням за допомогою МНК параметрів базисної регулярної структури без дефектів, другий експеримент – з відновленням за допомогою МНМ параметрів базисної регулярної структури з дефектами у невеликій області (441 піксель). Розроблені програми можна використовувати в діалоговому режимі для аналізу дефектів у регулярних зображеннях невеликих розмірів (5 секунд) та середніх розмірів (40 секунд).

Ключові слова: регулярна 3D-структура, метод найменших квадратів, метод найменших модулів, r-алгоритм.

UDC 519.85

Volodymyr Zhydkov, Petro Stetsyuk, Olha Khomiak *

Least Squares Method and Least Modules Method for Finding Defects in Regular Images

V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine, Kyiv
* *Correspondence:* *khomiak.olha@gmail.com*

The article describes methods based on method of least squares (LSM) and the method of least modules (LMM) to search for defects in regular images (regular 3D structures images). They correspond to the optimization problems of approximating the coefficients of a given matrix using matrix coefficients for a regular 3D structure according to the least squares criterion and the smallest modulus criterion. The difference (exceeding statistical average) between the coefficients of the given matrix and the coefficients of the found matrix marks the defect regions for a regular 3D structure.

Four optimization problems for finding the parameters of regular structures are formulated and their properties are formulated for the task. The first and second problems correspond to finding the best according to the criterion of least squares of regular and basic regular structures, and the third and fourth – the best according to the criterion of smallest modules of regular and basic regular structures. Methods of calculating gradients of smooth functions (LSM) and subgradients of non-smooth functions (LMM) for all four problems are described. Codes of octave functions are provided, where the calculation of function values and its (sub)gradients is implemented using matrix-vector operations.

Also, the article provides sample test results of the application of *r*-algorithm to estimate the time of solving test problems on modern PCs for regular images of small sizes – 400 pixels vertically and 600 pixels horizontally, and medium sizes – 1000 and 1500 pixels. The first experiment is related to the restoration by means of MNC of the parameters of the basic regular structure without defects, the second experiment is related to the restoration by means of MNM of the parameters of the basic regular structure with defects in a small area (441 pixels). The developed programs can be used in dialog mode to analyze defects in regular images of small sizes (5 seconds) and medium sizes (40 seconds).

Keywords: regular 3D-structure, least squares method, least modulus method, *r*-algorithm.