

Київський національний університет імені Тараса Шевченка

**ДОВГИЙ Б.П.
ВАКАЛ Є.С.
ВАКАЛ Ю.Є.**

**Методичні вказівки
до лабораторних робіт з дисципліни
“Методи обчислень” з використанням пакета MATLAB**

Навчальне видання

для студентів механіко-математичного факультету
спеціальності “математика”

Київ –2016

УДК 681.3.06

Рецензенти:

докт. фіз.-мат. наук, проф. Ю.А. Белов

канд. фіз.-мат. наук, доц. О.С. Тригуб

*Рекомендовано до друку вченою радою механіко-математичного факультету
(протокол № ___ від _____ 201__ року)*

Довгий Б.П.

Методичні вказівки до лабораторних робіт з дисципліни “Методи обчислень” з використанням пакета MATLAB. Навчальне видання / Б.П.Довгий, Є.С.Вакал, Ю.Є.Вакал. – К.:, 2016. – 41 с.

Навчальне видання містить перелік завдань з лабораторних робіт та методичні рекомендації для їх виконання з дисципліни “Методи обчислень”, що викладається студентам механіко-математичного факультету Київського національного університету імені Тараса Шевченка. За змістом видання поділено на теми у відповідності з навчальною програмою дисципліни. Кожне завдання супроводжується короткими теоретичними відомостями, текстами програм з використанням мови системи комп’ютерної математики MATLAB, посиланнями на відповідну літературу.

Для студентів і викладачів механіко-математичного факультету, які проводять заняття з курсу “Методи обчислень”.

Навчальне видання

ДОВГИЙ Борис Павлович

ВАКАЛ Євген Сергійович

ВАКАЛ Юлія Євгенівна

**МЕТОДИЧНІ ВКАЗІВКИ ДО ЛАБОРАТОРНИХ РОБІТ З ДИСЦИПЛІНИ
“МЕТОДИ ОБЧИСЛЕНЬ” З ВИКОРИСТАННЯМ ПАКЕТА MATLAB**

© Довгий Б.П., Вакал Є.С., Вакал Ю.Є. 2016
© Київський національний університет імені Тараса Шевченка,

ПЕРЕДМОВА

Навчальне видання складено у відповідності з програмою дисципліни “Методи обчислень”, що викладається на механіко-математичному факультеті і призначено для ознайомлення студентів з темами та завданнями лабораторних робіт (ЛР) і рекомендаціями до їх виконання за допомогою математичного пакета MATLAB. Серед інших пакетів обрано MATLAB, оскільки він дозволяє студенту природно перейти у своєму навчанні від простих обчислень до можливостей програмування з використанням потужних обчислювальних і графічних засобів. Універсальні бібліотеки чисельних методів і засобів візуалізації результатів обчислень, що містяться в пакеті MATLAB, у багатьох випадках дозволяють користувачеві отримати кінцевий результат швидше, ніж при використанні традиційних мов програмування.

В системі MATLAB реалізовано класичні чисельні алгоритми розв’язання рівнянь, задач лінійної алгебри, знаходження значень визначених інтегралів, апроксимації, розв’язання окремих рівнянь і систем диференціальних рівнянь. Для застосування більшості можливостей пакета достатньо базових знань з чисельних методів.

За змістом видання поділено на теми, які містять постановки задач, вказівки до їх розв’язання, описи деяких алгоритмів і М-файлів для реалізації методів розв’язання задач чисельного аналізу за допомогою MATLAB. В кожній темі наведено посилання на необхідну літературу.

Навчальне видання може бути корисне студентам університетів при виконанні та контрольних і лабораторних робіт з обчислювальної математики, інформатики та інших дисциплін.

ЛАБОРАТОРНА РОБОТА №1
ІТЕРАЦІЙНІ МЕТОДИ РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ
АЛГЕБРАІЧНИХ РІВНЯНЬ

Постановка задачі. Розв'язати систему лінійних алгебраїчних рівнянь (СЛАР)

$$\begin{pmatrix} 2+k & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 5 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 7 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 6+g & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 8-g & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 5 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 4 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 9 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2+d \end{pmatrix} \vec{x} = \vec{b}$$

за допомогою *оптимального лінійного ітераційного процесу*.

Тут через k позначено номер за списком студента в групі, g – номер групи, d – день народження студента.

Вектор \vec{b} – задано (породжується множенням матриці A на обраний студентом контрольний вектор \vec{x}).

Методичні вказівки до розв'язання. Подамо метод простої ітерації в канонічній формі однокрокового ітераційного процесу (І П)

$$\frac{\vec{x}^{(k+1)} - \vec{x}^{(k)}}{\tau} + A\vec{x}^{(k)} = \vec{f}. \quad (1.1)$$

Для методу (1.1) відомо оптимальне значення ітераційного параметру у випадку коли $A=A'$, $A>0$. Такий І П має назву оптимального лінійного ітераційного процесу [1, с. 111]. Значення τ_{opt} визначається наступним способом

$$\tau_{opt} = \frac{2}{\lambda_{\min}(A) + \lambda_{\max}(A)}. \quad (1.2)$$

При цьому кількість кроків І П визначається як

$$n \geq n_0(\varepsilon), \quad n_0(\varepsilon) = \frac{1}{2} \sqrt{\frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}} \ln\left(\frac{2}{\varepsilon}\right),$$

де $\varepsilon > 0$ – потрібна точність розрахунків. Даний алгоритм реалізуємо у вигляді М-функції `m_1mpi` в системі MATLAB 7.6.0 (R2008a).

Для знаходження максимального значення власного числа матриці A використаємо *ітераційний степеневий метод* [2, с. 55], реалізований М-функцією `maxlam_h` в системі MATLAB [3, с. 40].

Для знаходження мінімального значення власного числа матриці A скористаємося відомим співвідношенням між власними числами матриці A і оберненої матриці A^{-1} :

$$\lambda_{\min}(A) = \frac{1}{\lambda_{\max}(A^{-1})},$$

що дає можливість знову застосувати ітераційний степеневий метод у формі $A\bar{x}^{(k+1)} = \bar{x}^{(k)}$ до матриці A^{-1} . Для розв'язання множини СЛАР (при змінній правій частині) використовуємо розклад матриці A на дві трикутні матриці L, U викликом стандартної функції MATLAB з іменем *lu* [3, стор. 20]. Даний алгоритм реалізуємо у вигляді М-функції `minlamh`.

Зауваження. Тут і надалі у фрагментах програм рядки, виділені курсивом, вказують на необхідність написання студентами відповідної частини алгоритму замість символу "...". Вказаним символом позначено місця, куди потрібно вставити відповідне числове значення, ім'я змінної, текст програми тощо.

Нижче наведено опис алгоритму і деякі М-файли для реалізації ЛР.

```
% lr1_main.m
% Лабораторна робота № 1 з МО
% Лінійний оптимальний ІП

clear all
clc

% ініціалізація параметрів задачі
...
% завдання матриці A
...
% завдання контрольного вектора x
...
% формування вектора b правої частини СЛАР
...

% завдання e - точності степеневого ІП
% h1 - вектор-стовпця початкового наближення
% i знаходження мінімального, максимального власного значення матриці A
e=1e-10;
h1=...;
[g1,h1]=minlamh(A,e,h1);
h1=...;
[g2,h1]=maxlam_h(A,e,h1);

[g1 g2]

% завдання e - точності лінійного оптимального ІП
% X - вектора початкового наближення
% i знаходження розв'язку СЛАР
e=1e-5; X=b;
[X,N]=m_lmpi(A,b,e,X,g1,g2);

N
X

function [l,x]=maxlam_h(A,e,x)
% Знаходження з точністю e максимального власного значення l
% i відповідного власного вектора x для квадратної матриці A
```

```

% ітераційно-степеневим методом
% Вхідні дані:
%     А- квадратна матриця;
%     е- точність;
%     х- вектор початкового наближення.
% Вихідні дані:
%     l- власне число;
%     х- власний вектор.
n=length(A); b=0; l=1; y=zeros(n,1);
while abs(b-l)>e
    b=l; l=0;
    for i=1:n
        yy=sum(A(i,:)*x(:));
        if abs(yy)>abs(l)
            l=yy;
        end
        y(i)=yy;
    end
    x=y./l;
end
b=norm(x); x=x./b;
return
end

```

function [l,x]=minlamh(A,e,x)

```

% Знаходження з точністю е мінімального власного значення l
% і відповідного власного вектора x для квадратної матриці А
% ітераційно-степеневим методом
% Вхідні дані:
%     А- квадратна матриця;
%     е- точність;
%     х- вектор початкового наближення.
% Вихідні дані:
%     l- власне число;
%     х- власний вектор.
n=length(A);
[L,U,P]=lu(A);
b=0; l=1;
while abs(b-l)>e
    b=l;
    % розв'язання СЛАР Lu=x (L - нижня трикутна)
    ...
    l=0;
    for i=n:-1:1
        s=y(i);
        for j=i+1:n
            s=s-U(i,j).*x(j);
        end
        s=s./U(i,i); x(i)=s;
        if abs(s)>abs(l)
            l=s;
        end
    end
    x=x./l; l=1./l;
end
b=norm(x); x=x./b;
return
end

```

function [x,N]=m_lmpi(A,f,e,x,g1,g2)

```

% Знаходження з точністю е наближеного розв'язку
% СЛАР A*x=f методом простої ітерації з оптимальним параметром
% Вхідні дані:

```

```
%      A- квадратна матриця;
%      f- відомий вектор (права частина);
%      e- точність ( 0<e<1 );
%      x- початкове наближення до розв'язку;
%      g1- нижня межа спектру матриці A;
%      g2- верхня межа спектру матриці A:
%          A=A'>0 ,  g1*E <= A <= g2*E.
% Вихідні дані:
%      x- розв'язок;
%      N- кількість виконаних ітерацій.
ksi=g1./g2;
t=2./(g1+g2);
N=fix(log(1./e)./(2.*ksi)+0.6);
for k=1:N
% реалізація одного кроку ІП (1.1)
...
end
return
end
```

ЛАБОРАТОРНА РОБОТА №2 МЕТОДИ РОЗВ'ЯЗАННЯ СИСТЕМ НЕЛІНІЙНИХ РІВНЯНЬ

Постановка задачі. Розв'язати нелінійну систему рівнянь

$$\begin{cases} \phi(y_1) = \gamma_1, \\ \frac{y_{i-1} - 2y_i + y_{i+1}}{h^2} - p(x_i)y_i = -f(x_i, y_i), \quad i = \overline{2, n-1}, \\ \psi(y_n) = \gamma_2, \end{cases}$$

використовуючи *ітераційний метод Ньютона*, а для відповідної СЛАР – *метод немонотонної прогонки*.

Тут введено позначення:

$$a = 0, \quad b = g;$$

$$x_i = a + (i-1)h, \quad i = \overline{1, n}; \quad h = \frac{b-a}{n-1}, \quad n = 11;$$

$$\bar{y} = (y_i)_{i=1}^n;$$

$$\varphi(t) = t^g + \cos(t+k), \quad \gamma_1 = 1, \quad \psi(t) = t^3 + \frac{\cos(t-k)^2}{k}, \quad \gamma_2 = d,$$

$$p(t) = (t-a)^g(b-t), \quad f(t, v) = \exp(-t+v) - tv;$$

k – номер за списком студента в групі;

g – номер групи;

d – день народження студента.

Методичні вказівки до розв'язання. Враховуючи специфіку даної нелінійної системи, можна запропонувати такий ефективний алгоритм розв'язання задачі:

а) спочатку знаходимо y_1 , а потім y_n , використовуючи метод Ньютона для одновимірного випадку [1, с. 193] реалізований М-функцією `njut1` в системі MATLAB. Для визначення початкового наближення можна побудувати графік відповідної функції і ввести інформацію в програму в діалоговому режимі.

б) враховуючи, що кожне з y_i входить нелінійно в одне рівняння, то його можна лінеаризувати, а для відповідної СЛАР використати метод немонотонної прогонки [4, с. 93], реалізований М-функцією `m_progn` в системі MATLAB [3, с. 44]. Для завдання початкового наближення $y_i^{(0)}$, $i = \overline{2, n-1}$ можна виконати лінійну інтерполяцію y_i між точками $A(a, y_1)$ і $B(b, y_n)$.

Нижче наведено опис алгоритму і деякі М-файли для реалізації ЛР.

```
% lr2_main.m
% Лабораторна робота № 2 з МО
% Розв'язання нелінійної системи рівнянь (НСР)
% з використанням методу Ньютона,
% а для лінеаризованої СЛАР – методу немонотонної прогонки
```

```
clear all
clc
```



```

% ініціалізація параметрів задачі
...
a=0; b=g; n=101;
n1=n-1; h=(b-a)./n1;

% опис функції fim(y(1))= (φ(y(1))-g1) = 0
% і її похідної для знаходження кореня y(1)=g1r
g1=1;
fim=@(t) (t.^g+cos(t+k)-g1);
dfim=@(t) (g.*t.^(g-1)-sin(t+k));

% опис функції psim(y(n))= (ψ(y(n))-g2) = 0
% і її похідної для знаходження кореня y(n)=g2r
g2=d;
psim=@(t) (t.^3+(t-k).^2./k-g2);
dpsim=@(t) (3.*t.^2+2.*(t-k)./k);

% опис функції p(x)
p=@(t) ((t-a).^g.*(b-t));

% опис функції правої частини f(t,v) і її похідної (по v)
f=@(t,v) (exp(-t+v)-t.*v);
dfv=@(t,v) (exp(-t+v)-t);

ep=1e-8;
mk=1000;
% знаходження кореня y(1)=g1r ітераційним методом Ньютона
x=[-5:0.05:5];
y=fim(x);
plot(x,y);
grid on
z=input('Введіть наближення до кореня в точці a= ');
[g1r,i]=njut1(z,ep,mk,fim,dfim);

% знаходження кореня y(n)=g2r ітераційним методом Ньютона
y=psim(x);
plot(x,y);
grid on
z=input('Введіть наближення до кореня в точці b= ');
[g2r,i]=njut1(z,ep,mk,psim,dpsim);

% розраховуємо початкове наближення розв'язку
x=[a:h:b];
z=(g2r-g1r)./(b-a);
y=g1r+z.*(x-a);

% формування незмінних (незалежних від номера ітерації s)
% коефіцієнтів СЛАР (A- піддіагональ, B- наддіагональ)
hh=h.*h;
A(1)=0; B(1)=0;
for i=2:n1
    A(i)=1; B(i)=1;
end
A(n)=0; B(n)=0;
C(1)=1; F(1)=0;
C(n)=1; F(n)=0;

% ітераційний процес (ІП) лінеаризації (метод Ньютона) для НСР
s=0; flag=0;
while ~flag & s<mk
    s=s+1;
    flag=1;
% формування C - діагоналі і F - правої частини СЛАР, на черговій ітерації s

```

```

for i=2:n1
    xi=x(i); yi=y(i);
    C(i)= ...
    F(i)= ...
end

% розв'язування СЛАР методом немонотонної прогонки (D - поправка);
% знаходження нового наближення у розв'язку НСР;
% обчислення умови flag закінчення ІП по s
[D AL]=m_progn(A,C,B,F);
for i=2:n1
    z=D(i); y(i)=y(i)+z;
    if flag
        flag=abs(z)<ep;
    end
end
end

end

[g k d s]
plot(x,y)
grid on

```

function [x,k]=njut1(x,e,mki,ffun,dffun)

```

% Знаходження кореня нелінійного рівняння  $f(x) = 0$ 
% ітераційним методом Ньютона  $x(n+1)=x(n)-f(x(n))/f'(x(n))$ ,
% де значення  $x(0)$  відоме.
% Умова закінчення ітерацій :  $abs(x(n+1)-x(n))<e$ .
% Вхідні дані:
% x - початкове наближення до кореня;
% e - точність ( $e>0$ );
% mki - максимальна кількість кроків ітерацій;
% ffun - функція  $f(x)$ ;
% dffun - функція  $f'(x)$ .
% Вихідні дані:
% x - фінішне наближення до кореня;
% k - кількість виконаних ітерацій.
if nargin<5
    error('NJUT1:Not Enough Inputs',...
        'Not enough input arguments. See NJUT1.');
```

```

    return
end
if e<=0
    error('NJUT1:error e<=0.',...
        'e<=0. See NJUT1.');
```

```

    return
end
k=0; F=1;
while F & k<mki
    k=k+1;
    o=ffun(x)./dffun(x);
    x=x-o;
    F=abs(o)>e;
end
return
end

```

function [y,alfa]=m_progn(a,c,b,f)

```

% Метод немонотонної прогонки для розв'язання СЛАР
% з тридіагональною матрицею
% метод Гауса з вибором провідного елемента по рядку)
%  $a(k)*y(k-1) - c(k)*y(k) + b(k)*y(k+1) = - f(k)$ 
%  $k= 1,2,\dots,n; a(1)=b(n)=0$ 

```

```

% Вхідні дані:
%   a - піддіагональ;
%   -c - діагональ матриці;
%   b - наддіагональ;
%   -f - права частина.
% Вихідні дані:
%   y - розв'язок;
% alfa - прогоночні коефіцієнти.
n=length(c); n1=n-1;
% пряма прогонка:
alfa(1)=0; beta(1)=0;
C=c(1); A=a(2); F=f(1); Q=f(2);
ka(1)=1; te(1)=0;
for k=1:n1
    j=k+1;
    if abs(C)>=abs(b(k))
        alfa(j)=b(k)./C; beta(j)=F./C;
        C=c(j)-A.*alfa(j); F=Q+A.*beta(j);
        te(j)=ka(k); ka(j)=j;
        if k~=n1
            j=j+1; A=a(j); Q=f(j);
        end
    else
        alfa(j)=C./b(k); beta(j)=-F./b(k);
        C=c(j).*alfa(j)-A; F=Q-c(j).*beta(j);
        te(j)=j; ka(j)=ka(k);
        if k~=n1
            j1=j; j=j+1;
            A=a(j).*alfa(j1); Q=f(j)+a(j).*beta(j1);
        end
    end
end
end
% зворотна прогонка:
y(ka(n))=F./C;
for k=n1:-1:1
    j=k+1;
    y(te(j))=alfa(j).*y(ka(j))+beta(j);
end
return
end

```

ЛАБОРАТОРНА РОБОТА №3

НАБЛИЖЕННЯ ЛІНІЙНОГО ДИФЕРЕНЦІАЛЬНОГО ОПЕРАТОРА МЕТОДОМ НЕВИЗНАЧЕНИХ КОЕФІЦІЄНТІВ

Постановка задачі. На відрізку $[a, b]$ методом невизначених коефіцієнтів Коллатца) побудувати в точці $x_i \in \{x_j\}_{j=1}^{r+1}$ апроксимацію лінійного диференціального оператора

$$Lu(x) = u(x) + du^{(2)}(x) + ku^{(4)}(x) + u^{(m)}(x)$$

з порядком $q = r + 1 - m \equiv g$.

Тут введено позначення:

$$a = 0, b = 1;$$

$$x_j = a + (j-1)h, \quad h = \frac{b-a}{r}, \quad j = \overline{1, r+1};$$

$$m = g + 4, \quad i = \text{mod}(d, r+1);$$

k – номер за списком студента в групі;

g – номер групи;

d – день народження студента.

Методічні вказівки до розв'язання. Метод невизначених коефіцієнтів, або, інакше, метод Коллатца, докладно описано в [2, с. 115–118]. Звернімо увагу на те, що в алгоритмі потрібно:

- правильно вибрати значення r, q, m ,
- належним чином сформулювати матрицю і праву частину СЛАР.

Нижче наведено опис алгоритму і деякі М-файли для реалізації ЛР.

% lr3_main.m

% Лабораторна робота № 3 з МО

% Наближення лінійного диференціального оператора $(m-1)$ -го порядку

% різницеvim оператором по $(r-1)$ точці з рівномірним кроком з

% порядком апроксимації $q=r+1-m$ за методом Коллатца

% (метод невизначених коефіцієнтів)

```
clear all
```

```
clc
```

```
% ініціалізація параметрів задачі
```

```
...
```

```
a=0; b=1; r1=8;
```

```
r=r1-1; h=(b-a)./r; X=[a:h:b];
```

```
m=g+4;
```

```
i=mod(d, r1);
```

```
A=zeros(1, m+1);
```

```
A(1)=1; A(3)=d; A(5)=k; A(m+1)=1;
```

```
[C, q]=collatz(A, X, i);
```

```
q
```

```
C
```

```

function [C,q]=collatz(A,X,i)
% Наближення в точці x(i) лінійного диференціального оператора (m-1)-го порядку
% різницевим оператором по (r-1) точці з рівномірним кроком з
% порядком апроксимації q=r+1-m за методом Колатца (невизначених коефіцієнтів)
% Вхідні дані :
%   A - коефіцієнти лінійного диференціального оператора (m-1)-го порядку;
%   X - значення вузлів сітки;
%   i - індекс точки x(i), в якій будується наближення.
% Вихідні дані :
%   C - коефіцієнти різницевого оператора;
%   q - порядком апроксимації (q=r+1-m).

ra=length(A); m=ra-1;
rx=length(X); r=rx-1;
h=X(2)-X(1);

for k=1:rx
    AL(k)=k-i;
end

% формування коефіцієнтів СЛАР (матриці МА і правої частини В)
...
V=B';

q=r+1-m;
% розв'язування СЛАР
C=MA\V;
end

```

ЛАБОРАТОРНА РОБОТА №4 ПОБУДОВА ІНТЕРПОЛЯЦІЙНИХ КВАДРАТУРНИХ ФОРМУЛ НАЙВИЩОГО ПОРЯДКА ТОЧНОСТІ

Постановка задачі. За допомогою побудованої *квадратурної формули (КФ) найвищого алгебраїчного степеня точності*

$$I = \int_a^b \rho(x)f(x)dx \approx I_n = \sum_{k=1}^n A_k f(x_k)$$

обчислити визначений інтеграл

$$I = \int_{\alpha}^{\beta} \rho(x)U(x)dx.$$

Тут введено позначення:

$\rho(x)$	$[a, b]$	n	$[\alpha, \beta]$	$U(x)$
$1 + x^g$	$[0, 1]$	$\min(g + 6, \max(d, k))$	$[\min(k, d), \max(k, d)]$	$\exp\left(\frac{-x^g}{k+x}\right)$

k – номер за списком студента в групі;

g – номер групи;

d – день народження студента.

Методичні вказівки до розв'язання. Побудова квадратурних формул найвищого алгебраїчного степеня точності, докладно описана в [2, с. 139–144]. Звернімо увагу на те, що в алгоритмі потрібно:

- розрахунок моментів потрібно здійснювати в загальній постановці:
 - а) не використовувати відносні моменти,
 - б) інтеграли розраховувати чисельно, використовуючи, наприклад, метод Сімпсона, реалізований М-функцією `i_simp` в системі MATLAB;
- корені многочлена $\omega(x)$ можна знайти, використовуючи вбудовану М-функцію `roots` системи MATLAB;
- розрахунок коефіцієнтів A_k здійснювати чисельно, використовуючи метод Сімпсона;
- для обчислення інтеграла з функцією $U(x)$ необхідно виконати заміну змінних, використовуючи М-функцію `preob`.

Нижче наведено опис алгоритму і деякі М-файли для реалізації ЛР.

```
% lr4_main.m
% Лабораторна робота № 4 з МО
% Побудова інтерполяційних квадратурних формул (КФ)
% найвищого алгебраїчного степеня точності

clear all
clc

% блок вхідних даних задачі
```

```

global g
% ініціалізація параметрів задачі
...
n=9;

% знаходимо вузли і коефіцієнти КФ
[x,A]=kvnast(a,b,n);

x
A

% обчислення
u=@(t)(exp(-t.^g./(k+t)));
al=min(k,d); be=max(k,d);
% al=a; be=b;

[ko,bs,xx,AA]=preob(al,be,a,b,x,A);
upr=@(t)(ko.*u(ko.*t+bs).*kv_po(ko.*t+bs)./kv_po(t));

s=dot(A,upr(x));
s

% перевірка
f=@(t)(kv_po(t).*u(t));
ss=i_simp(al,be,f,1e-10,1000);
ss
s-ss

```

function [x,A]=kvnast(a,b,n)

```

% Побудова інтерполяційних квадратурних формул (КФ)
% найвищого алгебраїчного степеня точності
% Вхідні дані :
% a - нижня межа визначеного інтегралу;
% b - верхня межа;
% n - кількість вузлів КФ.
% Вихідні дані :
% x - значення вузлів КФ;
% A - коефіцієнти КФ.

```

```
n1=n-1;
```

```

global N ii M
N=n;
e=1e-10; mki=1000;
% формуємо моменти
M=zeros(1,2*n);
for i=1:2.*n
    ii=i-1;
    M(i)= ...
end

```

```

% формуємо СЛАР
A=zeros(n); P=zeros(n,1);
for i=1:n
    for j=1:n
        A(i,j)=M(n+i-j);
    end
    P(i)=-M(n+i);
end

```

```

% розв'язуємо СЛАР
M=A\P;

```



```
% Вихідні дані:  
%     x - квадратурні вузли  
%     c - квадратурні коефіцієнти  
  
% знаходження q,r,x,c  
...  
return  
  
function y=kv_fc(t)  
global N ii M  
y=kv_po(t).*polyval(M,t);  
end  
  
function y=kv_fm(t)  
global N ii M  
y=kv_po(t).*t.^ii;  
end  
  
function y=kv_po(t)  
% Завдання вагової функції інтеграла  
global g  
y=1+t.^g;  
end
```

ЛАБОРАТОРНА РОБОТА №5
МЕТОД ПРЯМИХ РОЗВ'ЯЗАННЯ КРАЙОВИХ ЗАДАЧ

Постановка задачі. Застосувати *метод Рунге (прямих)* для розв'язання крайових задач для нестационарних рівнянь та систем рівнянь різних типів:

k	Постановка задачі	k	Постановка задачі
1	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = g\sqrt{U}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} = -\frac{1}{d} \frac{\partial V(x,t)}{\partial x} + \exp(-Ux-Vt), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = x^2(1-x), 0 \leq x \leq 1; \\ V(0,t) = t/g, 0 \leq t < T. \end{cases}$	2	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = U/(x+tV^2), \\ \frac{\partial V(x,t)}{\partial t} = \frac{1}{gd} \frac{\partial V(x,t)}{\partial x} + \sin(-UV), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = x^2(1-x), \\ 0 \leq x \leq 1; \\ V(1,t) = t/g, 0 \leq t < T. \end{cases}$
3	$\begin{cases} \frac{\partial U(x,t)}{\partial t} + \frac{1}{d} \frac{\partial U(x,t)}{\partial x} = \sqrt{U}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} - \frac{1}{d} \frac{\partial V(x,t)}{\partial x} = gVt + \exp(-Ux), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = \sin(\pi x), 0 \leq x \leq 1; \\ U(0,t) = t/g, V(1,t) = \sqrt{t}, 0 \leq t < T. \end{cases}$	4	$\begin{cases} \frac{\partial U(x,t)}{\partial t} + \frac{U}{d} \frac{\partial U(x,t)}{\partial x} = xt, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = \sin(\pi x), 0 \leq x \leq 1 \\ U(0,t) = t\sqrt{t/g}, 0 \leq t < T. \end{cases}$
5	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = U/(xtV^2), \\ \frac{\partial V(x,t)}{\partial t} = \frac{1}{d} \frac{\partial^2 V(x,t)}{\partial x^2} + (x+t)/(UV), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = x^g(1-x), 0 \leq x \leq 1; \\ V(0,t) = \sin(\pi t), V(1,t) = t/g, 0 \leq t < T. \end{cases}$	6	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = \sqrt{U}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + dU, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = g \sin(\pi x), \\ 0 \leq x \leq 1; \\ \frac{\partial V(0,t)}{\partial x} = 0, V(1,t) = t, 0 \leq t < T. \end{cases}$
7	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = \sqrt{VU}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + dU, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = g \sin(\pi x), 0 \leq x \leq 1; \\ V(0,t) = t/g, \frac{\partial V(1,t)}{\partial x} = 0, 0 \leq t < T. \end{cases}$	8	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = \sqrt{U}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} = d \frac{\partial^2 V(x,t)}{\partial x^2}, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = g \sin(\pi x), \\ 0 \leq x \leq 1; \\ \frac{\partial V(0,t)}{\partial x} + V(0,t) = 0, V(1,t) = 0, \\ 0 \leq t < T. \end{cases}$

9	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = \sqrt{g+U}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} = d \frac{\partial^2 V(x,t)}{\partial x^2}, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = \sin(\pi x), 0 \leq x \leq 1; \\ V(0,t) = 0, \frac{\partial V(1,t)}{\partial x} - V(1,t) = 0, 0 \leq t < T. \end{cases}$	10	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = \sqrt{U}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} = d \frac{\partial^2 V(x,t)}{\partial x^2} + U, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = g \sin(\pi x), \\ 0 \leq x \leq 1; \\ \frac{\partial V(0,t)}{\partial x} + V(0,t) = 0, \frac{\partial V(1,t)}{\partial x} = 0, \\ 0 \leq t < T. \end{cases}$
11	$\begin{cases} \frac{\partial U(x,t)}{\partial t} = \sqrt{VU}/(xt+V^2), \\ \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + dU, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = g \sin(\pi x), 0 \leq x \leq 1; \\ \frac{\partial V(0,t)}{\partial x} = 0, \frac{\partial V(1,t)}{\partial x} + V(1,t) = 0, 0 \leq t < T. \end{cases}$	12	$\begin{cases} \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + d/(g+V^2), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = gx \sin(\pi x), 0 \leq x \leq 1; \\ \frac{\partial V(0,t)}{\partial x} = 0, \frac{\partial V(1,t)}{\partial x} + V(1,t) = 0, \\ 0 \leq t < T. \end{cases}$
13	$\begin{cases} \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + gV, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = \sin(\pi x), 0 \leq x \leq 1; \\ \frac{\partial V(0,t)}{\partial x} + \frac{1}{d}V(1,t) = 0, V(1,t) = t, 0 \leq t < T. \end{cases}$	14	$\begin{cases} \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + gV, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = \sin(\pi x), 0 \leq x \leq 1; \\ \frac{\partial V(0,t)}{\partial x} + V(0.5,t) = 0, V(1,t) = 0, \\ 0 \leq t < T. \end{cases}$
15	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{1}{d} \frac{\partial^2 U(x,t)}{\partial x^2} + (x+t)/(U+g), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), \frac{\partial U(x,0)}{\partial t} = 0, 0 \leq x \leq 1; \\ U(0,t) = \sin(\pi t), U(1,t) = t, 0 \leq t < T. \end{cases}$	16	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{1}{d} \frac{\partial^2 U(x,t)}{\partial x^2} + 1/(U^2+g), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = \sin(\pi t), \frac{\partial U(x,0)}{\partial t} = 0, 0 \leq x \leq 1; \\ \frac{\partial U(0,t)}{\partial x} = 0, U(1,t) = t/g, 0 \leq t < T. \end{cases}$
17	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{1}{d} \frac{\partial^2 U(x,t)}{\partial x^2} + \frac{\partial U(x,t)}{\partial x}, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), \frac{\partial U(x,0)}{\partial t} = 0, 0 \leq x \leq 1; \\ U(0,t) = \sin(\pi t), U(1,t) = t/g, 0 \leq t < T. \end{cases}$	18	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{\partial^2 U(x,t)}{\partial x^2} + \frac{1}{d} \frac{\partial U(x,t)}{\partial x} + U, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = \sin(\pi x), \frac{\partial U(x,0)}{\partial t} = 0, 0 \leq x \leq 1; \\ U(0,t) = 0, U(1,t) = gt, 0 \leq t < T. \end{cases}$

19	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{1}{d} \frac{\partial^2 U(x,t)}{\partial x^2} + gU, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = \sin(\pi t), \frac{\partial U(x,0)}{\partial t} = 0, 0 \leq x \leq 1; \\ \frac{\partial U(0,t)}{\partial x} = 0, \frac{\partial U(1,t)}{\partial x} = t, 0 \leq t < T. \end{cases}$	20	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{1}{d} \frac{\partial^2 U(x,t)}{\partial x^2}, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = \sin(\pi t), \frac{\partial U(x,0)}{\partial t} = 0, 0 \leq x \leq 1; \\ \frac{\partial U(0,t)}{\partial x} + U(0,t) = 0, U(1,t) = t, \\ 0 \leq t < T. \end{cases}$
21	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{1}{d} \frac{\partial^2 U(x,t)}{\partial x^2}, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = 0, \frac{\partial U(x,0)}{\partial t} = \sin(\pi t), 0 \leq x \leq 1; \\ \frac{\partial U(0,t)}{\partial x} + U(0,t) = 0, \frac{\partial U(1,t)}{\partial x} = t, 0 \leq t < T. \end{cases}$	22	$\begin{cases} \frac{\partial^2 U(x,t)}{\partial t^2} = \frac{\partial^2 U(x,t)}{\partial x^2} + g \frac{\partial U(x,t)}{\partial x}, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = 0, \frac{\partial U(x,0)}{\partial t} = 0, 0 \leq x \leq 1; \\ U(0,t) = t, U(1,t) = t, 0 \leq t < T. \end{cases}$
23	$\begin{cases} \frac{\partial U(x,t)}{\partial t} + \frac{\partial U(x,t)}{\partial x} + gV = 1/(x+t), \\ \frac{\partial V(x,t)}{\partial t} = d \frac{\partial^2 V(x,t)}{\partial x^2} + U, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = g \sin(\pi x), 0 \leq x \leq 1; \\ U(0,t) = t(1-t), \\ V(0,t) = 0, \frac{\partial V(1,t)}{\partial x} = 0, 0 \leq t < T. \end{cases}$	24	$\begin{cases} \frac{\partial U(x,t)}{\partial t} + \frac{\partial U(x,t)}{\partial x} + U = g/(xV+t), \\ \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + dU, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = x(1-x), V(x,0) = g \sin(\pi x), \\ 0 \leq x \leq 1; \\ U(0,t) = t, \frac{\partial V(0,t)}{\partial x} = 0, V(1,t) = 0, \\ 0 \leq t < T. \end{cases}$
25	$\begin{cases} \frac{\partial U(x,t)}{\partial t} + \frac{\partial U(x,t)}{\partial x} = 1/(x+t), \\ \frac{\partial V(x,t)}{\partial t} = d \frac{\partial^2 V(x,t)}{\partial x^2} + UV, \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = 0, V(x,0) = 0, 0 \leq x \leq 1; \\ U(0,t) = t(1-t), \\ \frac{\partial V(0,t)}{\partial x} - V(0,t) = 0, \frac{\partial V(1,t)}{\partial x} = 0, 0 \leq t < T. \end{cases}$	26	$\begin{cases} \frac{\partial U(x,t)}{\partial t} + \frac{\partial U(x,t)}{\partial x} + gU/(xt+V) = 0, \\ \frac{\partial V(x,t)}{\partial t} = \frac{\partial^2 V(x,t)}{\partial x^2} + d/(1+U^2), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = 0, V(x,0) = 0, 0 \leq x \leq 1; \\ U(0,t) = t, \\ V(0,t) = 0, V(1,t) = 0, 0 \leq t < T. \end{cases}$

Тут введено позначення :

k – номер за списком студента в групі;

g – номер групи;

d – день народження студента.

Методичні вказівки до розв'язання. Метод прямих (Роте) належить до групи наближених методів розв'язання крайових задач, основна ідея яких пов'язана з пониженням розмірності поставленої задачі [5, с. 243]. У цьому методі розмірність понижується за рахунок апроксимації тільки певної частини диференціального оператора, що приводить до системи звичайних

диференціальних рівнянь. Такий підхід досить часто застосовується до систем, що складаються з рівнянь математичної фізики різних типів, особливо у випадку нелінійних рівнянь.

Розглянемо наступну модельну задачу

$$\begin{cases} \frac{\partial U(x,t)}{\partial t} + \frac{1}{d} \frac{\partial U(x,t)}{\partial x} + gV = \pi \cos(\pi x - t), \\ \frac{\partial V(x,t)}{\partial t} - \frac{1}{d} \frac{\partial V(x,t)}{\partial x} - kU = \pi \sin(\pi x - t), \\ 0 < x < 1, 0 < t \leq T; \\ U(x,0) = u_0(x) \equiv \sin(\pi x), V(x,0) = v_0(x) \equiv \cos(\pi x), 0 \leq x \leq 1; \\ U(0,t) = \mu_1(t) \equiv \sin(-t), V(1,t) = \mu_2(t) \equiv \cos(\pi - t), 0 \leq t < T, \end{cases} \quad (5.1)$$

яка при значеннях $g = 1$, $k = 1$, $d = 1$ має наступний розв'язок

$$U(x,t) = \sin(\pi x - t), \quad V(x,t) = \cos(\pi x - t).$$

Застосуємо метод Рунге по змінній x . Введемо рівномірну сітку

$$\bar{\omega}_h = \{x_i = (i-1)h, i = 1, 2, \dots, N, h = 1/(N-1)\}$$

і позначимо на прямих $x = x_i$ невідомі функції $U(x_i, t) = U_i(t)$, $V(x_i, t) = V_i(t)$. Виконаємо в (5.1) різницеву апроксимацію всіх диференціальних операторів, що містять x .

В результаті отримаємо наступну систему :

$$\begin{cases} \frac{dU_i(t)}{dt} + \frac{1}{d} \frac{U_i(t) - U_{i-1}(t)}{h} + gV_i(t) = \pi \cos(\pi x_i - t), \\ i = 2, 3, \dots, N; \\ \frac{dV_i(t)}{dt} - \frac{1}{d} \frac{V_{i+1}(t) - V_i(t)}{h} - kU_i(t) = \pi \sin(\pi x_i - t), \\ i = 1, 2, \dots, N-1; \\ U_i(0) = \sin(\pi x_i), V_i(0) = \cos(\pi x_i), i = 1, 2, \dots, N; \\ U_1(t) = \sin(-t), V_N(t) = \cos(\pi - t). \end{cases} \quad (5.2)$$

Для застосування вбудованої функції MATLAB *ode45* або *ode23s* приведемо систему (5.2) до стандартної форми запису. Враховуючи крайові умови (5.1), використовуючи позначення

$$\begin{aligned} \bar{Y}(t) &= \{U_1(t), U_2(t), \dots, U_{N-1}(t), U_N(t), V_1(t), V_2(t), \dots, V_{N-1}(t), V_N(t)\} = \\ &= \{Y_1(t), Y_2(t), \dots, Y_{N-1}(t), Y_N(t), Y_{N+1}(t), Y_{N+2}(t), \dots, Y_{2N-1}(t), Y_{2N}(t)\} = \\ &= \mu_1(t); \{y_1(t), \dots, y_{N-2}(t), y_{N-1}(t), y_N(t), y_{N+1}(t), \dots, y_{2N-2}(t)\}; \mu_2(t), \end{aligned} \quad (5.3)$$

отримаємо таку задачу Коші для звичайних диференціальних рівнянь

$$\begin{cases} \frac{d\bar{y}(t)}{dt} = \bar{f}(t, \bar{y}(t)), \\ \bar{y}(0) = \bar{y}_0. \end{cases}$$

Тут введено позначення:

$$D = 1/(dh);$$

$$\bar{\mathbf{f}}(t, \bar{\mathbf{y}}(t)) = \{f_j\}_{j=1}^{2N-2} = \begin{cases} f_1 & \equiv -D(y_1(t) - \mu_1(t)) - gy_{N-1}(t) + \pi \cos(\pi x_2 - t), & i = 1; \\ f_i & \equiv -D(y_i(t) - y_{i-1}(t)) - gy_{N-i}(t) + \pi \cos(\pi x_{i+1} - t), & i = 2, 3, \dots, N-2; \\ f_{N-1} & \equiv -D(y_{N-1}(t) - y_{N-2}(t)) - g\mu_2(t) + \pi \cos(\pi x_N - t), & i = N-1; \\ f_N & \equiv D(y_{N+1}(t) - y_N(t)) + k\mu_1(t) + \pi \sin(\pi x_1 - t), & i = 1; \\ f_{N-1+i} & \equiv D(y_{N+i}(t) - y_{N-1+i}(t)) + ky_{i-1}(t) + \pi \sin(\pi x_i - t), & i = 2, 3, \dots, N-2; \\ f_{2N-2} & \equiv D(\mu_2(t) - y_{2N-2}(t)) + ky_{N-2}(t) + \pi \sin(\pi x_{N-1} - t), & i = N-1; \end{cases}$$

$$\bar{\mathbf{y}}_0 = \{u0(x_1), u0(x_1), \dots, u0(x_{N-1}), u0(x_N), v0(x_1), v0(x_1), \dots, v0(x_{N-1}), v0(x_N)\}.$$

Зауваження.

1. При наявності в крайових умовах похідних, рекомендуємо використовувати метод “законтурних точок”.
2. Перехід в позначеннях (5.3) треба виписувати окремо і дуже уважно перевіряти відповідність за індексами.

Для модельної задачі (5.1) нижче наведено повний алгоритм розв’язання, а також порівняння наближеного і точного розв’язку.

```
% lr5_main.m
% Лабораторна робота №5 з МО
% Розв'язування крайової задачі для нестационарних
% рівнянь та систем різнотипових рівнянь
% методом Рунге (прямих)

clc
clear

global k g D N N1 NN x

% параметри модельної задачі :
k=1; g=1; d=1;
% параметри різницевої схеми :
N=41; N1=N-1; h=1./N1; x=[0:h:1]; x(N)=1; NN=2.*N; NN1=NN-1;
tau=1e-2; M=1000;
D=1./(d.*h);
% формування початкових умов при t=0 :
Y=lr5_u0;
Y0=Y; % запам'ятовуємо для побудови графіку розв'язку в t=0

tk=0;
for j=1:M
% знаходження розв'язку на відрізку [t0,tk] :
t0=tk;
tk=t0+tau;
y0=(Y(2:NN1))';
[t,y]=ode45(@lr5_ff,[t0 tk],y0);

rf=size(y); m=rf(1);
Y(1)=lr5_my1(tk);
for i=2:NN1
Y(i)=y(m,i-1);
end
Y(NN)=lr5_my2(tk);

if j*2==M
```

```

    Y1=Y;    % запам'ятовуємо для побудови графіку розв'язку в t=M*tau/2
end
end

% побудова графіків розв'язку при t=0, M*tau/2, M*tau :
t=M.*tau; t1=0.5.*t; T1=num2str(t1); T=num2str(t);
plot(x,Y0(1:N),'b.-',x,Y1(1:N),'k.--',x,Y(1:N),'r*-');
xlabel('x');
ylabel('U');
legend('U(x,0)',strcat('U(x,',T1,')'),strcat('U(x,',T,')'),'Location','Best');
pause

plot(x,Y0(N+1:NN),'b.-',x,Y1(N+1:NN),'k.--',x,Y(N+1:NN),'r*-');
xlabel('x');
ylabel('V');
legend('V(x,0)',strcat('V(x,',T1,')'),strcat('V(x,',T,')'),'Location','Best');
pause

% порівняння точного розв'язку U(x,t)=sin(pi*x-t), V(x,t)=cos(pi*x-t)
% і наближеного Y(:) в точці t=M*tau :
u=@(x,t)(sin(pi.*x-t));
U=u(x,t);
plot(x,Y(1:N),'b.-',x,U,'r*-');
xlabel('x');
ylabel('u');
legend(strcat('u(x,',T,')'),strcat('U(x,',T,')'),'Location','Best');
norm(Y(1:N)-U,inf)
pause

v=@(x,t)(cos(pi.*x-t));
V=v(x,t);
plot(x,Y(N+1:NN),'b.-',x,V,'r*-');
xlabel('x');
ylabel('v');
legend(strcat('v(x,',T,')'),strcat('V(x,',T,')'),'Location','Best');
norm(Y(N+1:NN)-V,inf)

function u=lr5_u0
% Функція U(x,0), V(x,0)
global N x
for i=1:N
    xx=pi.*x(i);
    u(i)=sin(xx);
    u(i+N)=cos(xx);
end
return

function y=lr5_my1(t)
% Функція U(0,t)
y=sin(-t);
return

function y=lr5_my2(t)
% Функція V(1,t)
y=cos(pi-t);
return

function f=lr5_ff(t,U)
% Опис правої частини системи ЗДР
global k g D N N1 NN x
n=NN-2; ii=N1-1; jj=N+1; f=zeros(n,1); yy1=lr5_my1(t); yyNN=lr5_my2(t);
f(1)=pi.*cos(pi.*x(2)-t)-g.*U(jj)-D.*(U(1)-yy1);

```

```

for i=2:ii
    i1=N1+i; i2=N+i; i3=i-1;
    f(i)=pi.*cos(pi.*x(i+1)-t)-g.*U(i2)-D.*(U(i)-U(i3));
    f(i1)=D.*(U(i2)-U(i1))+k.*U(i3)+pi.*sin(pi.*x(i)-t);
end
f(N1)=pi.*cos(pi.*x(N)-t)-g.*yyNN-D.*(U(N1)-U(ii));
f(N)=D.*(U(jj)-U(N))+k.*yy1+pi.*sin(pi.*x(1)-t);
f(n)=D.*(yyNN-U(n))+k.*U(ii)+pi.*sin(pi.*x(N1)-t);
return

```

На рис. 5.1 – 5.4 наведено графіки отриманих розв’язків для різних моментів часу.

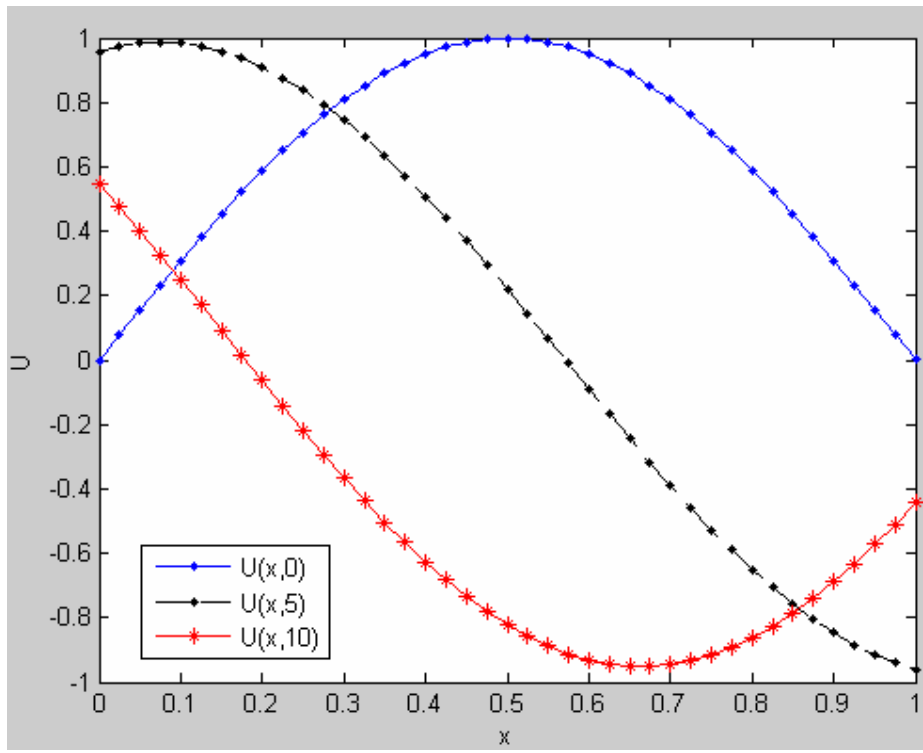


Рис. 5.1. Графік наближеного розв’язку $U(x,t)$ для різних значень t .

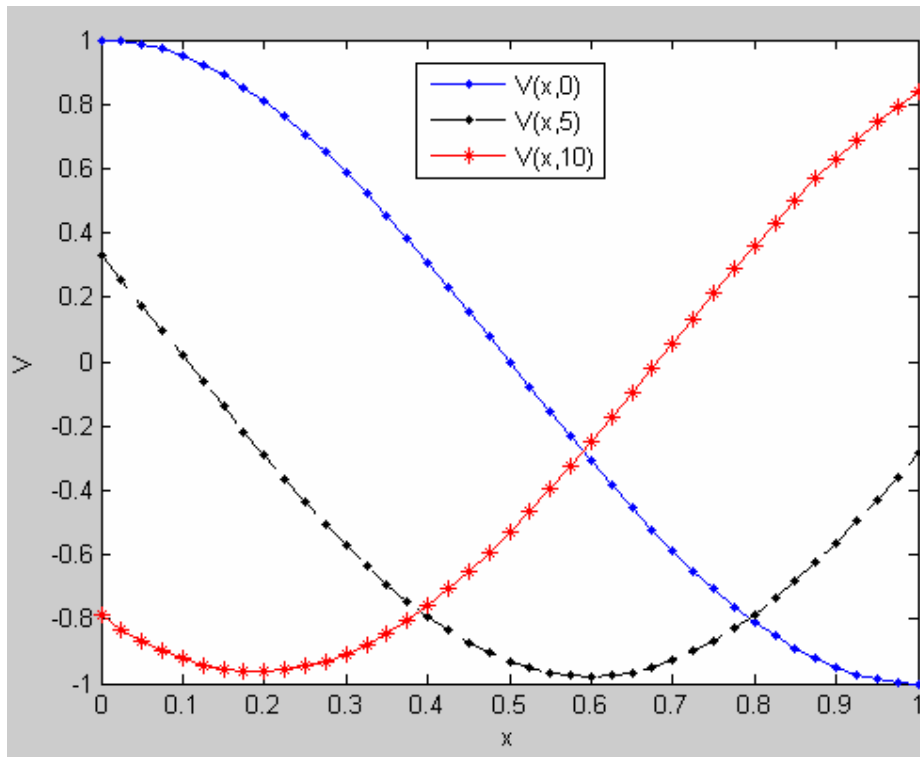


Рис. 5.2. Графік наближеного розв'язку $V(x,t)$ для різних значень t .

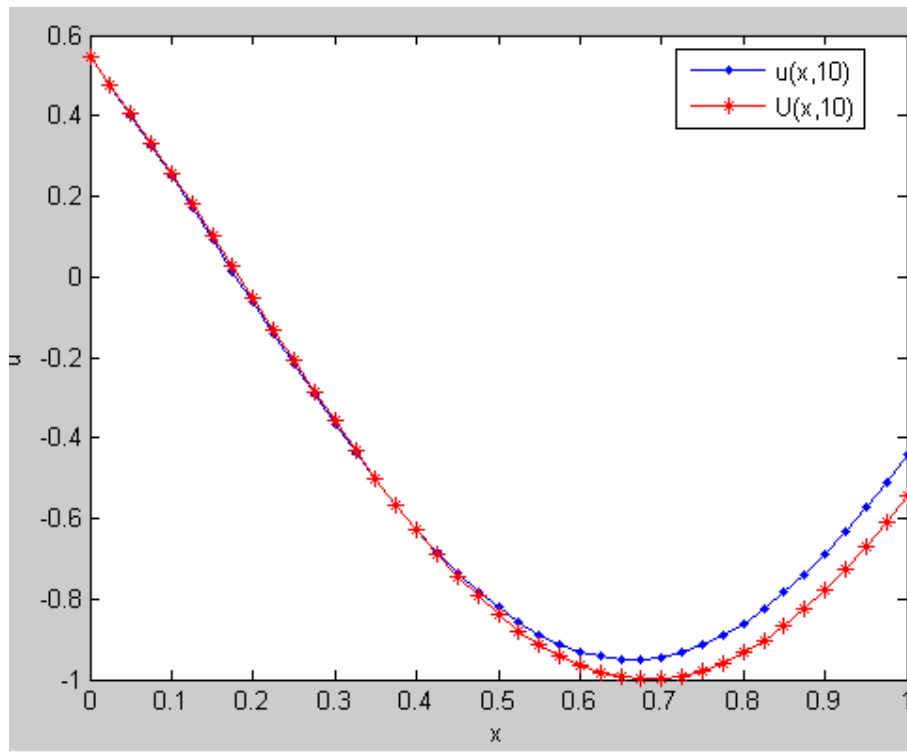


Рис. 5.3. Графік наближеного розв'язку $u(x,t)$ і точного $U(x,t)$ для $t=10$.

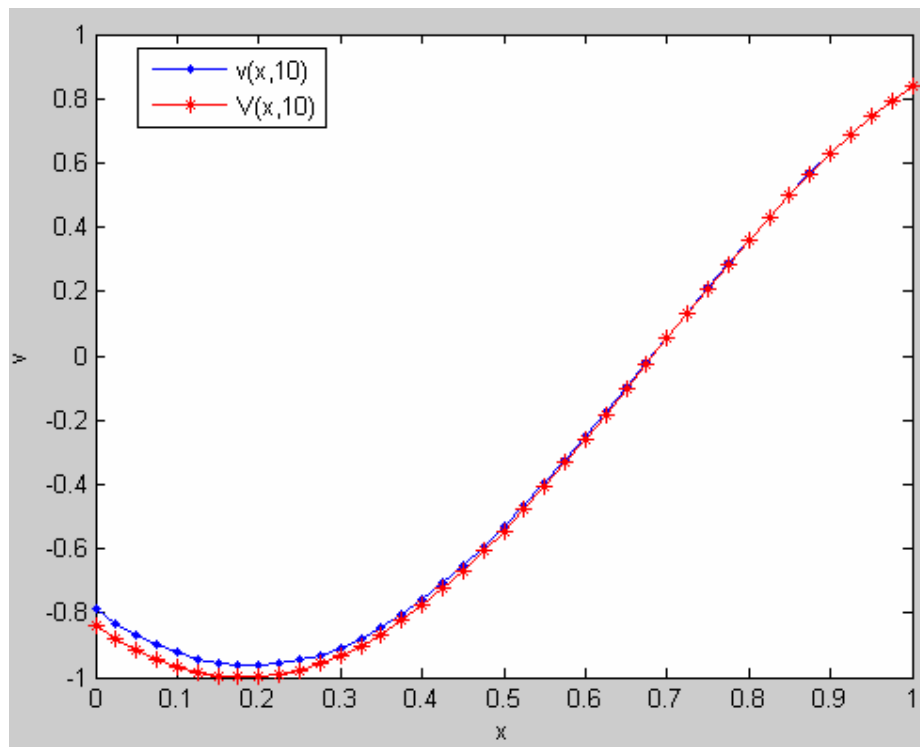


Рис. 5.4. Графік наближеного розв'язку $v(x,t)$ і точного $V(x,t)$ для $t=10$.

ЛАБОРАТОРНА РОБОТА №6
РОЗВ'ЯЗАННЯ КРАЙОВОЇ ЗАДАЧІ ДЛЯ ОДНОВИМІРНОГО
РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ РІЗНИЦЕВИМ МЕТОДОМ

Постановка задачі. Використовуючи *різницевий сітковий метод* розв'язати крайову задачу

$$\begin{cases} \frac{\partial U(x,t)}{\partial t} = \frac{\partial^2 U(x,t)}{\partial x^2} + f(x,t), & 0 < x < l, \quad 0 < t < T; \\ U(x,0) = U_0(x), & 0 \leq x \leq l; \\ c_1 U(0,t) + d_1 \frac{\partial U(0,t)}{\partial x} = \varphi_1(t), \quad c_2 U(l,t) + d_2 \frac{\partial U(l,t)}{\partial x} = \varphi_2(t), & 0 \leq t < T. \end{cases} \quad (6.1)$$

Тут введено позначення :

k	$f(x,t)$	l	$U_0(x)$	c_1	d_1	$\varphi_1(t)$	c_2	d_2	$\varphi_2(t)$
1	$(1-x^g)/(1+t)$	1	$0.5x^2d$	0	1	0	0	1	d
2	$(1-x)/(1+t^{1+g})$	2	$2x$	1	0	0	2	1	10
3	$-2xt(1-x)$	1	$3\sin(7\pi x)$	1	0	0	1	0	0
4	$-U+x+tx^2$	1	0	1	0	0	1	0	t
5	$-U+(4x+6)x^2$	1	3	0	1	0	0	1	0
6	-6	3	$3x^2(3-x)^2$	0	1	0	0	1	0
7	$-U-\cos(2.5\pi x)$	1	x^2-1	0	1	0	1	0	0
8	$(1-x/2)(\cos t-x\sin t)$	2	$\sin(\pi x)$	1	0	$\sin t$	1	0	0
9	$-U$	1	$0.5x^2d$	0	1	0	0	1	d
10	$0.1\pi \cos 0.25\pi x$	2	0	0	1	0	1	0	0
11	$-Ud$	1	$-36.6\cos^2(\pi x)$	0	1	0	0	1	0
12	$-U+d\exp(-t)$	1	$x(1-x)$	1	1	0	1	0	0
13	$-x(2-x)\sin t$	2	$3\sin(3\pi x)$	1	0	0	1	0	0
14	$-Ud$	1	$2x(1-x)$	1	0	0	1	0	0
15	$(x^2-4)\sin t$	2	$4\cos(0.75\pi x)$	0	1	0	1	0	0
16	$-4U$	3	x^2	0	1	0	0	1	6
17	$(1-x^g)/(1+t)$	1	0	-1	1	$5t$	1	0	$5t$
18	$(1-x)/(1+t^{1+g})$	1	$x^3d/3$	0	1	0	0	1	d
19	$0.5x^2$	1	0	0	1	0	0	1	t
20	$-2(U-t)$	2	0	0	1	0	0	1	0
21	$-Ud$	1	$x-1$	-2	1	3	1	0	0
22	$-6U$	1	$1-\cos(2\pi x)$	1	0	0	d	1	0
23	$-U-xt$	1	0	1	1	0	1	1	0
24	$-25U$	1	$2\cos x + \sin x$	-0.5	1	0	1	0	0
25	$-U+\operatorname{sh} t$	1	1	0	1	0	1	0	0

k – номер за списком студента в групі;
 g – номер групи;
 d – день народження студента.

Зауваження. Для більшості задач можна знайти аналітичний розв'язок [6].

Методичні вказівки до розв'язання. Розглянемо наступну модельну задачу.
 При значеннях

$$\begin{aligned} f(x,t) &= (2 - x(1-x))\exp(-t), \\ U0(x) &= x(1-x), \\ l=1, c_1 &= 1, d_1 = 0, \phi_1(t) = 0, \\ c_2 = 0, d_2 &= 1, \phi_2(t) = -\exp(t), \end{aligned} \quad (6.2)$$

задача має наступний розв'язок $U(x,t) = x(1-x)\exp(-t)$.

Введемо рівномірну сітку за змінними x,t :

$$\begin{aligned} \bar{\omega} &= \bar{\omega}_h \times \bar{\omega}_\tau, \\ \bar{\omega}_h &= \{x_i = (i-1)h, i = \overline{1, N}, h = 1/(N-1)\}, \\ \bar{\omega}_\tau &= \{t_j = j\tau, j = \overline{0, M}, \tau > 0\}. \end{aligned}$$

Виконаємо на шаблоні $S\{(x_{i-1}, t_j), (x_i, t_j), (x_{i+1}, t_j); (x_i, t_{j+1})\}$ безпосередню заміну диференціальних операторів на різницеві, застосуємо метод “законтурних точок” для апроксимації з порядком $O(h^2)$ крайової умови в точці $x=1$.

Отримаємо явну різницеву схему з порядком $O(\tau + h^2)$ [1, с. 272] :

$$\begin{aligned} y_i^0 &= U0(x_i), i = \overline{1, N}; \\ y_1^{j+1} &= \phi_1(t_{j+1}); \\ y_i^{j+1} &= y_i^j + \gamma(y_{i-1}^j - 2y_i^j + y_{i+1}^j) + \tau f_{i,j+0.5}, \quad i = 2, 3, \dots, N-1; \quad \gamma = \tau h^{-2}; \\ y_N^{j+1} &= y_N^j + 2\gamma(y_{N-1}^j - y_N^j) + D\phi_2(t_j) + \tau f_{N,j+0.5}, \quad D = 2\gamma h; \\ j &= \overline{0, M-1}. \end{aligned} \quad (6.3)$$

При використанні схеми (6.3) необхідно враховувати умову стійкості $\tau \leq 0.5h^2$.

Використовуючи шаблон $S\{(x_{i-1}, t_j), (x_i, t_j), (x_{i+1}, t_j); (x_{i-1}, t_{j+1}), (x_i, t_{j+1}), (x_{i+1}, t_{j+1})\}$, і числовий параметр $0 < \sigma \leq 1$ можна отримати неявну різницеву схему (порядок апроксимації залежить від σ):

$$\begin{aligned} y_i^0 &= U0(x_i), i = \overline{1, 2, \dots, N}, \\ \begin{cases} A_i y_{i-1}^{j+1} - C_i y_i^{j+1} + B_i y_{i+1}^{j+1} = -F_i, \\ i = \overline{1, 2, \dots, N}; j = \overline{0, 1, \dots, M-1}. \end{cases} \end{aligned} \quad (6.4)$$

Тут введено наступні позначення:

$$\begin{aligned} i=1: & \quad A_1 = 0, \quad C_1 = 1, \quad B_1 = 0, \quad F_1 = \phi(t_{j+1}); \\ i = \overline{2, N-1}: & \quad A_i = \sigma\gamma, \quad C_i = 1 + 2\sigma\gamma, \quad B_i = A_i, \\ & \quad F_i = y_i^j + (1-\sigma)\gamma(y_{i-1}^j - 2y_i^j + y_{i+1}^j) + \tau f_{i,j+0.5}; \\ i=N: & \quad A_N = 2\sigma\gamma, \quad C_N = 1 + 2\sigma\gamma, \quad B_N = 0, \\ & \quad F_N = y_N^j + 2(1-\sigma)\gamma(y_{N-1}^j - y_N^j) + D_1\phi_2(t_j) + \tau f_{N,j+0.5} + D_2\phi_2(t_{j+1}), \\ & \quad D_1 = 2(1-\sigma)\gamma h, \quad D_2 = 2\sigma\gamma h. \end{aligned}$$

Для кожного значення j в (6.3) отримали СЛАР з тридіагональною матрицею, яку ефективно розв'язуємо методом монотонної прогонки `m_prog` [3, с. 43].

Для модельної задачі (6.1), (6.2) нижче наводиться повний алгоритм розв'язання (`lr60main` – явна схема, `lr6smain` – неявна схема), а також порівняння наближеного і точного розв'язку.

```
% lr60main.m
% Лабораторна робота №6 з МО
% Розв'язування крайової задачі для одновимірною
% рівняння параболічного типу
% різницевим методом (явна схема)

clc
clear

% параметри модельної задачі :
l=1; c1=1; d1=0; c2=0; d2=1;
% параметри різницевої схеми :
N=51; N1=N-1; h=1/N1; x=[0:h:1]; x(N)=1;
tau=2e-4; M=25000;
% обчислення додаткових констант (з урахуванням крайових умов) :
gam=tau./(h.*h); D2=2.*h./d2;
% формування початкових умов при t=0 :
for k=1:N
    Y(k)=lr6_u0(x(k));
end
Y0=Y; % запам'ятовуємо для побудови графіку розв'язку в t=0
for j=1:M
% знаходження наближеного розв'язку Y(:) на черговому шарі t=t(j) :
    t=(j-1).*tau;
    t1=t+0.5.*tau;
    yc=Y(1);
    Y(1)=lr6_fil(t+tau);
    for k=2:N1
        yl=yc; yc=Y(k);
        Y(k)=yc+gam.*(yl-2.*yc+Y(k+1))+tau.*lr6_f(x(k),t1);
    end
    yl=yc; yc=Y(N); yr=yl+D2.*lr6_fi2(t);
    Y(N)=yc+gam.*(yl-2.*yc+yr)+tau.*lr6_f(1,t1);
    if j*2==M
        Y1=Y; % запам'ятовуємо для побудови графіку розв'язку в t=M*tau/2
    end
end

% побудова графіків розв'язку при t=0, M*tau/2, M*tau :
t=M.*tau; t1=t./2; T1=num2str(t1); T=num2str(t);
plot(x,Y0,'b.-',x,Y1,'k.--',x,Y,'r*-');
xlabel('x');
ylabel('u');
legend('u(x,0)',strcat('u(x,',T1,')'),strcat('u(x,',T,')'),'Location','Best');
pause

% порівняння точного розв'язку U(x,t)=x(1-x)exp(-t) і наближеного Y(:)
% в точці t=M*tau :
u=@(x,t)(x.*(1-x).*exp(-t));
U=u(x,t);
plot(x,Y,'b.-',x,U,'r*-');
xlabel('x');
ylabel('u');
legend(strcat('u(x,',T,')'),strcat('U(x,',T,')'),'Location','Best');
norm(Y-U,inf)
```

```

% lr6smain.m
% Лабораторна робота №6 з МО
% Розв'язування крайової задачі для одновимірного
% рівняння параболічного типу
% різницевим методом (неявна схема)

clc
clear

% параметри задачі :
l=1; c1=1; d1=0; c2=0; d2=1;
% параметри різницевої схеми :
N=51; N1=N-1; h=1/N1; x=[0:h:1]; x(N)=1;
tau=1e-2; M=500;
sigma=0.5;
% обчислення додаткових констант (з урахуванням крайових умов) :
gam=tau./(h.*h); gs=gam.*sigma; gs2=2.*gs; s1=1-sigma; gs1=gam.*s1;
ck=1+gs2; ckj=1-2.*gs1; D2=2.*h./d2; D2gs=gs.*D2;
% формування початкових умов при t=0 і матриці СЛАР :
for k=1:N
    Y(k)=lr6_u0(x(k));
    A(k)=gs; C(k)=ck; B(k)=gs;
end
A(1)=0; C(1)=1; B(1)=0; % враховуємо крайові умови при x=0
A(N)=gs2; B(N)=0;      % враховуємо крайові умови при x=1
Y0=Y; % запам'ятовуємо для побудови графіку розв'язку в t=0
for j=1:M
% знаходження наближеного розв'язку Y(:) на черговому шарі t=t(j) :
    t=j.*tau;
    t1=t-0.5.*tau;
% формування правої частини СЛАР :
    F(1)=lr6_fi1(t);
    for k=2:N1
        F(k)=gs1.*(Y(k-1)+Y(k+1))+ckj.*Y(k)+tau.*lr6_f(x(k),t1);
    end
    F(N)=gs1.*(2.*Y(N-1)+D2.*lr6_fi2(t-
tau))+ckj.*Y(N)+tau.*lr6_f(1,t1)+D2gs.*lr6_fi2(t);
% розв'язок СЛАР, знаходження Y(:) :
    [Y,alfa,err]=m_prog(A,C,B,F);
    if j*2==M
        Y1=Y; % запам'ятовуємо для побудови графіку розв'язку в t=M*tau/2
    end
end

% побудова графіків розв'язку при t=0, M*tau/2, M*tau :
t=M.*tau; t1=t./2; T1=num2str(t1); T=num2str(t);
plot(x,Y0,'b.-',x,Y1,'k.--',x,Y,'r*-');
xlabel('x');
ylabel('u');
legend('u(x,0)',strcat('u(x,',T1,')'),strcat('u(x,',T,')'),'Location','Best');
pause

% порівняння точного розв'язку U(x,t)=x(1-x)exp(-t) і наближеного Y(:)
% в точці t=M*tau :
u=@(x,t)(x.*(1-x).*exp(-t));
U=u(x,t);
plot(x,Y,'b.-',x,U,'r*-');
xlabel('x');
ylabel('u');
legend(strcat('u(x,',T,')'),strcat('U(x,',T,')'),'Location','Best');
norm(Y-U,inf)

```

```
function f=lr6_f(x,t)
% Функція f(x,t)
f=(2-x.*(1-x)).*exp(-t);
return
```

```
function u=lr6_u0(x)
% Функція u0(x)
u=x.*(1-x);
return
```

```
function fi1=lr6_fi1(t)
% Функція fi1(t)
fi1=0;
return
```

```
function fi2=lr6_fi2(t)
% Функція fi2(t)
fi2=-exp(-t);
return
```

На рис. 6.1 – 6.2 наведено графіки отриманих розв'язків для різних моментів часу.

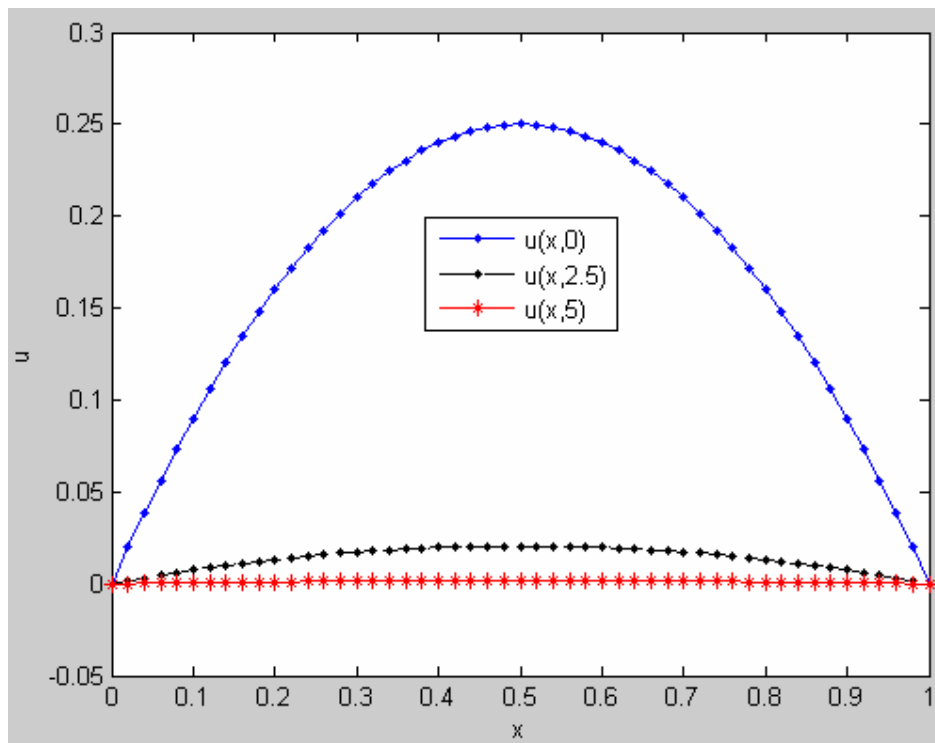


Рис. 6.1. Графік наближеного розв'язку $U(x,t)$ для різних значень t .

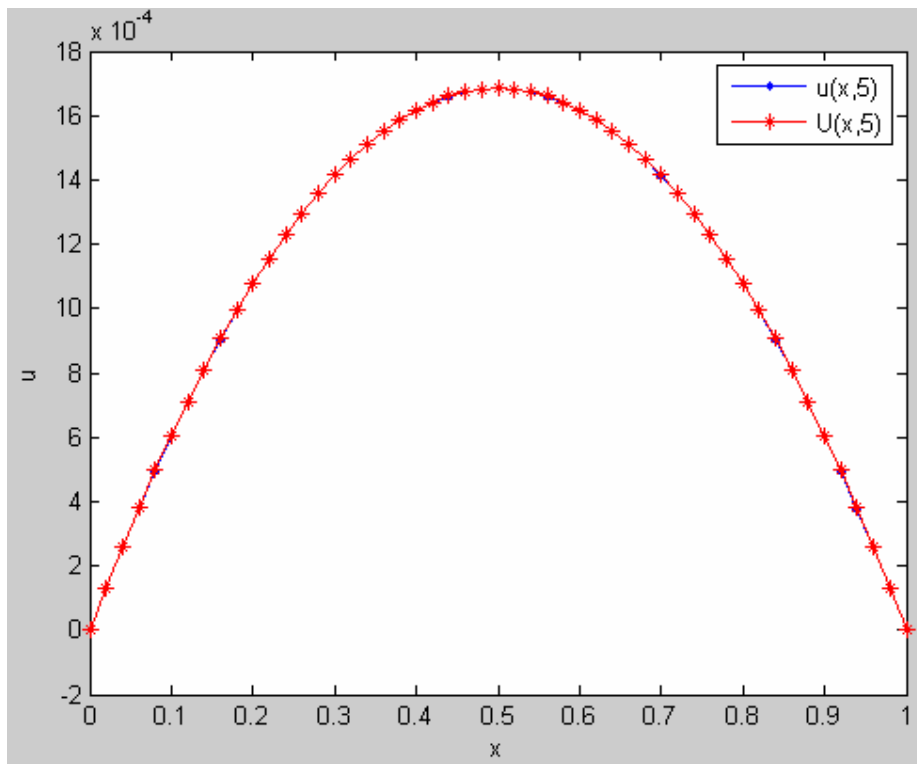


Рис. 6.2. Графік наближеного розв'язку $u(x,t)$ і точного $U(x,t)$ для $t=5$.

ЛАБОРАТОРНА РОБОТА №7 РОЗВ'ЯЗАННЯ КРАЙОВИХ ЗАДАЧ ДЛЯ РІВНЯННЯ ПУАССОНА РІЗНИЦЕВИМ МЕТОДОМ

Постановка задачі. Використовуючи *різницевий сітковий метод* та *ітераційний метод (ІМ) верхньої релаксації* (або *ІМ Зейделя*) розв'язати крайову задачу для рівняння

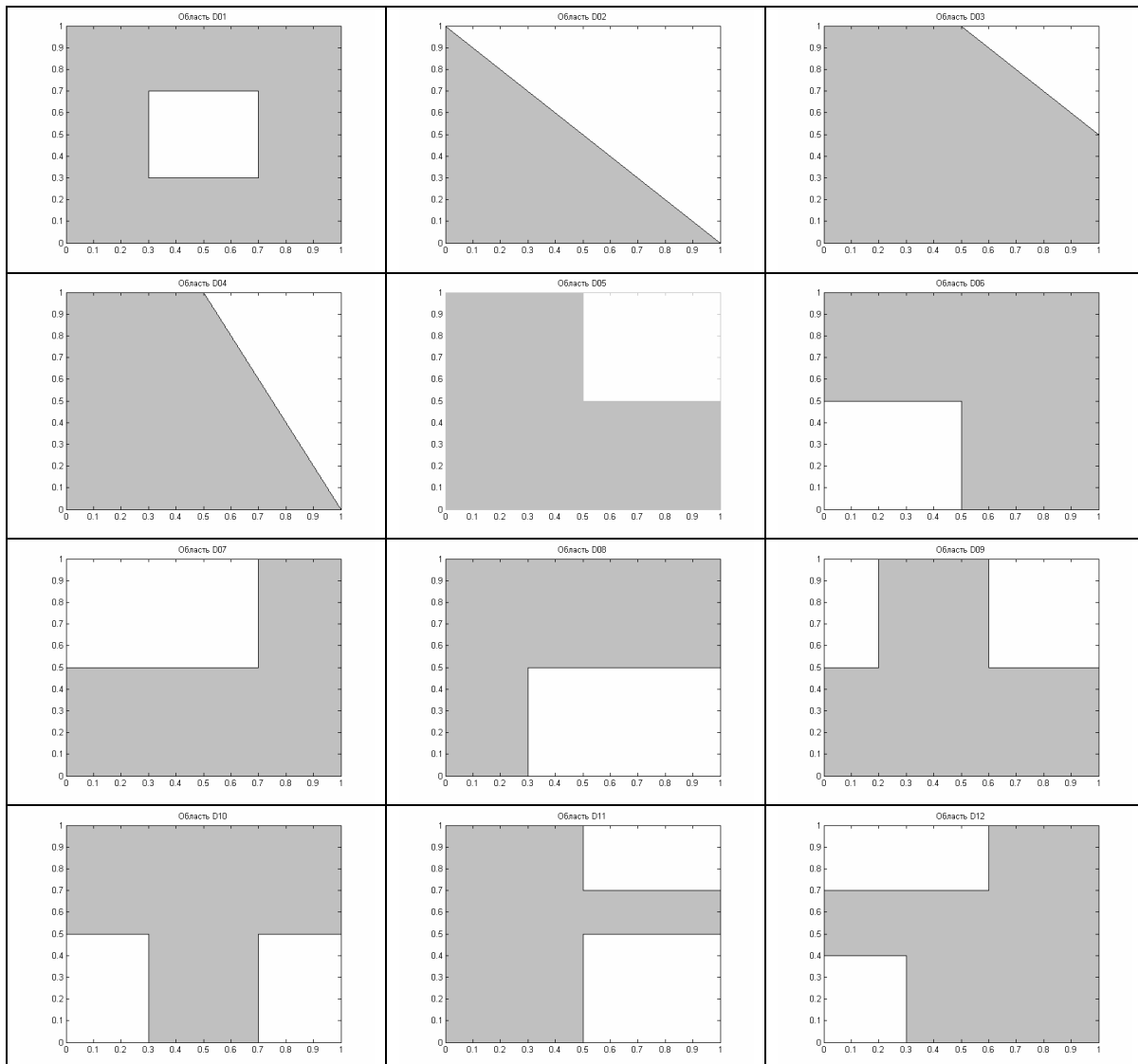
$$\begin{cases} \frac{\partial^2 u(x,y)}{\partial x^2} + \frac{\partial^2 u(x,y)}{\partial y^2} = -x(1-x^{2+g})y(1-y^2), \\ (x,y) \in D \end{cases}$$

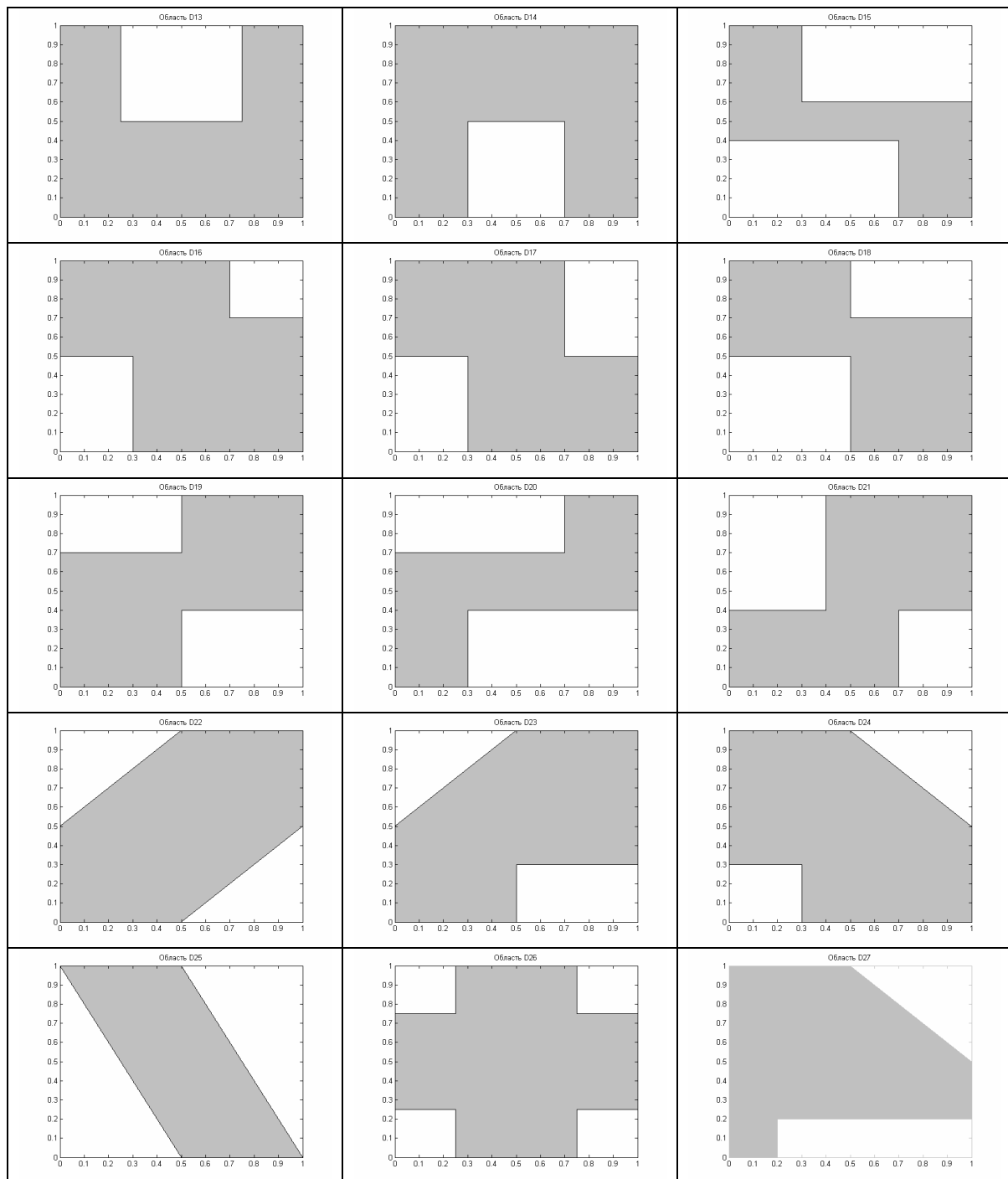
з крайовими умовами, визначеними у варіанті завдання.

Тут введено позначення:

g – номер групи;

k – номер за списком студента в групі, визначає геометрію області D :





d – день народження студента, визначає завдання однорідних крайових умов:

- на похилих лініях області D – 1-го роду;
- на горизонтальних при $\text{mod}(d,2)=0$ – 1-го роду,
при $\text{mod}(d,2)=1$ – 2-го роду;
- на вертикальних при $\text{mod}(d,2)=0$ – 2-го роду,
при $\text{mod}(d,2)=1$ – 1-го роду.

Методичні вказівки до розв'язання. Розглянемо наступну модельну задачу:

$$\frac{\partial^2 U(x_1, x_2)}{\partial x_1^2} + \frac{\partial^2 U(x_1, x_2)}{\partial x_2^2} = -f(x_1, x_2), \quad (x_1, x_2) \in G, \quad (7.1)$$

$$G = \{0 < x_1 < 1 \& 0 < x_2 < 1\} \setminus \{0 < x_1 < 0.3 \& 0 < x_2 < 0.3\} \setminus \{x_2 \geq 2 - 2x_1\},$$

$$f(x_1, x_2) = 2(x_1(1 - x_1) + x_2(1 - x_2)),$$

$$d = 2,$$

яка має точний розв'язок $U(x_1, x_2) = x_1(1 - x_1)x_2(1 - x_2)$.

Введемо зв'язану рівномірну сітку за змінними x_1, x_2 :

$$\bar{\omega} = \bar{\omega}_{h_1} \times \bar{\omega}_{h_2},$$

$$\bar{\omega}_{h_1} = \{x_{1_i} = (i - 1)h_1, i = 1, 2, \dots, N_1, h_1 = 1/(N_1 - 1)\},$$

$$\bar{\omega}_{h_2} = \{x_{2_j} = (j - 1)h_2, j = 1, 2, \dots, N_2, h_2 = 1/(N_2 - 1)\};$$

$$h_2 / h_1 = 2.$$

Виконаємо на шаблоні

$$S\{(x_{1_{i-1}}, x_{2_j}), (x_{1_i}, x_{2_j}), (x_{1_{i+1}}, x_{2_j}), (x_{1_i}, x_{2_{j-1}}), (x_{1_i}, x_{2_j}), (x_{1_i}, x_{2_{j+1}})\}$$

безпосередню заміну диференціальних операторів на різниці, застосуємо метод “законтурних точок” для апроксимації з порядком $O(h_1^2)$ крайових умов на вертикальних лініях і отримаємо різницеву задачу з порядком апроксимації $O(h_1^2 + h_2^2)$ [2, с. 238]. Для розв'язання отриманої СЛАР використаємо ІМ верхньої релаксації [1, с. 384]:

$$2(h_1^{-2} + h_2^{-2})y_{i,j}^{(s)} = \rho h_1^{-2}(y_{i-1,j}^{(s)} + y_{i+1,j}^{(s-1)}) + \rho h_2^{-2}(y_{i,j-1}^{(s)} + y_{i,j+1}^{(s-1)}) - 2(\rho - 1)(h_1^{-2} + h_2^{-2})y_{i,j}^{(s-1)} + \rho f_{i,j}; \quad (7.2)$$

$$1 \leq \rho < 2.$$

Зауваження.

1. При реалізації формул (7.2) потрібно враховувати геометрію області G і крайові умови (див. програмну реалізацію для модельної задачі).
2. При значенні параметра $\rho = 1$ метод верхньої релаксації перетворюється в “точковий” метод Зейделя [1, с. 381]) з кількістю ітерацій $n \approx O((h_1 + h_2)^{-2})$.
3. Для квадратної області G , крайових умов Діріхле і однакових кроках ($h_1 = h_2 = h$) існує оптимальне значення параметру ρ [1, с. 388]), при цьому кількість ітерацій $n \approx O(h^{-1})$.

4. Чисельні експерименти показують, що $\rho_{opt} \in \left(\frac{2}{1 + 2 \sin(0.5\pi \min(h_1, h_2))}, 2 \right)$. Для нашої модельної задачі $\rho_{opt} = 1.965$, а кількість ітерацій $n = 512$ (для довідки, у методі Зейделя $n = 16519$ при $N_1 \cdot N_2 = 101 \cdot 201$).

Для модельної задачі (7.1) нижче наведено повний алгоритм розв'язання, а також порівняння наближеного і точного розв'язку.

§ `lr7_main.m`

§ Лабораторна робота №7 з МО

§ Розв'язування змішаної крайової задачі для рівняння Пуассона

§ в області, складеної з класичних геометричних елементів

```

% (прямокутників, трапецій, трикутників)

clc
clear

% параметри модельної задачі :
a=1; b=1;
Hx=0.005; Hy=0.01;
Nx=201; Ny=101;
Nx11=61; Nx21=101; Ny11=31;
ro=1.965; e=1e-8; kmax=20000;
Nx1=Nx-1; Ny1=Ny-1;
X=[0:Hx:a];
Y=[0:Hy:b];
HxH=1./(Hx.*Hx); HyH=1./(Hy.*Hy);
C=ro.*0.5./(HxH+HyH); A=HxH.*C; B=HyH.*C; D=ro-1;
% формування крайових умов та початкового наближення :
U=zeros(Ny,Nx);
% (0 <= X < 0.3) & (0.3= Y)
for i=1:Nx11
    U(Ny11,i)=lr7_mx11(X(i));
end
% (0.3<= X <= 1) & (0 = Y)
for i=Nx11:Nx
    U(1,i)=lr7_mx12(X(i));
end
% (0 <= X < 0.5) & (1 = Y)
for i=1:Nx21
    U(Ny,i)=lr7_mx21(X(i));
end
% (0.5 <= X <= a) & (2-2*X = Y)
for j=2:Ny1
    i=Nx-j+1;
    U(j,i)=lr7_mxy(X(i));
end

k=0;
Fla=0;
while ~Fla & k<kmax
    Fla=1;
    k=k+1;

    for j=2:Ny11-1
        i=Nx11;
        u=U(j,i);
        UU=A.*2.0.*(U(j,i+1)-Hx.*lr7_my11(Y(j)))+...
            B.*(U(j-1,i)+U(j+1,i))-D.*u+C.*lr7_f(X(i),Y(j));
        if Fla
            Fla=abs(UU-u)<e;
        end
        U(j,i)=UU;
        for i=Nx11+1:Nx-j
            u=U(j,i);
            UU=A.*(U(j,i-1)+U(j,i+1))+B.*(U(j-1,i)+U(j+1,i))-...
                D.*u+C.*lr7_f(X(i),Y(j));
            if Fla
                Fla=abs(UU-u)<e;
            end
            U(j,i)=UU;
        end
    end
end

j=Ny11;

```

```

for i=Nx11+1:Nx-j
    u=U(j,i);
    UU=A.*(U(j,i-1)+U(j,i+1))+B.*(U(j-1,i)+U(j+1,i))-...
        D.*u+C.*lr7_f(X(i),Y(j));
    if Fla
        Fla=abs(UU-u)<e;
    end
    U(j,i)=UU;
end

for j=Ny11+1:Ny1;
    i=1;
    u=U(j,i);
    UU=A.*2.0.*(U(j,i+1)-Hx.*lr7_my12(Y(j)))+B*(U(j-1,i)+U(j+1,i))-...
        D.*u+C.*lr7_f(X(i),Y(j));
    if Fla
        Fla=abs(UU-u)<e;
    end
    U(j,i)=UU;
    for i=2:Nx-j
        u=U(j,i);
        UU=A.*(U(j,i-1)+U(j,i+1))+B*(U(j-1,i)+U(j+1,i))-...
            D.*u+C.*lr7_f(X(i),Y(j));
        if Fla
            Fla=abs(UU-u)<e;
        end
        U(j,i)=UU;
    end
end

end

% побудова графіку розв'язку :
k
surfl(U);
shading interp;
colormap(pink);
pause

u=@(x,y)(x.*(1-x).*y.*(1-y));
UU=zeros(Ny,Nx);
for j=1:Ny
    for i=1:Nx
        if (j>=Ny11 | i>=Nx11) & i<=Nx-j+1
            UU(j,i)=u(X(i),Y(j));
        end
    end
end
surfl(UU);
shading interp;
colormap(pink);
DIF=U-UU;
norm(DIF,inf)
pause

surfl(DIF);
shading interp;
colormap(pink);

function f=lr7_f(x,y)
% Функція f(x,y)
f=2.*(x.*(1-x)+y.*(1-y));
return

```

end

```
function u=lr7_mx11(x)
% Крайові умови на  $(0 \leq x \leq 0.3) \& (y=0.3)$ 
u=0.21.*x.*(1-x);
return
end
```

```
function u=lr7_mx12(x)
% Крайові умови на  $(0.3 \leq x \leq 1) \& (y=0)$ 
u=0;
return
end
```

```
function u=lr7_mx21(x)
% Крайові умови на  $(0 \leq x \leq 0.5) \& (y=1)$ 
u=0;
return
end
```

```
function u=lr7_mxy(x)
% Крайові умови на  $(0.5 \leq x \leq 1) \& (y=2-2x)$ 
xx=2.*x;
u=xx.*(1-x).^2.*(xx-1);
return
end
```

```
function u=lr7_my11(y)
% Крайові умови на  $(0 < y < 0.3) \& (x=0.3)$ 
u=0.4.*y.*(1-y);
return
end
```

```
function u=lr7_my12(y)
% Крайові умови на  $(0.3 < y < 1) \& (x=0)$ 
u=y.*(1-y);
return
end
```

На рис. 7.1–7.2 наведено графіки наближеного розв’язку задачі і різниці між наближеним і точним розв’язками.

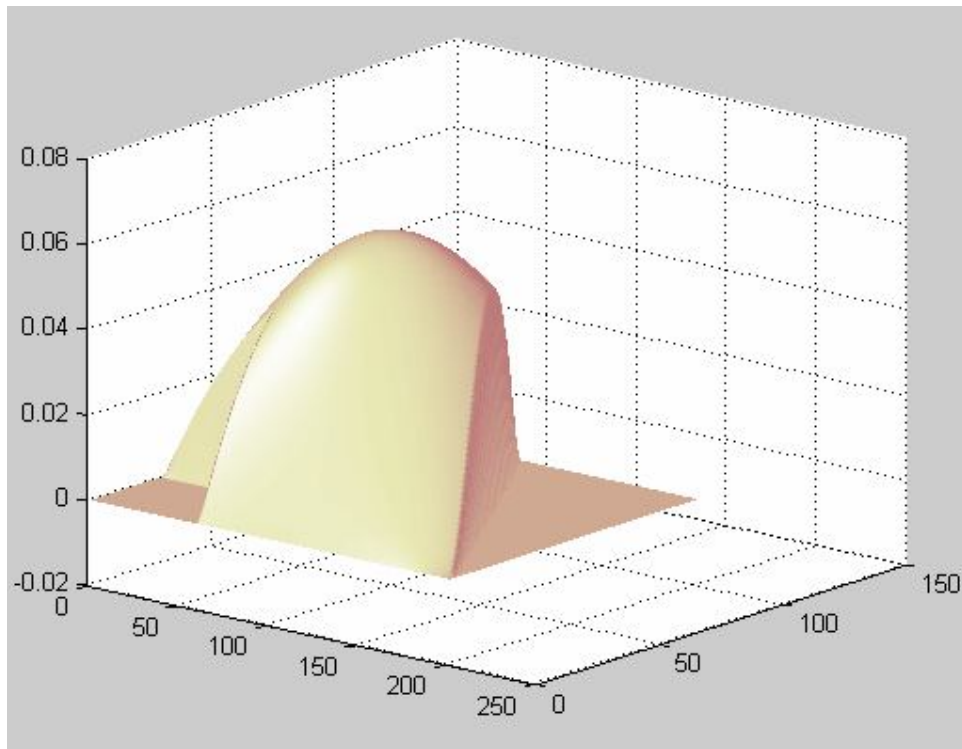


Рис. 7.1. Графік наближеного розв'язку $U(x_1, x_2)$.

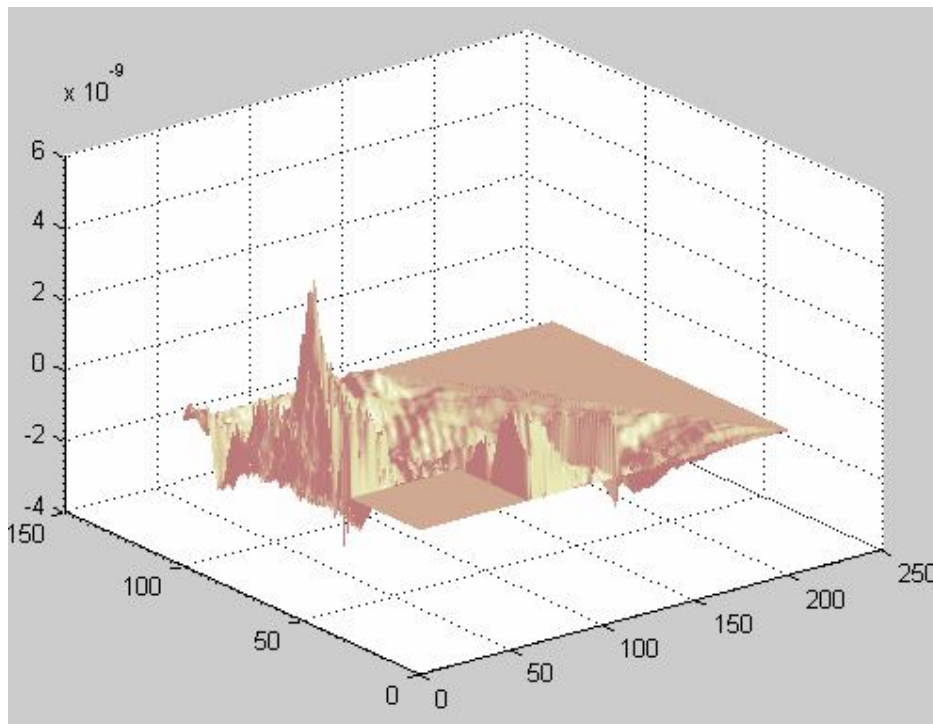


Рис. 7.2. Графік різниці наближеного розв'язку $u(x_1, x_2)$ і точного $U(x_1, x_2)$.

ЛІТЕРАТУРА

1. Самарский А.А. Численные методы / А.А. Самарский, А.В. Гулин – М.: Наука, 1989. – 439 с.
2. Попов В.В. Методи обчислень / В.В. Попов – К.: Видавничо-поліграфічний центр “Київський університет”, 2012. – 303 с.
3. Довгий Б.П. Використання математичного пакета MATLAB для розв’язування прикладних задач / Б.П. Довгий, Є.С. Вакал, Ю.Є. Вакал, А.В. Попов. – К.: Фітосоціоцентр, 2012. – 78 с.
4. Самарский А.А. Методы решения сеточных уравнений / А.А. Самарский, Е.С. Николаев. – М.: Наука, 1978. – 92 с.
5. Ляшко И.И. Методы вычислений / И.И. Ляшко, В.Л. Макаров, А.А. Скоробогатько. – К.: Вища школа, 1977. – 408 с.
6. Маринець В.В. Збірник задач з математичної фізики / В.В. Маринець, М.О. Перестюк, В.Л. Рего. – К.: Либідь, 2009. – 245 с.

ЗМІСТ

ПЕРЕДМОВА	3
Лабораторна робота №1 ІТЕРАЦІЙНІ МЕТОДИ РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ	4
Лабораторна робота №2 МЕТОДИ РОЗВ'ЯЗАННЯ СИСТЕМ НЕЛІНІЙНИХ РІВНЯНЬ	8
Лабораторна робота №3 НАБЛИЖЕННЯ ЛІНІЙНОГО ДИФЕРЕНЦІАЛЬНОГО ОПЕРАТОРА МЕТОДОМ НЕВИЗНАЧЕНИХ КОЕФІЦІЄНТІВ	12
Лабораторна робота №4 ПОБУДОВА ІНТЕРПОЛЯЦІЙНИХ КВАДРАТУРНИХ ФОРМУЛ НАЙВИЩОГО ПОРЯДКА ТОЧНОСТІ.....	14
Лабораторна робота №5 МЕТОД ПРЯМИХ РОЗВ'ЯЗАННЯ КРАЙОВИХ ЗАДАЧ	18
Лабораторна робота №6 РОЗВ'ЯЗАННЯ КРАЙОВОЇ ЗАДАЧІ ДЛЯ ОДНОВИМІРНОГО РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ РІЗНИЦЕВИМ МЕТОДОМ	27
Лабораторна робота №7 РОЗВ'ЯЗАННЯ КРАЙОВИХ ЗАДАЧ ДЛЯ РІВНЯННЯ ПУАССОНА РІЗНИЦЕВИМ МЕТОДОМ.....	33
ЛІТЕРАТУРА	40