

Санкт-Петербургский государственный политехнический университет

Институт прикладной математики и механики

Кафедра «Прикладная математика»

Ануфриев И. Е., Осипов П. А.

Математические методы моделирования физических
процессов.

Метод конечных разностей.

С решениями типовых задач

Учебное пособие

Санкт-Петербург

2014

УДК 518:683.03 (075.8)

И.Е. Ануфриев, П.А. Осипов. **Математические методы моделирования физических процессов. Метод конечных разностей. С решениями типовых задач.** Учебное пособие.

Исправленная и дополненная электронная версия существующего пособия «*И.Е. Ануфриев, П.А. Осипов.* Математические методы моделирования физических процессов. Часть 1. Метод конечных разностей». Учебное пособие является руководством по выполнению лабораторных работ по курсу лекций «Математические методы моделирования физических процессов». Соответствует государственному образовательному стандарту дисциплины ОПД.Ф.07 «Математические методы моделирования физических процессов» направления 553100 «Техническая физика» бакалаврской и магистерской подготовки студентов факультета медицинской физики и биоинженерии.

Рассматривается применение метода конечных разностей к решению краевых задач математической физики. Изучается устойчивость и точность численных схем. Приводятся примеры решений типовых задач и коды программ на m – языке МАТЛАБ.

Пособие предназначено для студентов четвертого курса факультета медицинской физики и биоинженерии. Может быть использовано студентами других факультетов как введение в сеточные методы решения краевых задач. Пособие может заинтересовать преподавателей в качестве задачника.

Стр. 130, ил. 57, библиогр. 3 назв.

Печатается по решению редакционно-издательского совета Санкт-Петербургского государственного политехнического университета.

© Санкт-Петербургский государственный политехнический университет, 2014

Оглавление

ВВЕДЕНИЕ.....	3
§1. Устойчивость разностных схем.	4
Листинги к теме «Устойчивость разностных схем».	11
§2. Краевая задача для линейного дифференциального уравнения 18	
Листинги к теме «Краевая задача для линейного дифференциального уравнения»	23
Листинги к задаче 11.....	26
§3. Решение краевой задачи для нелинейного дифференциального уравнения.....	28
Листинги к теме 3. Решение краевой задачи для нелинейного дифференциального уравнения	32
§4. Уравнение теплопроводности. Явная схема.....	43
Листинги к теме 4. Уравнение теплопроводности. Явная схема.	51
§5. Уравнение теплопроводности. Неявная схема.....	62
Листинги к теме 5. Уравнение теплопроводности. Неявная схема.	67
§6. Колебания струны	72
Листинги к теме 6. Колебания струны.	88
§7. Задача Дирихле для уравнения Пуассона.....	113
Листинги к теме 7. Задача Дирихле для уравнения Пуассона.....	120
Литература	129

ВВЕДЕНИЕ

Многие физические явления подчиняются законам, описываемым дифференциальными уравнениями, не всегда разрешимыми аналитически. Аналитические методы решения поставленных задач рассматриваются в первой части курса лекций. Во второй части курса делается введение в метод конечных разностей (МКР).

В пособии рассмотрено использование МКР для решения краевых задач. Студентам предлагается запрограммировать и исследовать решение задач в MATLAB. Язык программирования высокого уровня MATLAB дает возможность писать короткий программный код. Графические функции позволяют просто визуализировать результаты расчетов для их последующего анализа. Приведены листинги программ решения типовых задач. Эти программы отлажены, проверены в учебном процессе, снабжены многочисленными комментариями. Использование «говорящих имен» и явных циклов должно упростить программирование при самостоятельном выполнении предлагаемых задач. Настройки скриптов для решения остальных предложенных задач даны в виде комментариев.

Отметим, что существует большое количество готовых программных продуктов для решения задач математической физики. Однако, осознанное использование готовых программных продуктов невозможно без ясного понимания постановки задачи и численного метода ее решения, включая: выбор сетки; получение и решение соответствующей системы алгебраических уравнений; оценку погрешности решения; оценку скорости его сходимости при измельчении сетки и т. д.

Пособие предназначено для студентов четвертого курса отделения медицинской физики и биоинженерии ИФНИТ, но может быть рекомендовано студентам других институтов, для которых курс по МКР носит обзорный характер.

Авторы благодарят старшего преподавателя каф. «Прикладная математика» В.М. Лыдкина, просмотревшего рукопись и сделавшего

ценные замечания, и доцента каф. «Прикладная математика» В.Л. Дорота за любезно предоставленные варианты задач к § 2.

§1. Устойчивость разностных схем.

Исследуем вопрос о приближении производных их конечно-разностными аналогами. Для приближения производных y', y'' используем следующие конечно-разностные аппроксимации, теоретическая ошибка которых составляет $O(h^2)$:

$$y'(x) \approx y'_h(x) \equiv \frac{y(x+h) - y(x-h)}{2h}, \quad y''(x) \approx y''_h(x) \equiv \frac{y(x+h) - 2y(x) + y(x-h)}{h^2}.$$

Следует иметь в виду, что при вычислениях на компьютере из-за ограниченности разрядной сетки, при уменьшении шага ошибки $|y'(x) - y'_h(x)|$ и $|y''(x) - y''_h(x)|$ не будут убывать в соответствии с теорией. Для проверки рекомендуется проделать приближение первых и вторых производных функции x , x^2 и $\sin x$ и объяснить характер изменения ошибки (см. листинги в приложении).

Пример 1. Приближение первой производной x , x^2 и $\sin x$ в точке $x_0 = 0.1$. Графики теоретической и практической зависимостей ошибки от шага приведены на рис. 1.1 – 1.3.

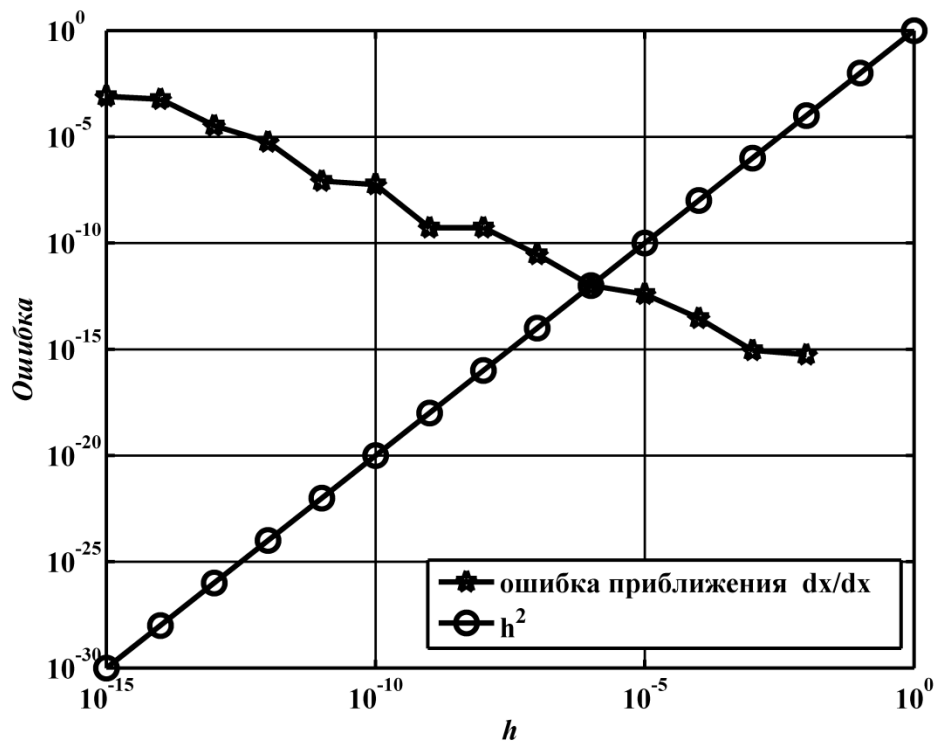


Рисунок 1.1. Ошибка приближения x' . (Deriv_1_x.m)

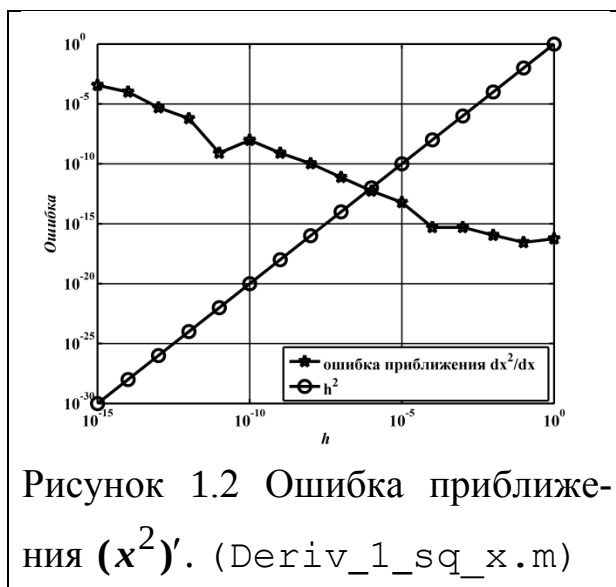


Рисунок 1.2 Ошибка приближения $(x^2)'$. (Deriv_1_sq_x.m)

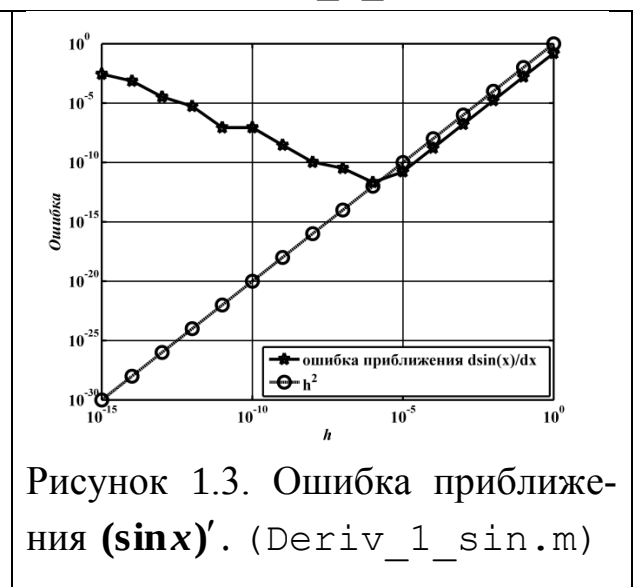


Рисунок 1.3. Ошибка приближения $(\sin x)'$. (Deriv_1_sin.m)

Пример 2. Приближение d^2 / dx^2 функций x , x^2 и $\sin x$ в $(\cdot)_{x_0} = 0.1$.

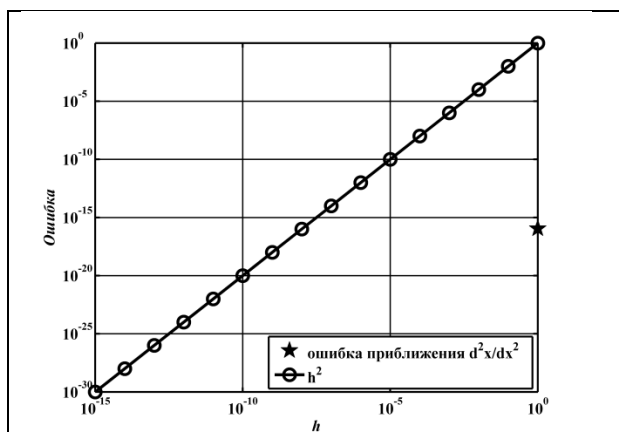


Рисунок 1.4. Ошибка приближения x'' . (Deriv_2_x.m)

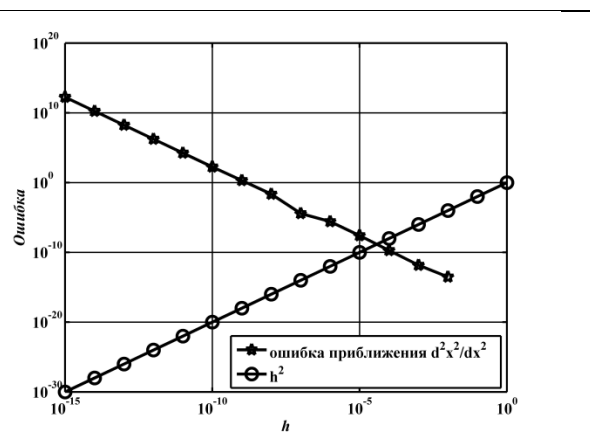


Рисунок 1.5. Ошибка приближения $(x^2)''$. (Deriv_2_sq_x.m)

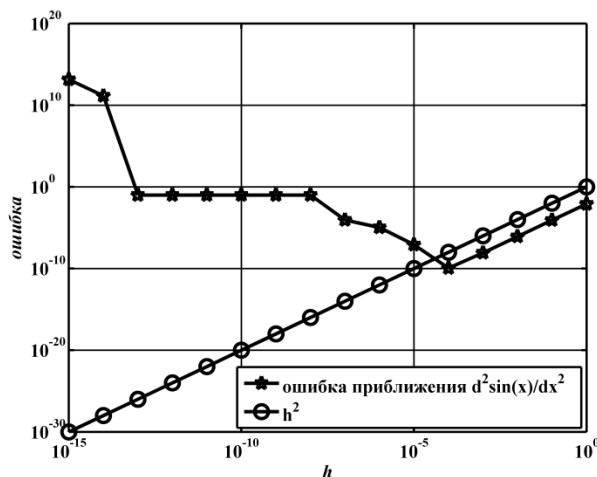


Рисунок 1.6. Ошибка приближения $(\sin x)''$. (Deriv_2_sin.m)

Рассмотрим вопрос устойчивости разностной схемы. Пусть при неограниченном сгущении сетки приближенное решение стремится к решению исходной задачи (схема сходится). Здесь неявно предполагается [1], что разностная задача решается точно и вычисления проводятся с любым количеством знаков. Однако на практике, на каждом шаге вычислений возникают малые ошибки округления.

Ошибки вычислений можно рассматривать как возмущения начальных данных или правой части уравнения [1]. Если малые ошибки округления при сгущении сетки приводят к большим погрешностям, такую схему называют неустойчивой. Исследуем вопросы устойчивости на следующем примере.

Решим задачу Коши

$$\begin{cases} u'(x) = -\alpha \cdot u(x); & x > 0; \alpha > 0; \\ u(0) = \beta \end{cases} \quad (1.1)$$

Точным решением задачи (1.1) является

$$u(x) = \beta e^{-\alpha \cdot x}. \quad (1.2)$$

Применим для решения (1.1) схему с правой конечной разностью. Введем равномерную сетку с шагом h : $x_i = (i-1)h$ и заменим (1.1) разностным уравнением

$$\frac{y_{i+1} - y_i}{h} = -\alpha \cdot y_i; \quad i = 1, 2, \dots; \quad y_1 = \beta. \quad (1.3)$$

Выразим отсюда y_{i+1} через y_i и получим расчетную формулу

$$y_{i+1} = (1 - \alpha h) y_i, \quad i = 1, 2, \dots; \quad y_1 = \beta. \quad (1.4)$$

Исследуем зависимость y_{i+1} от начальных данных β . Проведя некоторые преобразования (см. [1]), получим выражение для y_{i+1} : $y_{i+1} = \beta e^{-\alpha \cdot x_{i+1}} + O(h)$. Из него следует устойчивость по начальным условиям (малые изменения в β вызывают малые изменения в решении y_{i+1}).

Численно решим (1.1) на отрезке $[0, 1]$ при помощи (1.4) с $h=0.05$. Получим приближенное решение (см. рис.1.7. Листинг `Stability.m`).

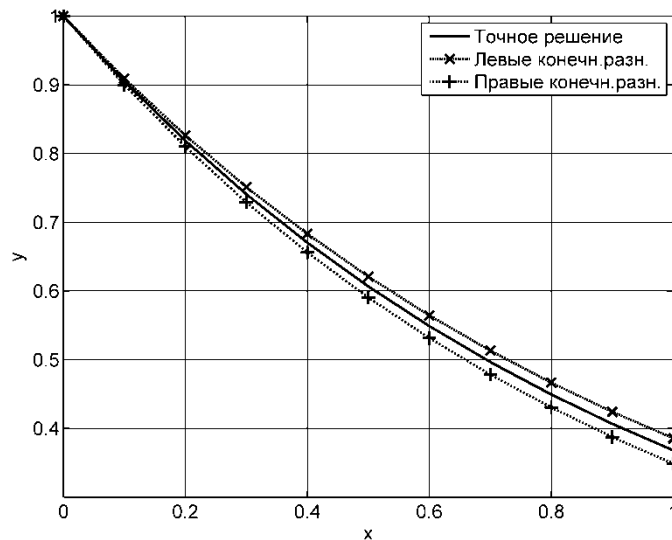


Рисунок 1.7. Точное и приближенное решение задачи (1.1)

Исследуем теперь зависимость ошибки от h , уменьшая h до 10^{-9} (см. рис. 1.8). Ошибка убывает с уменьшением шага, что подтверждает устойчивость метода, а возрастание погрешности при $h < 10^{-8}$ связано с ошибками округления при аппроксимации производной. (ErrorByH.m).

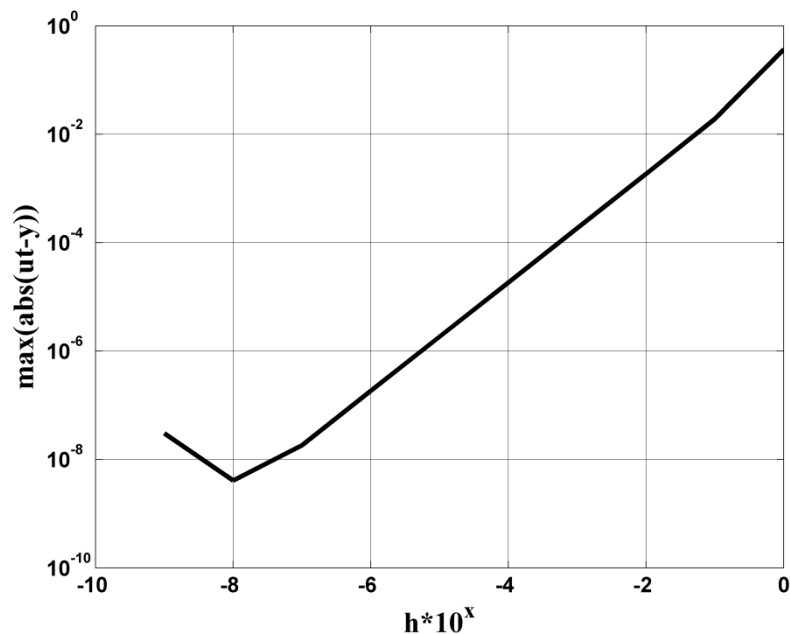


Рисунок 1.8. Зависимость ошибки от шага h

Для численного решения (1.1) можно применить левые конечные разности, которые так же приводят к устойчивой схеме

$$\frac{y_i - y_{i-1}}{h} = -\alpha \cdot y_i; \quad i=1,2,\dots; \quad y_1 = \beta. \quad (1.5)$$

Соответствующие вычисления и исследование устойчивости рекомендуется проделать самостоятельно.

Рассмотрим теперь условно устойчивую схему для решения (1.1). Аппроксимируем дифференциальное уравнение комбинацией предыдущих схем (1.5) и (1.3), соответственно, с весами σ и $(1-\sigma)$, где $\sigma > 0, \sigma \neq 1$.

$$\sigma \frac{y_i - y_{i-1}}{h} + (1-\sigma) \frac{y_{i+1} - y_i}{h} = -\alpha \cdot y_i. \quad (1.6)$$

Схема трехточечная, начальные условия должны быть заданы в 2-х первых точках. Положим $y_1 = \beta$, а y_2 вычислим, используя точное решение (1.2): $y_2 = \beta e^{-\alpha \cdot x_2}$. Получим расчетную формулу

$$y_{i+1} = \frac{1}{1-\sigma} [\sigma y_{i-1} - (2\sigma - 1 + \alpha h) y_i], \quad i = 2, 3, \dots \quad (1.7)$$

Исследуем ее устойчивость. Можно показать (см. [1]), что общее решение имеет вид

$$y_i = A \left(\frac{\sigma}{\sigma-1} \right)^{\frac{\tilde{x}}{h}} e^{\tilde{x}(\alpha+O(h))} + B e^{-\tilde{x}(\alpha+O(h))}, \quad (1.8)$$

где $\tilde{x} = ih$, а A и B однозначно определяются через начальные значения y_1 и y_2 .

Очевидно, что первое слагаемое определяет устойчивость схемы. Если оно неограниченно возрастает при уменьшении шага, то схема (1.6) неустойчива. Построим график $y(\sigma) = \sigma/(\sigma-1)$ (рис. 1.9). Видим, что положительная ветвь гиперболы имеет асимптоты $\sigma=1$ и $y=1$. При $\sigma > 1$ верно $\sigma/(\sigma-1) > 1$, и первое слагаемое в (1.8) при $h \rightarrow 0$ неограниченно растет при любом $A \neq 0$.

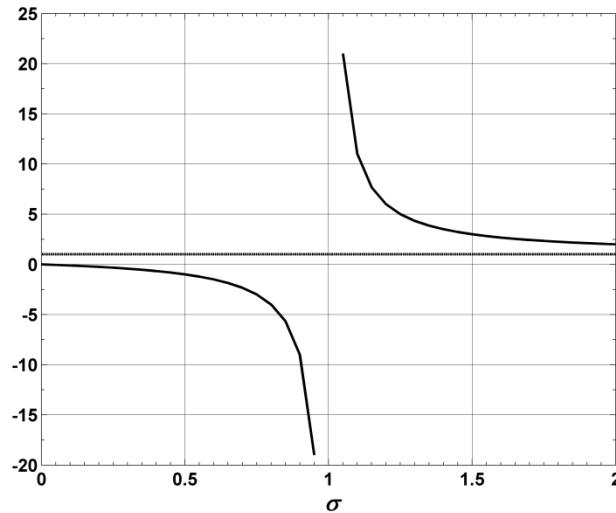
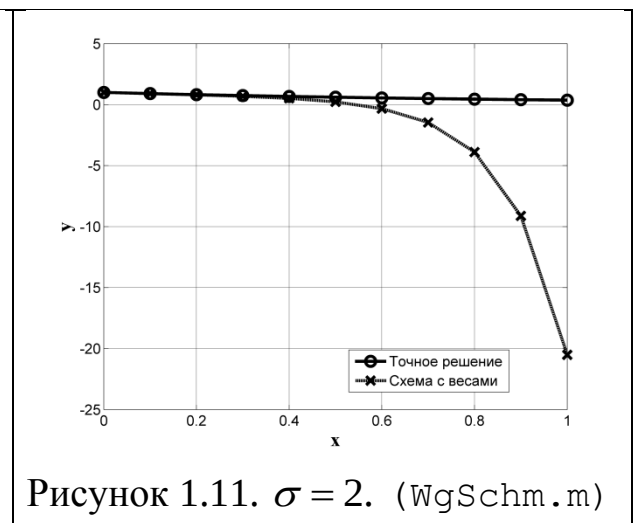
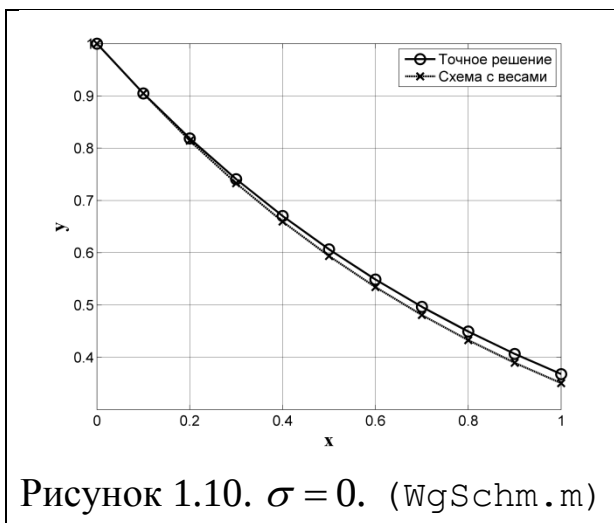


Рисунок 1.9. График $\sigma/(\sigma-1)$. (TestSigma.m)

Если выбрать начальные условия y_1, y_2 так, что $A=0$, то при сколь угодно малом возмущении y_1, y_2 , например, из-за ошибок округления, станет $A \neq 0$; и решение будет неограниченно возрастать при $h \rightarrow 0$.

Покажем неустойчивость схемы с весами при помощи численного эксперимента. Поведение решения зависит от значения σ . Возьмем значения σ симметрично асимптоте $\sigma = 1$. Сначала $\sigma = 0$, потом $\sigma = 2$. Получим решения по устойчивой схеме (рис. 1.10) и по неустойчивой схеме (рис. 1.11) с шагом $h = 0.1$. При вычислении по неустойчивой схеме заметно быстрое накопление ошибки.



Листинги к теме «Устойчивость разностных схем».

```
%Deriv_1_sin.m
% 1-я производная синуса
clear all;close all;
f = @(x)sin(x);
df = @(x)cos(x);
x0 = 0.1;
df_FDM = @(h) (f(x0+h)-f(x0-h))/(2*h);
H = logspace(-15,0,16);
err = abs(df_FDM(H) - df(x0));
figure('color','w');
loglog(H, err, H, H.^2, '--');
grid on
legend('ошибка приближения первой производной
sin(x)', 'h^2')
xlabel('h')
ylabel('ошибка')
```

```
%Deriv_1_square_x.m
% 1-я производная x^2
clear all;close all;
f = @(x)x.^2;
df = @(x)2*x;
x0 = 0.1;
df_FDM = @(h) (f(x0+h)-f(x0-h))/(2*h);
H = logspace(-15,0,16);
err = abs(df_FDM(H) - df(x0));
figure('color','w');
%get(h,'PropertyName');
loglog(H, err, H, H.^2, '--');
%set(h,'PropertyName');
%ha=gcf;
%propName= set(ha,'PropertyName');
grid on
```

```

legend('ошибка    приближения    первой    производной
x^2', 'h^2')
xlabel('h')
ylabel('ошибка')

```

```

%Deriv_1_x.m
f = @(x)x;
df = @(x)1;
x0 = 0.1;
df_FDM = @(h) (f(x0+h)-f(x0-h)) ./ (2*h);
H = logspace(-15,0,16);
err = abs(df_FDM(H) - df(x0));
figure
loglog(H, err, H, H.^2, '--')
grid on
legend('ошибка    приближения    первой    производной',
'h^2')
xlabel('h')
ylabel('ошибка')

```

```

%Deriv_2_sin.m 2-я производная синуса
clear all;close all;
f = @(x)sin(x);
d2f = @(x)-sin(x);
x0 = 0.1;
d2f_FDM = @(h) (f(x0+h)-2*f(x0)+f(x0-h)) ./ (h.^2);
H = logspace(-15,0,16);
h=0.1;
err = abs(d2f_FDM(H) - d2f(x0));
figure('color','w');
%get(h,'PropertyName');
loglog(H, err, H, H.^2, '--');
%set(h,'PropertyName');
%ha=gcf;

```

```

%propName= set(ha,'PropertyName');
grid on
legend('ошибка приближения второй производной
sin(x)', 'h^2')
xlabel('h')
ylabel('ошибка')

%Deriv_2_square_x.m 2-я производная x в квадрате
clear all;close all;
f = @(x)x.^2;
d2f = @(x) 2;
x0 = 0.1;
d2f_FDM = @(h) (f(x0+h)-2*f(x0)+f(x0-h)) ./ (h.^2);
H = logspace(-15,0,16);
h=0.1;
err = abs(d2f_FDM(H) - d2f(x0));
figure('color','w');
%get(h,'PropertyName');
loglog(H, err, H, H.^2, '--');
%set(h,'PropertyName');
%ha=gcf;
%propName= set(ha,'PropertyName');
grid on
legend('ошибка приближения второй производной
x^2', 'h^2')
xlabel('h')
ylabel('ошибка')

%Deriv_2_x.m 2-я производная x
clear all;close all;
f = @(x)x;
d2f = @(x) 0;
x0 = 0.1;
d2f_FDM = @(h) (f(x0+h)-f(x0-h)) ./ (2*h);

```

```

H = logspace(-15,0,16);
h=0.1;
err = abs(d2f_FDM(H) - d2f(x0));
figure('color','w');
loglog(H, err, H, H.^2, '--');
grid on
legend('ошибка приближения второй производной x',
'h^2')
xlabel('h')
ylabel('ошибка')

%ErrorByH.m
%+++++++max| ut-y|=f(h):+++++++
clear all;close all;tic;
H = logspace(-3, 0, 10);fh=zeros(1,length(H));
k=1;
while k<=length(H)
x0=0;x1=1;alp=1;h=H(1,k);
x=x0:h:x1;u0=1;
ut=u0*exp(-alp*x);y=zeros(1,length(x));
y(1)=u0;
for i=2:length(x)
    y(i)=y(i-1)/(1+alp*h);
end
fh(k)=max(abs(ut-y));y(1)=u0;
for i=1:length(x)-1
    y(i+1)=y(i)*(1-alp*h);
end
fh(k)=max(fh(k), max(abs(ut-y)));k=k+1;
end%k
figure('name','Погрешность, как f(h)','Color',
'w');
ha = axes; set(ha, 'FontSize', 14, 'LineWidth', 3,
'GridLineStyle', '-', ...

```

```

    'FontName', 'Times New Roman')
hx = xlabel('{\it h}');set(hx, 'FontSize', 14);
hy = xlabel('{\it max(abs(ut-y))}');set(hy, 'Font-
Size', 14);
loglog(H,fh,'k-');grid on;
xlabel('h');ylabel('max(abs(ut-y))');
toc

```

```

%Stability.m
%u'(x)=-alp*u(x);    x>0;    alp>0;    AccSolution:
u=u0*exp(-alp*x);
clear all;close all;
x0=0;x1=1;alp=1;h=0.1;x=x0:h:x1;u0=1;
%построим точное решение
ut=u0*exp(-alp*x);
figure('name','Устойчивость                разностных
схем','Color','w');
ha = axes;
set(ha, 'FontSize', 14, 'LineWidth', 3, 'GridLin-
eStyle', '-', ...
    'FontName', 'Times New Roman')
hx = xlabel('{\it x}');
set(hx, 'FontSize', 14)
hy = xlabel('{\it y}');
set(hy, 'FontSize', 14)
plot(x,ut,'k-');grid on;
ht =title('Точное и приближенные решения');
set(ht, 'FontSize', 14)
xlabel('x');ylabel('y');hold on;
y=zeros(1,length(x));
%аппроксимация левыми конечными разностями
%(y(i)-y(i-1))/h=-alp*y(i);
y(1)=u0;
%Приводя подобные

```



```

for i=2:length(x)
    y(i)=y(i-1)/(1+alp*h);
end
plot(x,y,'k--x');
%аппроксимация правыми конечными разностями
%(y(i+1)-y(i))/h=-alp*y(i);
y(1)=u0;
%Приводя подобные
for i=1:length(x)-1
    y(i+1)=y(i)*(1-alp*h);
end
plot(x,y,'k-.+');
legend('Точное решение','Левые конечн.разн.','Правые конечн.разн.');
```

hold off;

%Аппроксимация: $Sgm*Left+(1-Sigm)*Right+alp*y(i)=0$;

%3-х точечная схема -> 2 нач.усл.

%Возьмем Sigm симметрично асимптоте: $Sigm=1+/-1$;

%сначала Sigm=0 потом Sigm=2;) и сравним графики

%TestSigma.m

```

clear all;close all;
Sigm=0:0.05:2;
y=Sigm./(Sigm-1);y1=ones(length(Sigm));
figure('name','Гипербола','Color','w')
ha = axes;
set(ha, 'FontSize', 14, 'LineWidth', 1, 'GridLineStyle', '-', ...
    'FontName', 'Times New Roman')

plot(Sigm,y,'k-',Sigm,y1,'k--');grid on;
hx = xlabel('{\it \sigma}');
set(hx, 'FontSize', 14)
xlabel(' \sigma');
```

```

ht=title('\sigma/(\sigma-1)');
set(ht, 'FontSize', 14)

%WeightScheme.m
%+++++++ Схема с весами ++++++
x0=0;x1=1;alp=1;h=0.1;x=x0:h:x1;u0=1;
%построим точное решение
ut=u0*exp(-alp*x);y=zeros(1,length(x));
Sigm=0;%Sigm=2;
y(1)=u0;y(2)=ut(2);
for i=2:length(x)-1
    y(i+1)=(2*Sigm-1+alp*h)/(Sigm-1)*y(i)-
Sigm/(Sigm-1)*y(i-1);
end
figure('name','Устойчивость разностных
схем','Color','w');
ha = axes;
set(ha, 'FontSize', 14, 'LineWidth', 3, 'GridLin-
eStyle', '-', ...
    'FontName', 'Times New Roman')
hx = xlabel('{\it x}');
set(hx, 'FontSize', 14)
hy = ylabel('{\it y}');
set(hy, 'FontSize', 14)
plot(x,ut,'k-o',x,y,'k--s');grid on;
ht =title(['\sigma= ',num2str(Sigm)]);
set(ht, 'FontSize', 14)
legend('Точное решение','Схема с весами');
xlabel('x');ylabel('y');

```

§2. Краевая задача для линейного дифференциального уравнения

Требуется найти решение линейного дифференциального уравнения

$$y'' + p(x)y' + q(x)y = f(x), \quad x \in (a, b), \quad (2.1)$$

удовлетворяющее граничным условиям общего вида

$$Ay'(a) + By(a) = \alpha, \quad Cy'(b) + Dy(b) = \beta, \quad (2.2)$$

где $|A| + |B| \neq 0$, $|C| + |D| \neq 0$.

Применим МКР. Введем на $[a, b]$ равномерную сетку с n узлами

$$\bar{x} = [a + 0 \cdot h, a + h, \dots, a + (i-1)h, \dots, a + (n-1)h], \quad h = (b-a)/(n-1).$$

Обозначим: $y_i = y(x_i)$, $p_i = p(x_i)$, $q_i = q(x_i)$, $f_i = f(x_i)$. Уравнение (2.1) заменяется системой разностных уравнений:

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p_i \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i = f_i, \quad i = \overline{2, n-1}. \quad (2.3)$$

В граничных точках аппроксимируем производные $y'(x)$:

$$y'(a) = \frac{y_2 - y_1}{h} + O(h), \quad y'(b) = \frac{y_n - y_{n-1}}{h} + O(h).$$

Приводя к общему знаменателю, получим для $i = \overline{2, n-1}$:

$$(1 - \frac{h}{2} p_i) y_{i-1} + (h^2 q_i - 2) y_i + (1 + \frac{h}{2} p_i) y_{i+1} = h^2 f_i. \quad (2.4)$$

Граничные условия (2.2) дают недостающие 2 уравнения:

$$(hB - A)y_1 + Ay_2 = h\alpha, \quad -Cy_{n-1} + (hD + C)y_n = h\beta.$$

В итоге получаем СЛАУ с трехдиагональной матрицей

$$Ay = F. \quad (2.5)$$

Известно [1], что аппроксимация решения (2.1), (2.2) порядка h .

Запишем матрицу системы и вектор правой части

$$A = \begin{bmatrix} hB-A & A & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & (1-\frac{h}{2}p_2) & (h^2q_2-2) & (1+\frac{h}{2}p_2) & 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & (1-\frac{h}{2}p_{n-1}) & (h^2q_{n-1}-2) & (1+\frac{h}{2}p_{n-1}) & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -C & hD+C \end{bmatrix},$$

$$F = \begin{bmatrix} h\alpha & h^2 f_2 & \dots & h^2 f_{n-1} & h\beta \end{bmatrix}^T.$$

Отметим, что в матрице A большинство элементов нулевые. Для хранения таких матриц применяются специальные структуры данных, а для решения систем с ними — специальные методы¹. При реализации МКР в программах на MATLAB следует использовать специальную структуру `sparse` для хранения разреженных матриц. Для решения систем можно использовать символ `\`, поскольку автоматически применяется быстрый метод².

Пример 1.

При помощи МКР с шагом $h = 0.25$ решить задачу

$$u'' = x, \quad x \in (0, 1), \quad u(0) = 0, \quad u(1) = 1. \quad (2.6)$$

Здесь

$$p(x) = 0, \quad q(x) = 0, \quad f(x) = x, \quad B = 1, \quad A = 0, \quad \alpha = 0, \quad D = 1, \quad C = 0, \quad \beta = 1.$$

Для проверки, проинтегрировав (2.6) получим точное решение задачи $u_T(x) = (x^3 + 5x)/6$. В аппроксимации краевых условий нет необходимости, поскольку $A = 0$ и $C = 0$. Рассмотрим точность аппроксимации $u''(x)$ в задаче (2.6), воспользовавшись разложением в ряд Тейлора:

$u(x_i \pm h) = u(x_i) \pm u'(x_i)h/1! + u''(x_i)h^2/2! \pm u'''(x_i)h^3/3! + u^{(IV)}(x_i)h^4/4! \pm \dots$
Поэтому $u''_i \approx (u_{i+1} - 2u_i + u_{i-1})/h^2 = u'' + u^{(IV)}(x_i)h^2/12 + \dots$ и, если решение $u(x)$ - полином порядка ниже 4-го, то МКР точен $u''_i = u''(x_i)$.

Совпадающие численное и точное решения задачи (2.6) приведены на рис.2.1.

¹ Один из таких методов — метод прогонки [1] — приведен в приложении к пособию.

² Более подробно о работе с разреженными матрицами говорится в приложении к пособию.

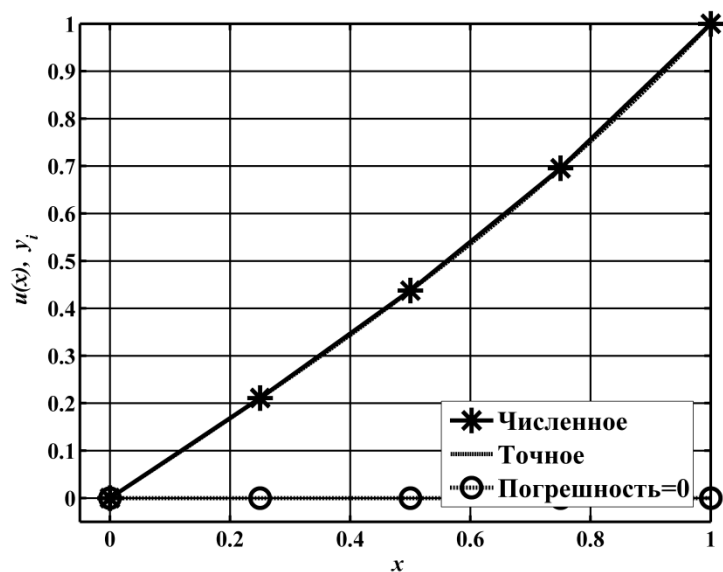


Рисунок 2.1. Точное решение. (MyTest.m, ut.m, f.m)

Пример 2. Используя МКР, для $x \in [0,1]$ с $h = 0.25$, решить

$$u'' = x^2 + 1; \quad u(0) = 0; \quad u(1) = 19/12; \quad u_T(x) = x^4/12 + x^2/2 + x; \quad (2.7)$$

Здесь уже нет совпадения точного и приближенного решения (см. рис. 2.2).

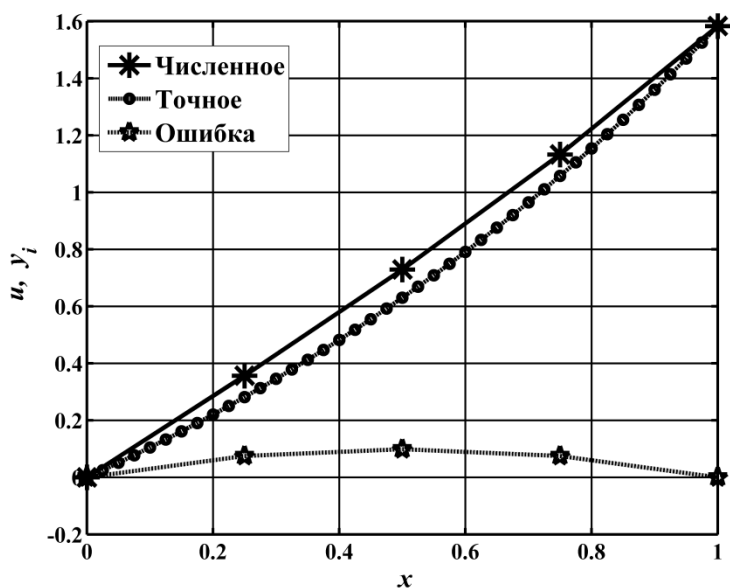


Рисунок 2.2. Точное и приближенное решения. (MyTest.m, ut.m)

Пример 3. Используя МКР, для $x \in [0,1]$ с $h = 0.25$, решить

$$u'' = x; \quad u(0) = 0; \quad u'(1) = 1; \quad u_T(x) = x^3/6 + x/2; \quad (2.8)$$

Построив разностную схему (2.4) для задачи (2.8) и решив (2.5) при $n = 10$, получим следующие результаты (см. рис. 2.3).

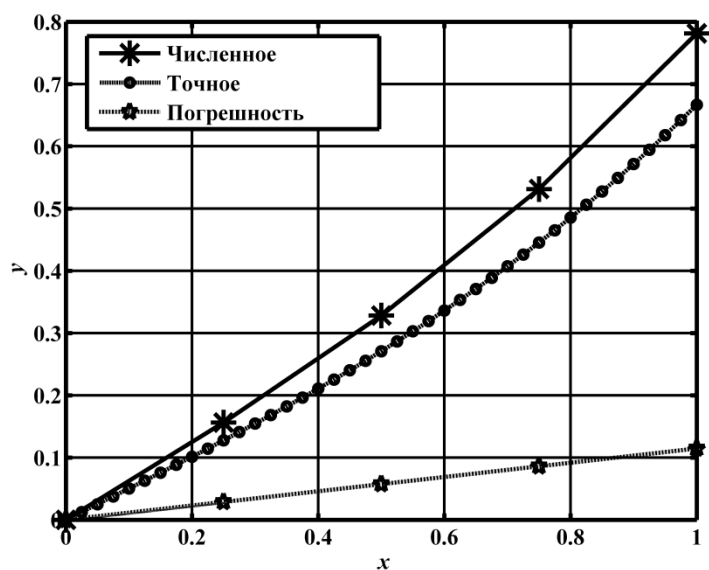


Рисунок 2.3. Решения (2.8) при $n=10$. (MyTest.m, ut.m)
Зависимость ошибки от шага разбиения приведена на рис. 2.4.

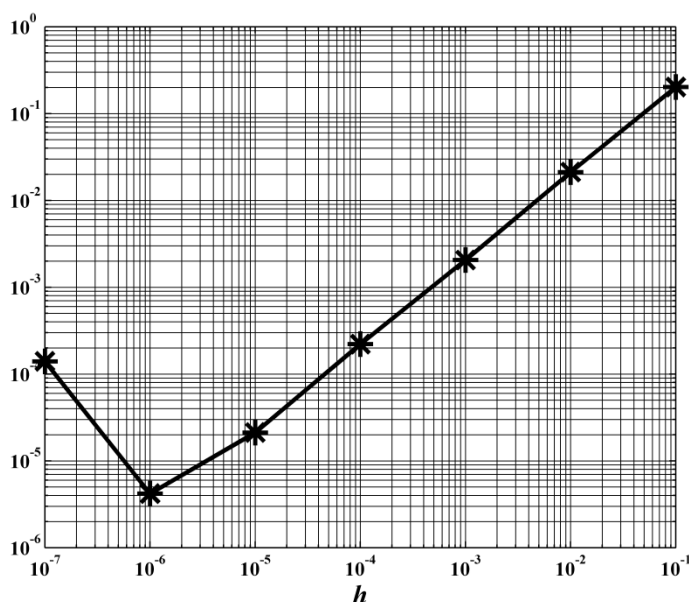


Рисунок 2.4. Ошибка МКР, как функция h . (MyTestConverge.m)

Рассмотрим пример с граничными условиями второго рода.

Пример 4. Используя МКР, для $x \in [0,1]$ с $h = 0.25$, решить

$$u'' = x; \quad u'(0) = 0; \quad u'(1) = 0; \quad (2.9)$$

Здесь $p(x) = 0$; $q(x) = 0$; $B = 0$; $u_T(x) = x^3/3 + C_1x + C_2$ — общее решение уравнения задачи (2.9). Произвольная постоянная C_1 не определяется однозначно. Попробуем решить краевую задачу (2.9).

Матрица A и вектор F системы (2.5) имеют вид

$$A = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}; \quad F = \begin{bmatrix} 0 \\ h^2 x_2 \\ h^2 x_3 \\ h^2 x_4 \\ 0 \end{bmatrix}.$$

То, что $\det(A) = 0$, соответствует сущности поставленной задачи (2.9), у которой имеется бесконечно много решений, отличающихся на константу.

Задачи (в фигурных скобках дано точное решение).

1. $u'' + x^2 u' + \operatorname{tg}^2(x)u = e^x(x^2 + \operatorname{tg}^2(x) + 1)$
 $u(0) = 1; u(1) = e^1; \quad \{u = e^x\}$
2. $u'' + xu' + \sin^2(x)u = \cos(x)(x - \sin(2x)/2)$
 $u(0) = 0; u(1) = \sin(1); \quad \{u = \sin(x)\}$
3. $u'' + e^{2x}u' + \cos(x)u = e^{2x} + x \cos(x)$
 $u(0) = 0; u(1) = 1; \quad \{u = x\}$
4. $u'' + xu' + x^3u = x^5 + 2x^2 + 2$
 $u(0) = 0; u(1) = 1; \quad \{u = x^2\}$
5. $u'' + e^{2x}u' + \cos(x)u = e^{2x}(2e^{2x} + \cos(x) + 4)$
 $u(0) = 1; u(1) = e^2; \quad \{u = e^{2x}\}$
6. $u'' + x^3u' + \cos(x)u = e^x(\cos(x) + x^3 + 1)$
 $u(0) = 1; u(1) = e; \quad \{u = e^x\}$
7. $u'' + \sin(x)u' + \sin(x)u = x^2 \sin(x) + 2x \sin(x) + 2$
 $u(0) = 0; u(1) = 1; \quad \{u = x^2\}$
8. $u'' + \sqrt{x}u' + e^xu = e^x(e^x + \sqrt{x} + 1)$
 $u(0) = 1; u(1) = e^1; \quad \{u = e^x\}$

$$9. \quad u'' + \cos(x)u' + e^x u = e^x (\cos(x) + e^x + 1)$$

$$u(0) = 1; u(1) = e^1; \quad \{u = e^x\}$$

$$10 \quad u'' + x^2 u' + xu = 4x^4 + 6x$$

$$u(0) = 0; u(1) = 1; \quad \{u = x^3\}$$

Листинги к теме «Краевая задача для линейного дифференциального уравнения»

```
%MyTest y' '=x; y(0)=0; y(1)=1; x->[0,1]
clear all;close all;
n=5; x0=0;
x1=1;
%N=[3e0 3e1 3e2 3e3 ]
%Nn=length(N) ;Err=zeros (length(N) ,1) ;
%H=zeros (length(N) ,1) ;
%for in=1:length(N)
    % n=N(in) ;
    h=1/(n-1) ;%H(in,1)=h;
x=x0:h:x1;
A=-2*diag(ones(n,1),0)+...
%diag(ones(n-1,1),1)+...
%diag(ones(n-1,1),-1) ;
A(1,1)=1;%h;
A(1,2)=0;A(n,n-1)=0;
A(n,n)=1;%h;
b=h^2*f(x) ;b(1)=0;
%b(n)=1/3;%b(n)=1;
%b(n)=2/3;%
b(n)=19/12;
y=A\b' ;
yTstx=ut(x) ;
xx=x0:h/10:x1;yTstxx=ut(xx) ;
figure('Color', 'w');ha = axes;
%Err(in)=max(abs(y-yTstx')) ;
```



```

plot(x',y,'k-*',xx,yTstxx,'k--',...
x,y-yTstx','k-.','MarkerSize',14);grid on;
set(ha, 'FontSize', 14, 'LineWidth', 1,...
'GridLineStyle', '-', ...
'FontName', 'Times New Roman');
hx = xlabel('{\it x}');
set(hx, 'FontSize', 14);
hy = ylabel('{\it y}');
set(hy, 'FontSize', 14);
ht = title('Численное и точное решение');
set(ht, 'FontSize', 14);
str=num2str(max(abs(y'-yTstx)) );
hl=legend('Численное','Точное',...
['Погрешность=' str],2);
set(hl, 'FontSize', 14);
%ErrStr=num2str(Err',3);
%end%ih
%semilogx(H,Err);
%ht =title(['Err= ',ErrStr,']');
set(ht, 'FontSize', 12);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%MyTestConverg y' '=x; y(0)=0; y(1)=1; x->[0,1]
clear all;close all;
n=5; x0=0; x1=1;
N=[3e0 3e1 3e2 3e3 ];
Nn=length(N);Err=zeros(length(N),1);
H=zeros(length(N),1);
for in=1:length(N)
    n=N(in); h=1/(n-1);H(in,1)=h;x=x0:h:x1;
A=-2*diag(ones(n,1),0)+...
    diag(ones(n-1,1),1)+...
    diag(ones(n-1,1),-1);
A(1,1)=h;A(1,2)=0;A(n,n-1)=0;A(n,n)=h;
b=h^2*f(x);b(1)=0;

```

```

b(n)=h*19/12;
y=A\b;yTstx=ut(x);
xx=x0:h/10:x1;yTstxx=ut(xx);
figure('Color','w');ha = axes;
Err(in)=max(abs(y-yTstx'));
plot(x',y,'k-',xx,yTstxx,'k--',...
x,y-yTstx','k-.','MarkerSize',12);grid on;
set(ha, 'FontSize', 14, 'LineWidth', 1,...
      'GridLineStyle', '-', ...
      'FontName', 'Times New Roman');
hx = xlabel('{\it x}');
set(hx, 'FontSize', 14);
hy = ylabel('{\it y}');
set(hy, 'FontSize', 14);
ht = title('Численное и точное решение');
set(ht, 'FontSize', 14);
str=num2str(max(abs(y'-yTstx')));
hl=legend('Численное','Точное',...
['Погрешность=' str],2);
set(hl, 'FontSize', 14);
ErrStr=num2str(Err',3);
end%ih
figure('name','Err(h)');
loglog(H,Err);grid on;
ht =title(['Err= [' ,ErrStr,']']);
set(ht, 'FontSize', 12);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function r=f(x)
%подсчет коэффициентов f - Правая часть
r=x;
%r=x.^2+1;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=ut(x)
z=zeros(length(x),1);

```

```

%z=x.^3/6+5*x/6;
%z=x.^3/6+x/2;
z=x.^4/12+x.^2/2+x;
%z=(x.^3+x)/6;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Листинги к задаче 11

```

%MyLinearBoundaryODU
% №11
%x.*(x.^2+6)*u' '-4*(x.^2+3)*u'+7*x.*u=x.^4;
%u(0)=0;u(1)=1;{u=x.^3}
%or u' '+[-4*(x.^2+3)/x.*(x.^2+6)]*u'+
%7*x./x.*(x.^2+6)*u=x.^4./x.*(x.^2+6)
%p(x)=-4*(x.^2+3)./x./(x.^2+6);
%q(x)=7*x./x.*(x.^2+6);
%f(x)=x.^4./x.*(x.^2+6);
clear all;close all;
n=5;
%n=100;
x0=0;x1=1;h=(x1-x0)/(n-1);x=x0:h:x1;x=x';
yt=x.^3;%точное решение
%A=zeros(n,n);
%aa - диагональ A (n -2) внутренн. ур-я
% n1=2;n2=n-1;n3=n2-n1+1;
aa=h^2.*q(x,2,n-1)-2;
% добавим ур-я ГУ
%AA*u'(x0)+BB*y(x0)=alpha
%CC*u'(x1)+DD*y(x1)=beta
%Здесь, в этой задаче
AA=0;BB=1;alpha=0;
CC=0;DD=1;beta=1;
%aaa=[1;aa;1];
aaa=[h*BB-AA;aa;h*DD+CC];
%bb - поддиагональ A

```

```

%bb=zeros(n-2,1);
bb=1-h/2*p(x,2,n-1);
bbb=[bb;-CC];
%cc - наддиагональ A (n-й лишний)
cc=1+h/2*p(x,2,n-1);
ccc=[AA;cc];
% собираем матрицу A
A=diag(aaa,0)+diag(bbb,-1)+diag(ccc,1);
% меняем крайние строки см. гр. условия
%A(1,1)=h;A(n,n)=h;
% % зачищаем A(1,2) и A(n,n-1)
%A(1,2)=0;A(n,n-1)=0;
disp('A=');disp(A);%
% вектор свободных членов
%b=zeros(1,n);
b=h^2*f(x);
b(1)=h*alpha;
b(n)=h*beta;
disp('b=');disp(b);
y=A^(-1)*b;
eps=abs(yt-y);Max=max(eps);
figure('name','LinearODU','color','w');
[AX,H1,H2] = plotyy(x,y,x,eps,'plot');
set(get(AX(1),'Ylabel'),'String',...
'Численное решение')
set(get(AX(2),'Ylabel'),'String',...
'Погрешность')
set(H1,'LineStyle','--','LineWidth',3)
set(H2,'LineStyle',':','LineWidth',3)
%plotyy(x,y,x,eps);
grid on;
xlabel(['x_i; i=1,..., ' num2str(n) '];...
max(\epsilon)=' num2str(Max)]);
legend('y','eps');

```

```

title('Численное решение и его погрешность')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function t=f(x)
%подсчет коэффициентов f - Правая часть
t=x.^3./(x.^2+6);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function t1=p(x,n1,n2)
%подсчет коэффициентов p
t1=zeros(n2-n1+1,1); %t2=zeros(n2-n1+1,1);
t1=-4*(x(n1:n2,1).^2+3)./...
x(n1:n2,1)./(x(n1:n2,1).^2+6);
% for i=n1:n2
% t2(i)=-4*(x(i,1).^2+3)./
% x(i,1).*(x(i,1).^2+6);
% end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%5
function t1=q(x,n1,n2)
%подсчет коэффициента q
t1=zeros(n2-n1+1,1);
t1=7./(x(n1:n2,1).^2+6);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

§3. Решение краевой задачи для нелинейного дифференциального уравнения

Используя МКР, для $x \in [0,1]$ с $h = 0.1$, решить

$$u'' - (x - 2u)u' + 3u = 6x(1 + x^4), \quad u(0) = 0; u(1) = 1; \quad (3.1)$$

для проверки использовать точное решение $u_T(x) = x^3$.

$$\text{Здесь} \quad \begin{cases} p = 2u - x; q = 3; f = 6x(1 + x^4); \\ A = 0; B = 1; \alpha = 0; C = 0; D = 1; \beta = 1. \end{cases} \quad (3.2)$$

Заметим, что $p = p(x, u)$, следовательно, решается нелинейная краевая задача. Положим в качестве начального приближения $u^{(1)} = x$

. Построим итерационную последовательность $\{u^{(k)}\}$; $k = 1, 2, \dots$, из решения следующей задачи при помощи МКР:

$$\frac{d^2 u^{(k+1)}}{dx^2} - (x - 2u^{(k)}) \frac{du^{(k+1)}}{dx} + 3u^{(k+1)} = 6x(1 + x^4), \quad (3.3)$$

$$u^{(k+1)}(0) = 0, u^{(k+1)}(1) = 1.$$

Задавшись условием остановки

$$\|u^{(k+1)} - u^{(k)}\| < \varepsilon, \quad (3.4)$$

получим приближенное решение $u^{(k+1)}$ задачи (3.1).

Введем на $[0, 1]$ равномерную сетку $\{x_i\}$, $i = \overline{1, N}$. Система уравнений (2.5) принимает следующий вид:

$$\begin{cases} y_1^{(k+1)} = \alpha, \\ (1 - \frac{h}{2} p(x_i, y_i^{(k)})) y_{i-1}^{(k+1)} + (h^2 q(x_i) - 2) y_i^{(k+1)} + (1 + \frac{h}{2} p(x_i, y_i^{(k)})) y_{i+1}^{(k+1)} = h^2 f_i, \\ y_{n+1}^{(k+1)} = \beta, \end{cases} \quad (3.5)$$

Будем решать (3.5), увеличивая k на единицу, пока не выполнено условие (3.4), которое имеет вид $\|y^{(k+1)} - y^{(k)}\| < \varepsilon$. Для наглядности положим $\varepsilon = 0.1$, чтобы показать первые 2 итерации (см. рис.3.1).

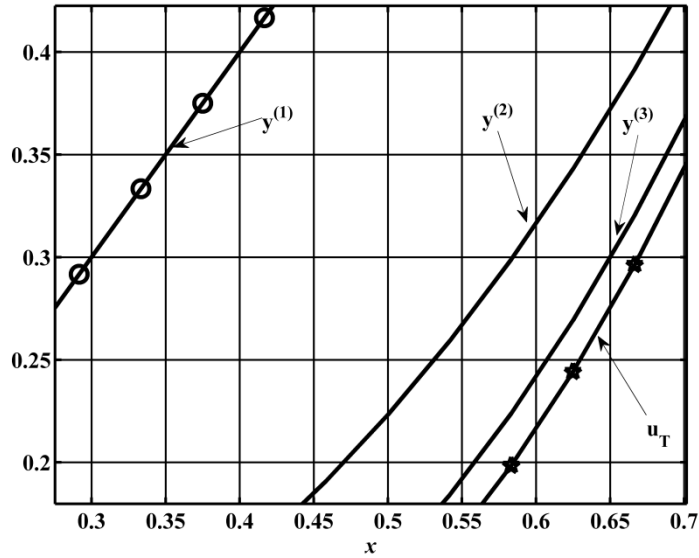


Рисунок 3.1. $y^{(1)} = x$ и две итерации. (MyMainNonLinear.m)

Сходимость решения изучим численно. Сначала для каждого фиксированного значения h получим поведение ошибки в зависимости от количества итераций (см. рис. 3.2). Мы видим, что начиная с некоторого k ошибка перестает уменьшаться (почему?). Далее, построим зависимость ошибки от шага h (см. рис. 3.3). Из графика следует, что $\max \|y_T - y^{(k)}\| \sim h^2$.

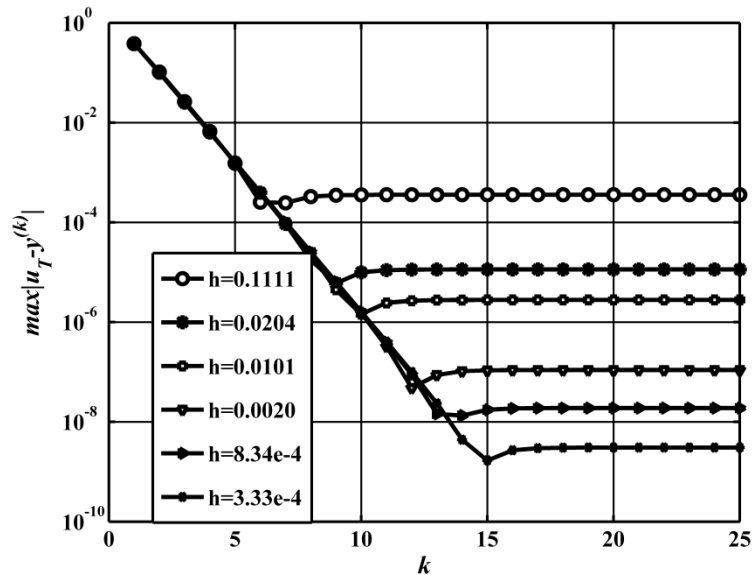


Рисунок 3.2. Поведение ошибки в зависимости от количества итераций (ErrorByIterCount)

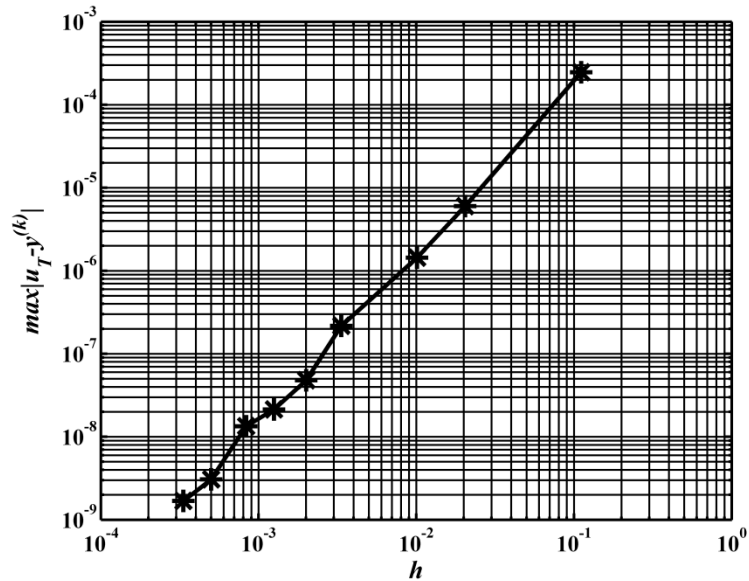


Рисунок 3.3. Зависимость ошибки от шага (ErrorByIterCount)
Задачи (в фигурных скобках дано точное решение).

Методом конечных разностей на отрезке $x \in [0, 1]$ с шагами $h_x = 0.1$ и $h_x = 0.05$ найти численное решение следующих нелинейных краевых задач:

1. $u'' + (u + x^3)u' - 6xu = 6x(x^3 - 1),$
 $u(0) = 0, u(1) = -1; \{u(x) = -x^3\}$
2. $u'' - (u^2 - 0.2)u' - 26u = -5u^3,$
 $u(0) = 0.01, u(1) = 1.48413; \{u(x) = e^{5x} / 100\}$
3. $u'' - 2uu' + 9u = -6u \cos 3x,$
 $u(0) = 0, u(1) = 1.4112; \{u(x) = \sin 3x\}$
4. $u'' - (x - 2u)u' + 3u = 6x(1 + x^4),$
 $u(0) = 0, u(1) = 1; \{u(x) = x^3\}$
5. $u'' - uu' + 16u = 4u \sin 4x,$
 $u(0) = 1, u(1) = -0.65364;$
 $\{u(x) = \cos 4x\}$
6. $u'' - uu' + 2x(1 + x/2)u = 2x(1 + x^3/3),$
 $u(0) = 0, u(1) = 1/3; \{u(x) = x^3/3\}$

7. $u'' - (u - 2)u' + 3x^2u = 6x(1 + x),$
 $u(0) = 0, \quad u(1) = 1; \quad \{u(x) = x^3\}$
8. $u'' - uu' + 5u^2 = 25u,$
 $u(0) = 0.01, \quad u(1) = 1.48413; \quad \{u(x) = e^{5x}/100\}$
9. $u'' + 2uu' + 10u^2 = 25u,$
 $u(0) = 0.01, \quad u(1) = 1.48413, \quad \{u(x) = e^{5x}/100\}$
10. $u'' - (u - 2x)u' + 3x^2u = 6x(1 + x^2),$
 $u(0) = 0, \quad u(1) = 1; \quad \{u(x) = x^3\}$
11. $u'' - u^2u' + 16u = 4u^2 \cdot \sin 4x,$
 $u(0) = 1, \quad u(1) = -0.65364; \quad \{u(x) = \cos 4x\}$
12. $u'' - 2uu' + (1/4 - 6x^2)u = -6x - x^3/4,$
 $u(0) = 0, \quad u(1) = -1; \quad \{u(x) = -x^3\}$
13. $u'' + (u + x^3)u' + 6xu = -3x^2(1 + 2x^2),$
 $u(0) = 1, \quad u(1) = 0; \quad \{u(x) = 1 - x^3\}$
14. $u'' - 2uu' + 3u = 6 \cos x \cdot \sin x (\cos x - u \sin x),$
 $u(0) = 0, \quad u(1) = 0.59582; \quad \{u(x) = \sin^3 x\}$

Листинги к теме 3. Решение краевой задачи для нелинейного дифференциального уравнения

```
function [a, b, c, d]=calc(n, h,A,B,x)
%Вычисление векторов a, b, c, d.
Nx=length(x);
tt=zeros(1,Nx-2);a=zeros(1,Nx);b=a;c=a;d=a;
%Вычисление a
xMid=x(2:Nx-1);tt=1-1/2*h*p(xMid);a=[0 tt -1];
%Вычисление b
xLeft=x(1:n-1);tt=h^2*q(xLeft)-2;
b=[ 1 tt 1];
%Вычисление c
tt=1+1/2*h*p(xLeft);c=[0 tt 0];
%Вычисление d
```

```

        tt=h^2*f(xMid);d=[A    tt B*h ];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Test convergence Err(h)
close all
h=linspace(1e-4,1e0,10);
y=0.1*h.^2;
loglog(h,y,'k-*');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
H =[0.785398163397448 ...
0.078539816339745    0.007853981633974];
Err =[0.015531918720590 ...
0.000001585108468 0.000000000158543];
loglog(H,Err,'k-*');hold on;
loglog(H,H.^4,'r-*');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%ErrorByIterCount
clear all;close all;
x0=0;x1=1;
N=[10 50 100 300 500 800 1200 2000 3000];
%linspace(10,1000,10);
H=(x1-x0)./(N-1);
MaxIter=25;%максимальное к-во итераций
Iters=zeros(1,MaxIter);
Err=zeros(MaxIter,length(N));
for j=1:length(N)
n=N(j);h=H(j);x=x0:h:x1;yOld=x;Yt=x.^3;
A=zeros(n,n);%м-ца СЛАУ
aa=zeros(1,n);
%aa - диагональ A
bb=zeros(1,n);
%bb - поддиагональ A (1-й лишний)
cc=zeros(1,n);
%cc - наддиагональ A (n-й лишний)
b=zeros(1,n);% вектор свободных членов

```

```

yNew=x;%нач.приближение,см. строку 11
Err(1,j)=max(abs(yNew-Yt));
%Err(1,j)=norm(yNew-Yt);
Iters(1)=1;
k=1;% счетчик итераций
while k < MaxIter
    k=k+1;
    yOld=yNew;
    aa=h^2*q(x)-2;%диагональ A
    aa(1)=1;aa(end)=1;
% меняем угловые, см. гр. условия
    bb=1-h/2*p(x,yOld);%bb - поддиагональ A
    bb(1)=[];%1-й лишний,
    bb(end)=0;%последний 0
    cc=1+h/2*p(x,yOld);%наддиагональ A
    cc(end)=[];%последний лишний,
    cc(1)=0;%1-й 0
    % собираем матрицу A
    A=diag(aa,0)+diag(bb,-1)+diag(cc,1);
    ff=f(x);b=h^2*ff;b(1)=0;b(end)=1;
%см. гр. условия
    y=A\b';%решаем СЛАУ
    yNew=y';
    Err(k,j)=max(abs(yNew-Yt));
    %Err(k,j)=norm(yNew-Yt);
    Iters(k)=k;
end%while
end
figure('color','w');
semilogy(Iters,Err');
legend(['h=' num2str(H)]);grid on;
xlabel('k');ylabel('max|u_T-y^(^k^)|');
[ErrMin,kMin]=min(Err);
%loglog(H,Err(end,:));

```

```

figure('color','white');
loglog(H,ErrMin);grid on;
xlabel('h');ylabel('max|u_T-y^(^k^)|');
% loglog(MaxIter,Norms,'k-');
% xlabel('IterCount');
%ylabel('|| y^(^n^+^1^)-y^(^n^)||');
% grid on;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function t=f(x)
%подсчет коэффициентов f - Правая часть
% t=zeros(length(x),1);
t=6*x.*(1+x.^4);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%MyMainNonLinear
clear all;close all;
n=25; %для отладки
%n=100;
x0=0;x1=1;h=(x1-x0)/(n-1);
x=x0:h:x1;
%Проверим условия устойчивости
% 1 о--> q(x)<0 не выполнено, q(x)=3
yOld=x;
pp=p(x,yOld);hCr=2/max(abs(pp));
% 2 о--> hCr=2>=h выполнено
A=zeros(n,n);%м-ца СЛАУ
aa=zeros(1,n);
%aa - диагональ A
bb=zeros(1,n);
%bb - поддиагональ A (1-й лишний)
cc=zeros(1,n);
%cc - наддиагональ A (n-й лишний)
b=zeros(1,n);% вектор свободных членов
yNew=x;%нач.приближение,см. строку 11
Yt=x.^3;eps=1e-1;k=1;% счетчик итераций

```

```

%Выведем первое приближение и точное решение
figure('color','white');plot(x,yNew,'b-
',x,Yt,'k--');grid on;hold on;
nrm=1;%с потолка,лишь бы начался цикл
while nrm>eps
    yOld=yNew;
    aa=h^2*q(x)-2;%диагональ A
    aa(1)=1;aa(end)=1;
% меняем угловые, см. гр. условия
    bb=1-h/2*p(x,yOld);
%bb - поддиагональ A
    bb(1)=[];%1-й лишний,
    bb(end)=0;%последний 0
    cc=1+h/2*p(x,yOld);
%наддиагональ A
    cc(end)=[];%последний лишний,
    cc(1)=0;%1-й 0
    % собираем матрицу A
    A=diag(aa,0)+diag(bb,-1)+diag(cc,1);
    ff=f(x);b=h^2*ff;b(1)=0;b(end)=1;
%см. гр. условия
    %Проверим условия устойчивости
    %    Lmbds=eig(A);
    %    MaxLmbd=max(abs(Lmbds))
    %    if MaxLmbd>1
    %
disp('*****MaxLmbd>1*****');
    %    %    break
    %    end
%Проверим условия устойчивости 3
% о--> диагональное преобладание
%if not...
    (min(abs(aa))>max(abs(bb))+max(abs(cc)))
%disp('** нет диагонального преобладания в A **');

```

```

%      end
      y=A\b';%решаем СЛАУ
      yNew=y';%см. стр. 15
      nrm=norm(yNew-yOld);
      plot(x,yNew,'b-');grid on;hold on;
      k=k+1;
      if k>1000
          break
      end
  endwhile
  plot(x,Yt,'r--');
  xlabel('x');ylabel('u & sequence of y_k');
  title(['Iteration # ',num2str(k),...
    ' norma=',num2str(norm(yNew-yOld)) ]);
  hold off;
  %Выведем окончательный результат и точное ре-
шение
  figure('color','white');
  plot(x,yNew,'b-',x,Yt,'r-');grid on;
  xlabel('x');ylabel('u & last y');
  title(['Iteration # ',num2str(k),' nor-
ma=',num2str(norm(yNew-yOld)) ]);
  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  %MyMainNonLinear
  clear all;close all;
  n=25;x0=0;x1=1;h=(x1-x0)/(n-1);x=x0:h:x1;
  yOld=x;
  pp=p(x,yOld);hCr=2/max(abs(pp));
  % 2 o--> hCr=2>=h выполнено
  A=zeros(n,n);%м-ца СЛАУ
  aa=zeros(1,n);
  %aa - диагональ A
  bb=zeros(1,n);
  %bb - поддиагональ A (1-й лишний)

```

```

cc=zeros(1,n) ;
%cc - наддиагональ A (n-й лишний)
b=zeros(1,n) ; % вектор свободных членов
yNew=x ; %нач.приближение, см. строку 11
Yt=x.^3;eps=1e-1;k=1; % счетчик итераций
nrm=1; %с потолка, лишь бы начался цикл
MaxIter=logspace(0,3,20) ;
Norms=zeros(1,length(MaxIter)) ;
Iter=1;MaxIterIndex=1;
while MaxIterIndex<=length(MaxIter)
while Iter <= MaxIter(MaxIterIndex)
    yOld=yNew;
    aa=h^2*q(x)-2; %диагональ A
    aa(1)=1;aa(end)=1;
    % меняем угловые, см. гр. условия
    bb=1-h/2*p(x,yOld) ;
    %bb - поддиагональ A
    bb(1)=[ ] ; %1-й лишний,
    bb(end)=0; %последний 0
    cc=1+h/2*p(x,yOld) ;
    %наддиагональ A
    cc(end)=[ ] ; %последний лишний,
    cc(1)=0; %1-й 0
    % собираем матрицу A
    A=diag(aa,0)+diag(bb,-1)+diag(cc,1) ;
    ff=f(x) ; b=h^2*ff;b(1)=0;b(end)=1;
    %см. гр. условия
    y=A\b' ; %решаем СЛАУ
    yNew=y' ;
    Iter=Iter+1;
end%while
Norms(MaxIterIndex)=norm(yNew-yOld) ;
MaxIterIndex=MaxIterIndex+1;
Iter=1;yNew=x;

```

```

end

figure('color','white');
loglog(MaxIter,Norms,'k-');
xlabel('IterCount');ylabel('|| y^{(n^{+1})} -
y^{(n)} ||');
grid on;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%ODE_bound1case.m
clear all;close all;
A=0; B=1;x0=0; x1=1;
N=[1e1 1e2 1e3 1e4 1e5 1e6 1e7];
Err=zeros(1,length(N));Numb=0;
for i=1:7
n=N(i);u=ones(1,n+1); Numb=Numb+1
h=(x1-x0)/n; xx=x0:h:x1;
[a, b, c, d]=calc(n,h,A,B,xx);
uu=walk1(a, b, c, d,n);
figure('name',...
'Краевая 2-й тип ГУ','color','w');
plot(xx,uu,'r-');title(['n=',num2str(n)]);
grid on;hold on;Ut=xx.^3/6+xx/2;
plot(xx,Ut,'b-');
legend('Численное','Точное',4);
Err(i)=max(abs(xx.^3/6+xx/2-uu));
end
figure('color','w');
loglog(1./N(1:Numb),Err(1:Numb));grid on;
xlabel('h');ylabel('Err');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function t=p(x,yOld)
%подсчет коэффициентов p
t=-(x-2*yOld);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```



```

function t=q(x)
%подсчет коэффициента q
t=3*ones(1,length(x));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% к задаче 3.1
function t=f(x)
%подсчет коэффициентов f - Правая часть
t=zeros(1,length(x));
t=6*x.*(1+x.^4);
t(1)=0;t(end)=1;%ГУ
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%MyMainNonLinear
% Задача:
%Используя МКР, для x из [0,1] , решить
% u''-(x-2u)*u'+3*u=6*x*(1+x^4);
% u(0)=0; u(1)=1;
% для проверки использовать точное решение
%ut(x)=x^3.
clear all;close all;
n=5; %для отладки
%n=100;
x0=0;x1=1;h=(x1-x0)/(n-1);
x=x0:h:x1; x=x';% x теперь столбец
yOld=x;
pp=p(x,yOld);hCr=2/max(abs(pp));
% 2 o--> hCr=2>=h выполнено
A=zeros(n,n);%м-ца СЛАУ
aa=zeros(n,1);%aa - диагональ A
bb=aa;
%bb - поддиагональ A (1-й лишний)
cc=aa;
%cc - наддиагональ A (n-й лишний)
b=aa;
% вектор свободных членов

```

```

yNew=x;%нач.приближение,см. строку 11
Yt=ut(x);eps=1e-1;k=1;% счетчик итераций
nrm=1;
%Норму берем с потолка,лишь бы начался цикл
figure('name','Sequence of y_k(x) and Finit
Result','color','white');
plot(x,yNew,'b-');grid on;hold on;
while nrm>eps
    yOld=yNew;
    aa=h^2*q(x)-2;%диагональ A
    aa(1)=1;aa(end)=1;
% меняем угловые, см. гр. условия
    bb=1-h/2*p(x,yOld);
%bb - поддиагональ A
    bb(1)=[];%1-й лишний,
    bb(end)=0;%последний 0
    cc=1+h/2*p(x,yOld);
%наддиагональ A
    cc(end)=[];
%последний лишний,
    cc(1)=0;%1-й 0
% собираем матрицу A
    A=diag(aa,0)+diag(bb,-1)+diag(cc,1);
    ff=f(x);b=h^2*ff;b(1)=0;b(end)=1;
%см. гр. условия
    %Проверим условие устойчивости
    Lmbds=eig(A);
    MaxLmbd=max(abs(Lmbds));
    if MaxLmbd>1

disp('*****MaxLmbd>1*****');
        % break
    end
    %Проверим условия устойчивости 3 о-->

```

```

%диагональное преобладание
    if
        not
        (min(abs(aa)) > max(abs(bb)) + max(abs(cc)))
    disp('* нет диагонального преобладания в A *');
    end
    y=A\b;%решаем СЛАУ
    yNew=y;%см. стр. 15
    plot(x,yNew,'b-');grid on;hold on;
    nrm=norm(yNew-yOld);
    k=k+1;
    if k>1000
% Ограничение на случай
%расходимости итрац.процесса
        break
    end
end%while
plot(x,Yt,'r--');
xlabel('x');ylabel('u & sequence of y_k');
title(['Iteration # ',num2str(k),...
' norma=',num2str(norm(yNew-yOld)) ]);
hold off;
%Выведем окончательный результат и
%точное решение
figure('name','Finit Result','color','white');
plot(x,yNew,'b-',x,Yt,'r-');grid on;
xlabel('x');ylabel('u & last y');
legend('у_ч_и_с_л(x)','у_т_о_ч_н(x)',4);
title(['Iteration # ',num2str(k),...
' norma=',num2str(norm(yNew-yOld)) ]);
%%%%%%%%%%%%
function t=p(x,yOld);
%подсчет коэффициентов p
t=-(x-2*yOld);
%%%%%%%%%%%%

```

```

function t=q(x)
%подсчет коэффициента q
t=3*ones(length(x),1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function t=ut(x)
t=x.^3;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

§4. Уравнение теплопроводности. Явная схема

Распространение тепла в однородном стержне длины L описывается уравнением теплопроводности:

$$\frac{\partial u(x,t)}{\partial t} = k \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t); \quad t \in (0,T), x \in (0,L), \quad (4.1)$$

где k — коэффициент теплопроводности, $f(x,t)$ — заданная плотность источников тепла. Не умаляя общности, можно положить $k = 1$:

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t); \quad t \in (0,T), x \in (0,L). \quad (4.2)$$

В начальный момент времени задано условие

$$u(x,0) = u_0(x). \quad (4.3)$$

На границах стержня заданы граничные условия, которые в общем случае имеют вид

$$A \frac{\partial u(x,t)}{\partial x} \Big|_{x=0} + Bu(0,t) = \alpha(t), \quad C \frac{\partial u(x,t)}{\partial x} \Big|_{x=L} + Du(L,t) = \beta(t). \quad (4.4)$$

Если в граничное условие (4.4) входит только u , то это означает поддержание заданной температуры на торце стержня (граничное условие 1-го рода). Вхождение в (4.4) лишь $\partial u / \partial x$ соответствует заданному потоку тепла через торец стержня (граничное условие 2-го рода). Граничное условие, содержащее u и $\partial u / \partial x$, соответствует теплообмену с окружающей средой заданной температуры $\theta(t)$ по закону Ньютона с коэффициентом теплообмена λ (граничное условие 3-го рода): $\partial u(x,t) / \partial x|_{x=0} = \lambda(u(0,t) - \theta(t))$.

Решим задачу (4.2) – (4.4) с помощью явной разностной схемы. Введем равномерную сетку в области $t \in [0, T]$, $x \in [0, L]$: $(x_i, t_k)_{i=\overline{1, N}, k=\overline{1, M}}$ с шагом τ по времени и h по координате. Обозначим через u_i^k (для $i = \overline{1, N}; k = \overline{1, M}$) значения приближенного решения.

Для аппроксимации уравнения воспользуемся четырехточечным шаблоном, приведенным на рис. 4.1.

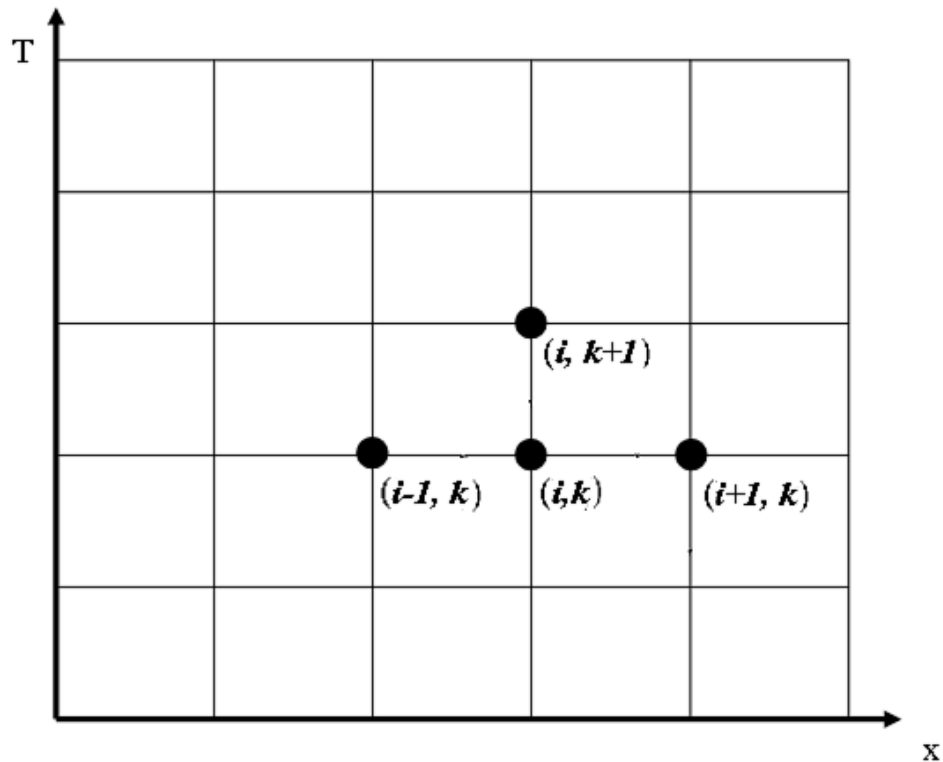


Рисунок 4.1 Шаблон явной схемы

Первую производную по t аппроксимируем правой конечной разностью с порядком $O(\tau)$.

$$\frac{\partial u(x_i, t_k)}{\partial t} \approx \frac{u_i^{k+1} - u_i^k}{\tau}.$$

Вторую производную по x для внутренних узлов аппроксимируем центральной конечной разностью с порядком $O(h^2)$.

$$\frac{\partial^2 u(x_i, t_k)}{\partial x^2} \approx \frac{u_{i-1}^k - 2u_i^k + u_{i+1}^k}{h^2}.$$

Аппроксимируем уравнение (4.1)

$$\frac{u_i^{k+1} - u_i^k}{\tau} = \frac{u_{i-1}^k - 2u_i^k + u_{i+1}^k}{h^2} + f_i^k, \quad (4.5)$$

где $f_i^k = f(x_i, t_k)$.

Разрешим (4.5) относительно u_i^{k+1} , получим

$$u_i^{k+1} = u_i^k + \frac{\tau}{h^2} (u_{i-1}^k - 2u_i^k + u_{i+1}^k) + \tau \cdot f_i^k. \quad (4.6)$$

Обозначим $\alpha_i(t_k) = \alpha_k$, $\beta(t_k) = \beta_k$. Граничные условия после аппроксимации производных с порядком $O(h)$ дадут

$$u_1^{k+1} = \frac{Au_2^{k+1} - h\alpha_{k+1}}{A - hB}; \quad u_N^{k+1} = \frac{Cu_{N-1}^{k+1} + h\beta_{k+1}}{C + hD}. \quad (4.7)$$

Получим явную схему, имеющую порядок аппроксимации $O(h + \tau)$. Если на обеих границах заданы условия 1-го рода, порядок аппроксимации равен $O(h^2 + \tau)$. Условием устойчивости явной схемы является $\tau / h^2 < 1/2$.

Пример 1. Условия Дирихле.

$$\frac{\partial u(x, t)}{\partial t} = \frac{\partial^2 u(x, t)}{\partial x^2} + x \cos(xt) + t^2 \sin(xt); \quad x \in (0, \pi); t \in (0, 1);$$

$$u(x, 0) = 0; \quad u(0, t) = 0; u(\pi, t) = \sin(\pi t); \quad \{u_T(x, t) = \sin(xt)\}$$

Решение на крупной сетке ($h = \pi/8$).

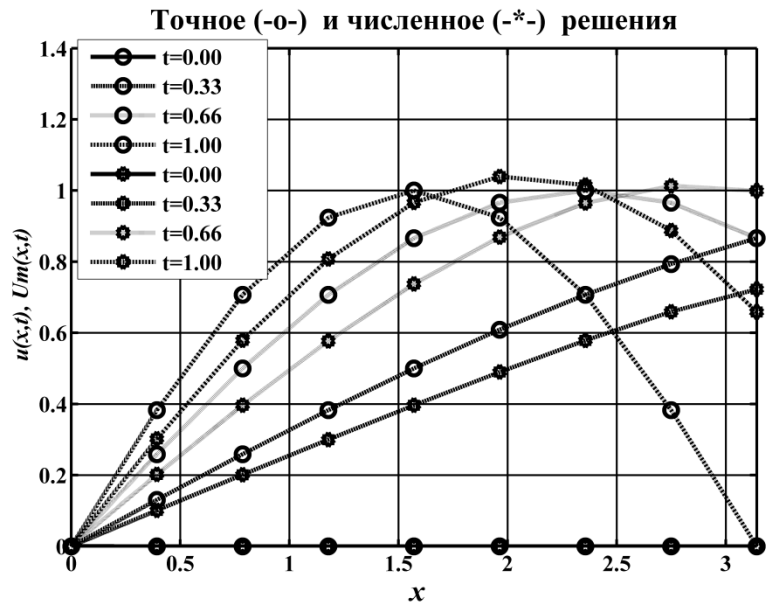


Рисунок 4.2. $u(x_i, t_j)$ и $u_T(x_i, t_j)$. (HeatTrExplDirich.m)

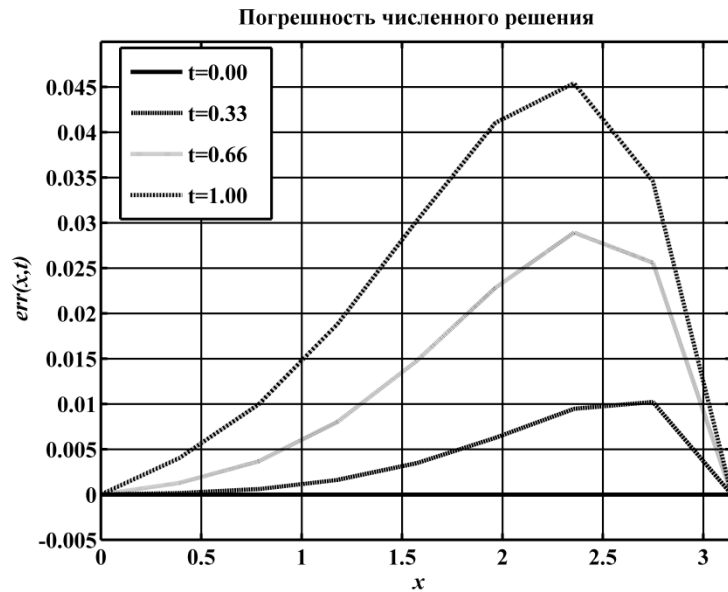


Рисунок 4.3. $|u(x_i, t_j) - u_T(x_i, t_j)|$. (HeatTrExplDirich.m)

Пример 2. Условия Неймана.

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + x \cos(xt) + t^2 \sin(xt); \quad x \in (0, \pi); t \in (0, 1);$$

$$u(x, 0) = 0; \quad \left. \frac{\partial u(x,t)}{\partial x} \right|_{x=0} = t; \quad \left. \frac{\partial u(x,t)}{\partial x} \right|_{x=1} = t \cos t; \quad \{u_T(x,t) = \sin(xt)\}$$

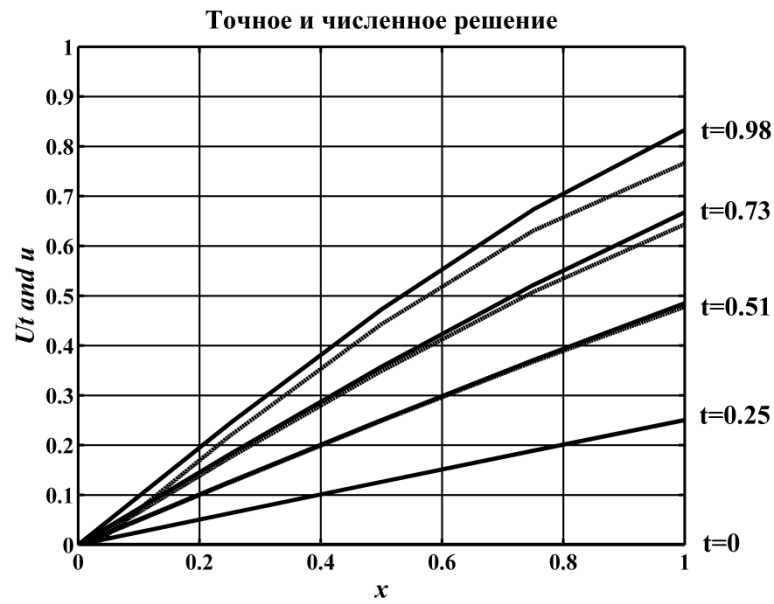


Рисунок 4.4. $u(x_i, t_j)$ и $u_T(x_i, t_j)$. (HeatTrExplNeumann.m)

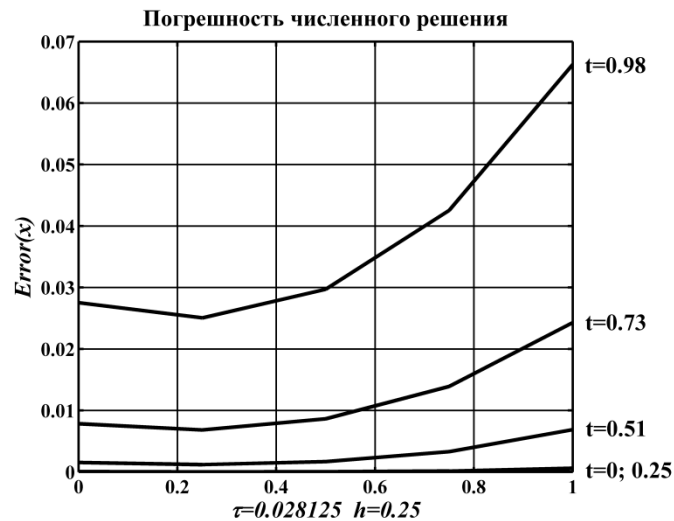


Рисунок 4.5. $|u(x_i, t_j) - u_T(x_i, t_j)|$. (HeatTrExplNeumann.m)

Пример 3. Условия общего вида.

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + 3\sin x + 4t; \quad t \in (0, T), x \in \left(0, \frac{\pi}{2}\right);$$

$$\left. \frac{\partial u(x,t)}{\partial x} + u(x,t) \right|_{x=0} = 3 + 2t^2, \quad \left. \frac{\partial u(x,t)}{\partial x} + u(x,t) \right|_{x=\frac{\pi}{2}} = 3 + 2t^2;$$

$$u(x,0) = 3\sin x; \quad \{u_T(x,t) = 3\sin x + 2t^2\}$$

Решение. Определяем $u_2^k, \dots, u_{N-1}^k$ по шаблону (рис. 4.1), затем находим значения в граничных точках по (4.7), где $\alpha_k = 3 + 2t_k^2$, $\beta_k = 3 + 2t_k^2$. На крупной сетке ($h = \pi/8$; $\tau = h^2/3$) получим:

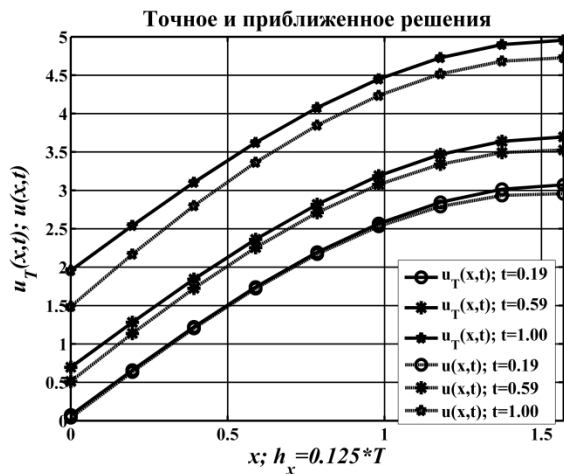


Рисунок 4.6. (HeatTrExplGenCond.m)

Погрешность численного решения:

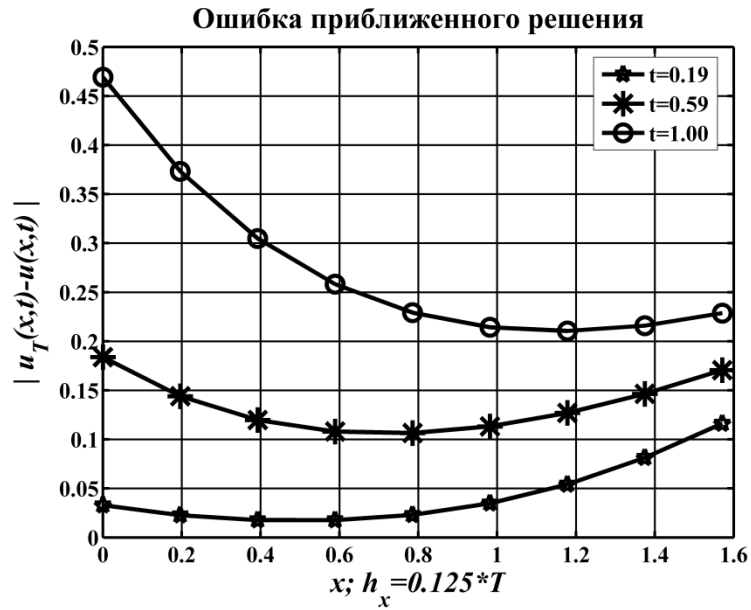


Рисунок 4.7. $|u(x_i, t_j) - u_T(x_i, t_j)|$ (HeatTrExplGenCond.m)

Пример 4. Неустойчивая схема.

Нарушим условие устойчивости в примере 1, взяв $h=0.2$,
 $\tau = h^2 \cdot (0.5 + 0.3)$

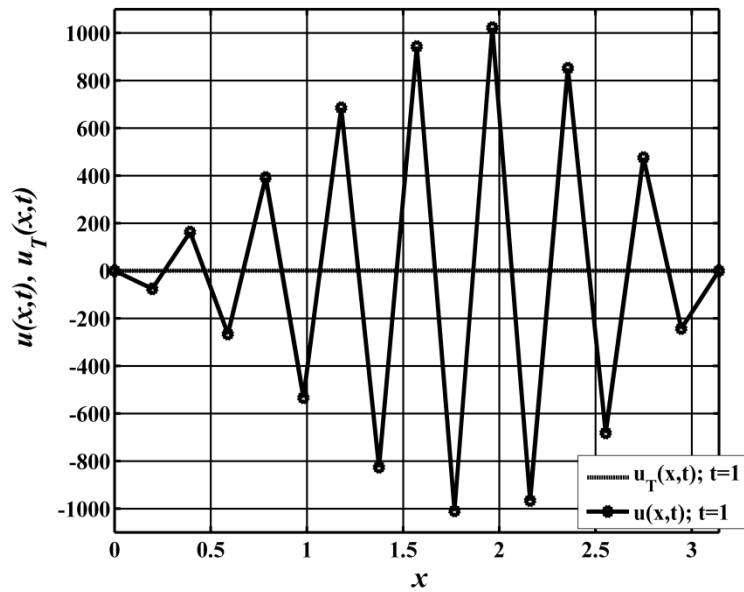


Рисунок 4.8. (HeatTrExplDirich)

◆ Задачи с граничными условиями 1-го рода.

1	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + x \cos(xt) + t^2 \sin(xt); \quad \{u_T(x,t) = \sin(xt)\}$ $u(x,0) = 0; \quad u(0,t) = 0; u(1,t) = \sin t$
2	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + t^2 \cos(xt) - x \sin(xt); \quad \{u_T(x,t) = \cos(xt)\}$ $u(x,0) = 1; \quad u(0,t) = 1; u(1,t) = \cos t$
3	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + e^{(xt)}(x - t^2); \quad \{u_T(x,t) = e^{(xt)}\}$ $u(x,0) = 1; \quad u(0,t) = 1; u(1,t) = e^t$
4	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + 3 \sin x + 2; \quad \{u_T(x,t) = 3 \sin(x) + 2t\}$ $u(x,0) = 3 \sin x; \quad u(0,t) = 2t; u(1,t) = 3 \sin(1) + 2t$
5	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + 2(t-1); \quad \{u_T(x,t) = x^2 + t^2\}$ $u(x,0) = x^2; \quad u(0,t) = t^2; u(1,t) = 1 + t^2$
6	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + e^t - 6x; \quad \{u_T(x,t) = x^3 + e^t\}$ $u(x,0) = x^3 + 1; \quad u(0,t) = e^t; u(1,t) = 1 + e^t$
7	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} - 2 \cos(2x) - \sin(t); \quad \{u_T(x,t) = \sin^2(x) + \cos t\}$ $u(x,0) = \sin^2 x + 1; \quad u(0,t) = \cos t; u(1,t) = \sin^2(1) + \cos t$
8	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + 2(t+x-1); \quad \{u_T(x,t) = (x+t)^2\}$ $u(x,0) = x^2; \quad u(0,t) = t^2; u(1,t) = (1+t)^2$

9	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + 1 - 6x; \quad \{u_T(x,t) = x^3 + t\}$ $u(x,0) = x^3; \quad u(0,t) = t; u(1,t) = 1 + t$
10	$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + (4\pi^2 - 1)e^{-t} \sin(2\pi x); \quad \{u_T(x,t) = \sin(2\pi x)e^{-t}\}$ $u(x,0) = \sin(2\pi x); \quad u(0,t) = 0; u(1,t) = 0;$

Листинги к теме 4. Уравнение теплопроводности. Явная схема.

```

function z=Alpha(t,k)
z=2*t(k)^2+3;
%z=sin(t(k));
%z=0;
%z=t(k);
%%%%%%%%%%
function z=Beta(t,k)
%Beta.m
%z=zeros(length(t));
z=3+2*t(k)^2;
%z=sin(t(k));
%z=t(k)*cos(t(k));
%z=sin(pi*t(k));
%%%%%%%%%%
function z=f
%f.m
z='3*sin(x)+4*t';
%z='x.*cos(x.*t)+t.^2.*sin(x.*t)';
%%%%%%%%%%
function HeatPrn(NewFig,x,t,U,NumOfPlots,...
TypeLin,... MyXlabel,MyYlabel,MyTitle)
%HeatPrn.m Печать результатов

```

```

%U(x,t) - матрица результатов
%Печать прореженных по x и t графиков %только
маркерами
%TypeLin - for example, 'k-o' - признаки
%кривых;
%Nx=length(x);Nt=length(t);
%XX=x(1:5:end);%Nxx=length(XX);
%TT=t(1:10:end);
if strcmp(NewFig, 'on')
    figure('color','w');
    ha=axes;
    set(ha,'FontSize',14,'LineWidth',1.5,...
        'GridLineStyle','-','FontName',...
        'Times NewRoman');
end
% TypeLinNoMarker=TypeLin(1:end-1);
% TypeLinMarker=TypeLin(1:end);
Nt = length(t);
%NumOfPlots = 5;
NumPlots = linspace(1,Nt,NumOfPlots);
NumPlots = round(NumPlots);
    %график 'k-'
UU=U(:,NumPlots);
hp=plot(x,UU,TypeLin);grid on;hold on;
set(hp,'LineWidth',3);
% hp=plot(x,UU,TypeLinMarker);%график 'o'
% set(hp,'LineWidth',3,'MarkerSize',12,...
% 'MarkerEdgeColor','k','MarkerFaceColor','w');
hx=xlabel(MyXlabel);set(hx,'FontSize',14);
hy=ylabel(MyYlabel);set(hy,'FontSize',14);
ht=title(MyTitle);set(ht,'FontSize',14);
hold off;
%%%%%%%%%%%%
%HeatTransExplicitCommonCond.m (Явная схема)

```

```

% усл.устойчивости tau<=h^2/2;
clear all;close all;
x0=0;x1=pi/2;h=pi/8;x=x0:h:x1;Nx=length(x);
%Постоянные в гран.усл.
T=1;A=1;B=1;C=1;D=1;
tau=h^2/3;t=0:tau:T;Nt=length(t);
u=zeros(Nx,Nt);%м-ца (Nx*Nt)
%Заполняем начальные условия
%и точное решение в узлах
u(:,1)=u0(x);
Ut=ut(x,t);
%получили м-цу точных решений в узлах
Utt=Ut(:,1:4:end);
plot(x,Utt)
% NewFig='on';NumOfPlots=5;
% HeatPrn(NewFig,x,t,Ut,NumOfPlots,...
%      'k- ','x','Ut(x,t) and u(x,t)',
% 'Точное и численное решение');hold on;
u=Solve_uCommon(u,x(:),t(:),f);
utt=u(:,1:4:end);
plot(x,utt)
% NewFig='off';
% HeatPrn(NewFig,x,t,u,NumOfPlots,...
%      'k: ','x','Ut(x,t) and u(x,t)',
% 'Точное и %численное решение');hold off;
err=Ut-u;
NewFig='on';
HeatPrn(NewFig,x,t,err,NumOfPlots,...
      'k- ','x h=' num2str(h) ],...
      'Err(\tau)',...
      'Погрешность численного решения');
BoundErrLeft=err(1,:);
BoundErrRight=err(end,:);
figure('color','w');

```

```

plot(t,BoundErrLeft,'k-',t,BoundErrRight,...
'k--');grid on;
title('Погрешности на границах');
xlabel('t');ylabel('ErrLeft & ErrRight');
legend('ErrLeft','ErrRight');
ErrLastLevel=err(:,end);
figure('color','w');
plot(x,ErrLastLevel,'k-');grid on;
title('Погрешность на последнем слое');
xlabel('x');ylabel('ErrLastLevel');
%%%%%%%%%%%%
%HeatTransExplicitDirichlet.m (Явная схема)
% усл.устойчивости tau<=h^2/2;
% clear all;
close all;
x0=0;x1=pi;h=pi/8;tau=h^2/3;t0=0;
%Постоянные в гран.усл.
T=1;A=0;B=1;C=0;D=1;
x=x0:h:x1;t=t0:tau:T;
Nx=length(x);Nt=length(t);
u=zeros(Nx,Nt);%м-ца (Nx*Nt)
%Заполняем начальные условия
u(:,1)=u0(x);
%и точное решение в узлах
Ut=ut(x,t);
NewFig='on';%новое графич окно
MyXlabel='\it x';
MyYlabel='\it {Ut(x,t)}';
MyTitle='Точное решение';
HeatPrn(NewFig,x,t,Ut,4,'k-',...
MyXlabel,MyYlabel,MyTitle);hold on;
u(:,:)=Solve_u(u,x(:),t(:),f);
% Численное решение найдено
NewFig='off';%Далее вывод в текущее окно

```

```

hold on;
MyXlabel='\it x';
MyYlabel='\it {u(x,t), Ut(x,t)}';
HeatPrn(NewFig,x,t,u,4,'kh',MyXlabel,...
MyYlabel,MyTitle);hold off;
err=u-Ut;% Ошибка
NewFig='on';%новое графич окно
MyXlabel='\it x';MyYlabel='\it {err(x,t)}';
MyTitle='Погрешность численного решения';
HeatPrn(NewFig,x,t,err,4,...
'k-p',MyXlabel,MyYlabel,MyTitle);
%%%%%%%%%%
%HeatTransExplicitNeumann.m (Явная схема)
% усл.устойчивости tau<=h^2/2;
clear all;close all;
x0=0;x1=1;h=0.20;x=x0:h:x1;Nx=length(x);
%Постоянные в гран.усл.
T=1;tau=h^2*(0.5-0.05);t=0:tau:T;Nt=length(t);
u=zeros(Nx,Nt);%м-ца (Nx*Nt)
%Заполняем начальные условия
%и точное решение в узлах
u(:,1)=u0(x);
Ut=ut(x,t);
%получили м-цу точных решений в узлах
%HeatPrn(NewFig,x,t,U,NumOfPlots,TypeLin,
% MyXlabel,MyYlabel,MyTitle)
NumOfPlots = 5;
HeatPrn('on',x,t,Ut,NumOfPlots,'k- ','x',...
'Ut and u','Точное и численное решение');
hold on;
u=Solve_uNoemann(u,x(:),t(:),f);
HeatPrn('off',x,t,u,NumOfPlots,'k: ','x',...
'Ut and u','Точное и численное решение')
hold off;

```



```

err=abs(Ut-u) ;
HeatPrn('on',x,t,err,NumOfPlots,'k- ',...
['\tau=' num2str(tau) ' h=' num2str(h)],...
'Error(x)', 'Погрешность численного решения');
%hold off;
xlabel(['\tau=' num2str(tau) ' h='
num2str(h)]);
ylabel('Err(\tau)');
%%%%%%%%%%
%HeatTransExplicitDirichlet.m (Явная схема)
% усл.устойчивости tau<=h^2/2;
% clear all;
close all;
x0=0;x1=pi;h=pi/16;%tau=h^2/3;
t0=0;tau=h^2*(0.5+0.3);
%Постоянные в гран.усл.
T=1;A=0;B=1;C=0;D=1;
x=x0:h:x1;t=t0:tau:T;Nx=length(x);
Nt=length(t);u=zeros(Nx,Nt);%м-ца (Nx*Nt)
%Заполняем начальные условия
u(:,1)=u0(x);
%и точное решение в узлах
Ut=ut(x,t);
NewFig='on';%новое графич окно
MyXlabel='{ \it x}';MyYlabel='{ \it {U\tau(x,t)}}';
MyTitle='Точное решение';
HeatPrn(NewFig,x,t,Ut,2,'k-',MyXlabel,...
MyYlabel,MyTitle);hold on;
u(:,:)=Solve_u(u,x(:),t(:),f);
% Численное решение найдено
NewFig='off';%Далее вывод в текущее окно
hold on;
MyXlabel='{ \it x}';MyYlabel='{ \it {u(x,t),...
U\tau(x,t)}}';

```

```

HeatPrn(NewFig,x,t,u,2,'kh',MyXlabel,...
MyYlabel,MyTitle);hold off;
err=u-Ut;% Ошибка
NewFig='on';%новое графич окно
MyXlabel='{ \it x}';MyYlabel='\it {err(x,t)}';
MyTitle='Погрешность численного решения';
HeatPrn(NewFig,x,t,err,4,...
'k-p',MyXlabel,MyYlabel,MyTitle);
%%%%%%%%%%
%HeatTransExplicitDirichlet.m (Явная схема)
% усл.устойчивости tau<=h^2/2;
% clear all;
close all;
x0=0;x1=pi;h=pi/8;tau=h^2/3;t0=0;
%Постоянные в гран.усл.
T=1;A=0;B=1;C=0;D=1;
x=x0:h:x1;t=t0:tau:T;
Nx=length(x);Nt=length(t);
u=zeros(Nx,Nt);%м-ца (Nx*Nt)
%Заполняем начальные условия
u(:,1)=u0(x);
%и точное решение в узлах
Ut=ut(x,t);
NewFig='on';%новое графич окно
MyXlabel='{ \it x}';
MyYlabel='\it {Ut(x,t)}';...
MyTitle='Точное решение';
HeatPrn(NewFig,x,t,Ut,4,'k-',MyXlabel,...
MyYlabel,MyTitle);hold on;
u(:,:)=Solve_u(u,x(:),t(:),f);
% Численное решение найдено
NewFig='off';%Далее вывод в текущее окно
hold on;
MyXlabel='{ \it x}';MyYlabel='\it {u(x,t),...

```

```

    Ut(x,t) }';
HeatPrn(NewFig,x,t,u,4,'kh',MyXlabel,...
MyYlabel,MyTitle);hold off;
err=u-Ut;% Ошибка
NewFig='on';%новое графич окно
MyXlabel='{ \it x}';MyYlabel='{ \it {err(x,t)}}';
MyTitle='Погрешность численного решения';
HeatPrn(NewFig,x,t,err,4,'k-p',MyXlabel,...
MyYlabel,MyTitle);
%%%%%%%%%%%%%%
%HeatTrExplGenCnd.m (Явная схема)
% усл.устойчивости tau<=h^2/2;
clear all;close all;
x0=0;x1=pi/2;h=pi/32;x=x0:h:x1;Nx=length(x);
%Постоянные в гран.усл.
T=1;A=1;B=1;C=1;D=1;
tau=h^2*(0.5-0.05);
%tau=h^2*0.3;
t=0:tau:T;Nt=length(t);
u=zeros(Nx,Nt);%м-ца (Nx*Nt)
u(:,1)=u0(x);%Заполняем начальные условия
Ut=ut(x,t);%и точное решение в узлах
    u=Solve_uCommon(u,x(:),t(:),f);
TimesPlot=3;kFirst=round(0.2*Nt);
TimPlt=round(linspace(kFirst,Nt,TimesPlot));
figure('color','w');
plot(x,Ut(:,TimPlt),'k-',x,...
u(:,TimPlt),'k--*');
title('Точное и приближенное решения');
str=...
['x; h_x=0.125*T; t_k=' num2str(t(TimPlt))];
xlabel(str);ylabel('t');grid on;
err=Ut-u;
figure('color','w');

```

```

plot(x,err(:,TimPlt),'k-');
title('Ошибка приближенного решения');
xlabel(str);ylabel('| Ut-u |');grid on;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%HeatTransExplicitNeumann.m (Явная схема)
% усл.устойчивости tau<=h^2/2;
clear all;close all;
x0=0;x1=1;h=0.20;x=x0:h:x1;Nx=length(x);
%Постоянные в гран.усл.
T=1;tau=h^2*(0.5-0.05);t=0:tau:T;
Nt=length(t);u=zeros(Nx,Nt);
%м-ца (Nx*Nt)
%Заполняем начальные условия
%и точное решение в узлах
u(:,1)=u0(x);
Ut=ut(x,t);
%получили м-цу точных решений в узлах
%HeatPrn(NewFig,x,t,U,NumOfPlots,TypeLin,...
    MyXlabel,MyYlabel,MyTitle)
NumOfPlots = 5;
HeatPrn('on',x,t,Ut,NumOfPlots,'k- ',...
    'x','Ut and u','Точное и численное решение');
hold on;
u=Solve_uNoemann(u,x(:),t(:),f);
HeatPrn('off',x,t,u,NumOfPlots,'k: ',...
    'x','Ut and u','Точное и численное решение')
hold off;
err=abs(Ut-u);
HeatPrn('on',x,t,err,NumOfPlots,...
    'k- ',...
    ['\tau=' num2str(tau) ' h=' num2str(h)],...
    'Error(x)','Погрешность численного решения');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=Solve_u(x,t,f)

```

```

%x0=0;x1=pi;h=pi/12;tau=h^2/3;
x=x0:h:x1;T=1;t=0:tau:T;
A=1;B=0;C=1;D=0;
z=zeros(length(x),length(t));ff=z;
h=x(2)-x(1);tau=t(2)-t(1);
f=inline(f,'x','t');Solve_u=1;
for k=1:length(t)% с единицы!
    for i=1:length(x)
        ff(i,k)=f(x(i),t(k));
    end%k
end
for k=1:length(t)-1% с единицы!
    for i=2:length(x)-1
        z(i,k+1)=z(i,k)+tau/h^2*...
        z(i-1,k)-2*z(i,k)+z(i+1,k))...
        +tau*ff(i,k);%f->val(t,x)
    end%k
    u(1,k+1)=(h*Alpha(t,k+1)-A*u(2,k+1))...
    /(h*B-A);
    u(Nx,k+1)=...
    (h*Beta(t,k+1)+C*u(Nx-1,k+1))...
    /(C+h*D);
end%k
Solve_u=Solve_u+1
%%%%%%%%%%
function u=Solve_uCommon(u,x,t,f)
%x0=0;x1=pi;h=pi/12;tau=h^2/3;
x=x0:h:x1;T=1;t=0:tau:T;
%A=0;B=1;C=0;D=1;%Dirichlet
%A=1;B=0;C=1;D=0;%Noemann
A=1;B=1;C=1;D=1;%Common
[m,n] = size(u);ff=zeros(m,n);
h=x(2)-x(1);tau=t(2)-t(1);
f=inline(f,'x','t');

```

```

for k=1:length(t)
    for i=1:length(x)
        ff(i,k)=f(x(i),t(k));
    end%k
end
for k=1:length(t)-1
    for i=2:length(x)-1%с 2-ки, внутр. (...)
        u(i,k+1)=u(i,k)+tau/h^2*...
        (u(i-1,k)-2*u(i,k)+u(i+1,k))...
        +tau*ff(i,k);
    end%k
    u(1,k+1)=(A*u(2,k+1)-h*Alpha(t,k+1))...
    /(h*B-A);%гр. слева
    u(end,k+1)=(C*u(end-
1,k+1)+h*Beta(t,k+1))...
    /(h*D+C);%гр. справа
end%k
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function u=Solve_uNoemann(u,x,t,f)
%x0=0;x1=pi;h=pi/12;tau=h^2/3;
x=x0:h:x1;T=1;t=0:tau:T;
%A=0;B=1;C=0;D=1;%Dirichlet
%A=1;B=0;C=1;D=0;%Noemann
A=1;B=1;C=1;D=1;%Common
[m,n] = size(u);ff=zeros(m,n);
h=x(2)-x(1);tau=t(2)-t(1);
f=inline(f,'x','t');
for k=1:length(t)% с единицы!
    for i=1:length(x)
        ff(i,k)=f(x(i),t(k));
    end%k
end
for k=1:length(t)-1% с единицы!
    for i=2:length(x)-1

```

```

        u(i,k+1)=u(i,k)+tau/h^2*...
        (u(i-1,k)-2*u(i,k)+u(i+1,k))...
        +tau*ff(i,k);
    end%k
    u(1,k+1)=(A*u(2,k+1)-h*Alpha(t,k+1))...
    /(h*B-A);
    u(end,k+1)=(h*Beta(t,k+1)+C*u(end-
1,k+1))...
    /(C+h*D);
end%k
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=u0(x)
%u0.m
%z=zeros(length(x),1);
z=3*sin(x);
%z=zeros(length(x),1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=ut(x,t)
%ut.m
z=zeros(length(x),length(t));
for i=1:length(x)
    for k=1:length(t)
        %z(i,k)=sin(x(i)*t(k));
        z(i,k)=3*sin(x(i))+2*t(k)^2;
    end
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

§5. Уравнение теплопроводности. Неявная схема

Решим краевую задачу для уравнения теплопроводности (4.2)-(4.4) с помощью неявной схемы МКР. Шаблон схемы на сетке $(x_i, t_k)_{i=\overline{1,N}, k=\overline{1,M}}$ с шагом τ по времени и h по координате приведен на рис.5.1.

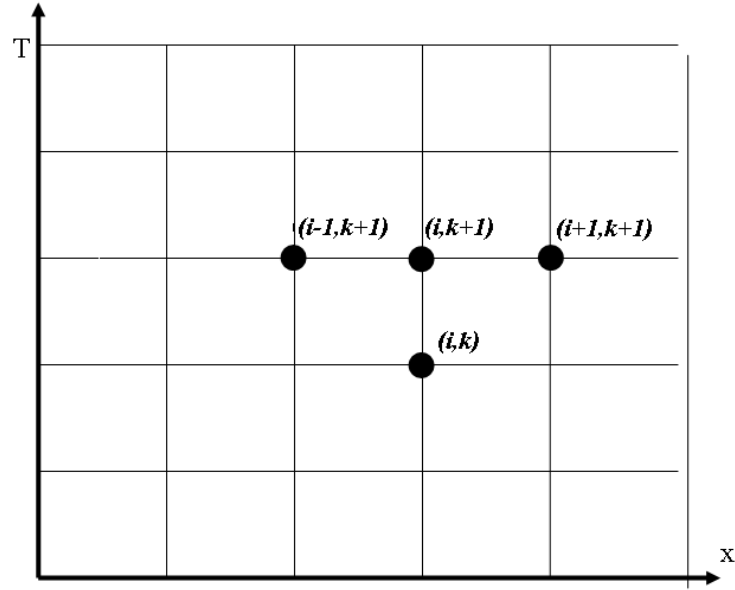


Рисунок 5.1 Шаблон неявной схемы

Обозначим через u_i^k (для $i = \overline{1, N}$; $k = \overline{1, M}$) значения приближенного решения. Аппроксимируем первую производную по t :

$$\frac{\partial u(x_i, t_{k+1})}{\partial t} \approx \frac{u_i^{k+1} - u_i^k}{\tau}.$$

Аппроксимируем вторую производную по x

$$\frac{\partial^2 u(x_i, t_{k+1})}{\partial x^2} \approx \frac{u_{i-1}^{k+1} - 2u_i^{k+1} + u_{i+1}^{k+1}}{h^2}.$$

Получаем для внутренних узлов по x (т.е. для $i = \overline{2, N-1}$):

$$\frac{u_i^{k+1} - u_i^k}{\tau} = \frac{u_{i-1}^{k+1} - 2u_i^{k+1} + u_{i+1}^{k+1}}{h^2} + f_i^{k+1}, \quad (5.3)$$

где $f_i^{k+1} = f(x_i, t_{k+1})$, $k = \overline{1, M-1}$. Преобразуем (5.3), получим:

$$-u_{i-1}^{k+1} + (2 + \frac{h^2}{\tau})u_i^{k+1} - u_{i+1}^{k+1} = \frac{h^2}{\tau}u_i^k + h^2 f_i^{k+1} \quad (5.4)$$

Аппроксимация граничных условий общего вида (4.4) дает

$$\begin{cases} (hB - A)u_1^{k+1} + Au_2^{k+1} = h\alpha_{k+1} \\ -Cu_{N-1}^{k+1} + (hD + C)u_N^{k+1} = h\beta_{k+1} \end{cases} \quad (5.5)$$

Уравнения (5.4) и (5.5) образуют систему из N уравнений относительно N неизвестных $u_1^{k+1}, \dots, u_N^{k+1}$, из которой находится решение на слое $(k+1)$.

Неявная схема устойчива при граничных условиях первого рода

$$u(x, t)|_{x=0} = \alpha(t); \quad u(x, t)|_{x=X} = \beta(t)$$

для любых τ, h с порядком аппроксимации $O(\tau + h^2)$, однако для граничных условий третьего рода (5.2) необходимо выбирать достаточно малые h и τ , чтобы избежать влияния аппроксимации 1-го порядка на границах. $\tau, h \sim 1e-3$ вполне достаточно.

Пример 1. Распространение тепла в стержне. Условия Дирихле. (см. пример 1. из лекции 3)

$$\begin{aligned} \frac{\partial u(x, t)}{\partial t} &= \frac{\partial^2 u(x, t)}{\partial x^2} + x \cos(xt) + t^2 \sin(xt); \quad x \in (0, \pi); t \in (0, 1); \\ u(x, 0) &= 0; \quad u(0, t) = 0; u(\pi, t) = \sin(\pi t); \quad \{u_T(x, t) = \sin(xt)\} \end{aligned}$$

Решение:

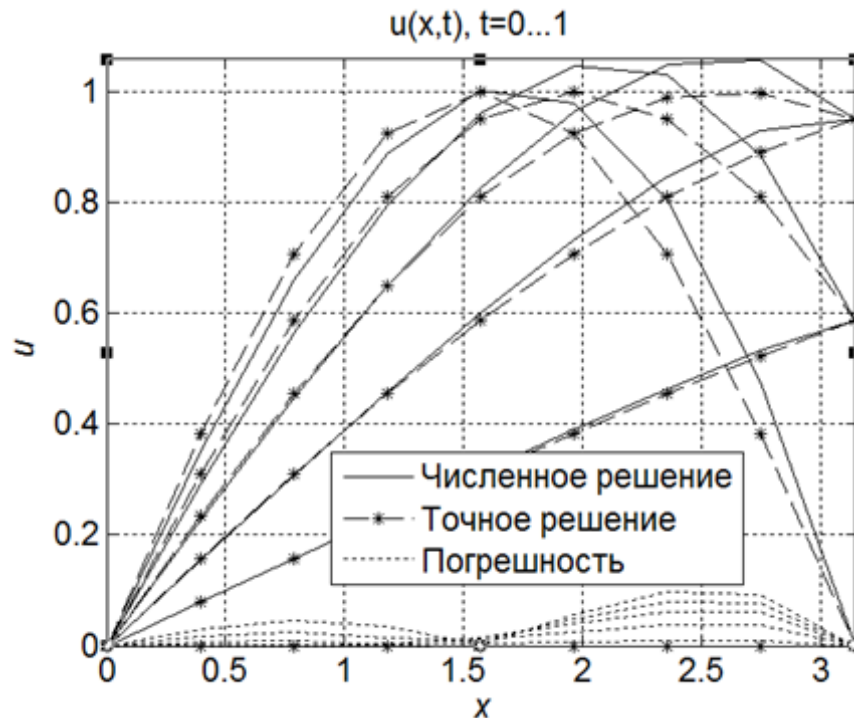


Рисунок 5.2. (HeatTransImplicit.m, $h = \pi/8$; $\tau = 0.2$)

Зависимость погрешности вычислений от величины шага h приведена на рис. 5.4.

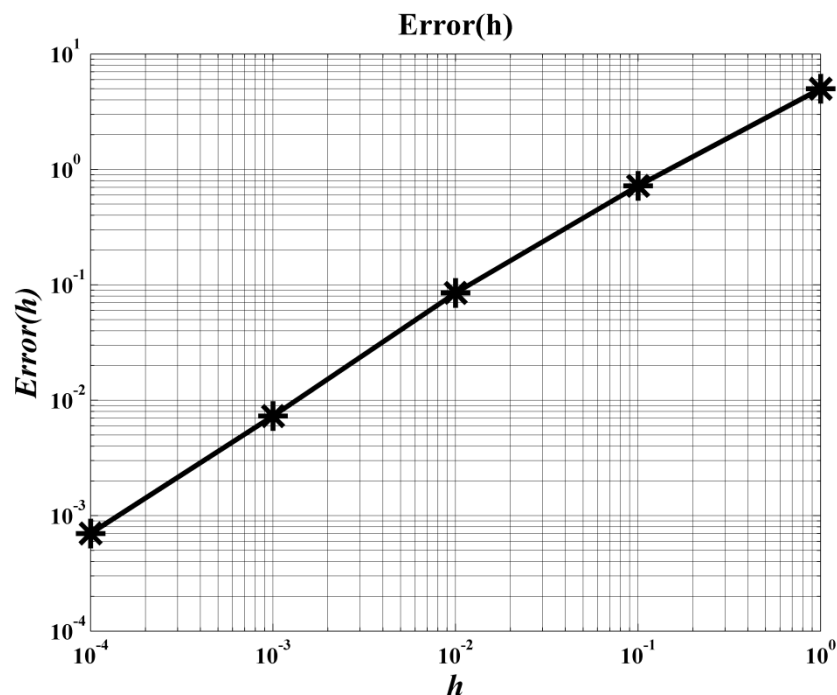


Рисунок 5.4. (HeatTransImpError.m)

♦ Задачи с граничными условиями 1-го рода.

1	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad u(t, x) \in [0, T] \times [0, 1];$ $u(0, x) = 0, u(t, 0) = 0, u(t, 1) = 1;$ $\{u_T(t, x) = x + 2 \sum_{k=1}^{\infty} (-1)^k \cdot (k\pi)^{-1} \cdot \exp(-k^2 \pi^2 t) \cdot \sin(k\pi x)\}$
2	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad u(t, x) \in [0, T] \times [0, 1];$ $u(0, x) = \sin(\pi \cdot x), u(t, 0) = 0, u(t, 1) = 0;$ $\{u_T(t, x) = \exp(-\pi^2 t) \cdot \sin(\pi x)\}$
3	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \sin(\pi x), \quad u(t, x) \in [0, T] \times [0, 1];$ $u(0, x) = 0, u(t, 0) = 0, u(t, 1) = 0;$ $\{u_T(t, x) = \frac{1 - \exp(-\pi^2 t)}{\pi^2} \cdot \sin(\pi \cdot x)\}$

4	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, (x, t) \in (0, 1) \times (0, 0.1);$ $u(x, 0) = \sin(\pi x) + \sin(3\pi x), u(t, 0) = 0, u(t, 1) = 0$ $\{u_T(x, t) = \sin(\pi x) \exp(-\pi^2 t) + \sin(3\pi x) \exp(-9\pi^2 t)\}$
5	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \sin(xt), u(t, x) \in [0, 0.01] \times [0, 1];$ $u(0, x) = \cos(\frac{\pi x}{2}), u(t, 0) = t, u(t, 1) = 0;$
6	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + a(x^2 - t^2) + bxt - 0.378(c - 1.9);$ $u(x, 0) = \alpha \sin \frac{\pi}{2} x + \beta \cos \frac{\pi}{2} x; u(0, t) = \beta(t + 1), u(1, t) = \alpha(t^2 + 1);$ $a = -1.55, b = 2.91, c = 1.08, \alpha = 1.0, \beta = 1.1$
7	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = x \cos(tx) + t^2 \sin(tx), u(t, x) \in [0, T] \times [0, 1];$ $u(0, x) = 0, u(t, 0) = 0, u(t, 1) = \sin(t); \{u_T = \sin(t, x)\}$
8	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = t^2 \cos(tx) + x \sin(tx), u(t, x) \in [0, T] \times [0, 1];$ $u(0, x) = 1, u(t, 0) = 1, u(t, 1) = \cos(t); \{u_T = \cos(t, x)\}$
9	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = e^{xt}(x - t^2), (x, t) \in (0, 1) \times (0, 0.1);$ $u(x, 0) = 1, u(t, 0) = 1, u(t, 1) = e^t; \{u_T = e^{xt}\}$
10	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + 3\sin(xt) + 2, u(t, x) \in [0, 1] \times [0, 1];$ $u(0, x) = 3\sin(x), u(t, 0) = 2t, u(t, 1) = 3\sin(1) + 2t; \{u_T = 2t + 3\sin x\}$
11	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = 2t - 2,$ $u(x, 0) = x^2, u(0, t) = t^2, u(1, t) = 1 + t^2; \{u_T = t^2 + x^2\}$

12	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = e^t - 6x,$ $u(x,0) = x^3 + 1, u(0,t) = e^t, u(1,t) = 1 + e^t; \quad \{u_T = x^3 + e^t\}$
13	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = -2 \cos(2x) - \sin t, \quad \{u_T = \sin^2(x) + \cos(t)\}$ $u(x,0) = \sin^2(x) + 1, u(0,t) = \cos t, u(1,t) = \sin^2(1) + \cos(t);$
14	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = 2t + 2x - 2,$ $u(x,0) = x^2, u(0,t) = t^2, u(1,t) = (1+t)^2; \quad \{u_T = (t+x)^2\}$
15	$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} = 1 - 6x - 2,$ $u(x,0) = x^3, u(0,t) = t, u(1,t) = (1+t); \quad \{u_T = t + x^3\}$

Листинги к теме 5. Уравнение теплопроводности. Неявная схема.

```

function z=Alpha(t,k)
z=3+2*t(k)^2;
%z=sin(t(k));
%z=0;
%z=t(k);
%%%%%%%%%%
function z=Beta(t,k)
%z=zeros(length(t));
z=3+2*t(k)^2;
%z=sin(t(k));
%z=t(k)*cos(t(k));
%%%%%%%%%%
function z=f(x,i,t,k)
z=3*sin(x(i))+4*t(k);
%%%%%%%%%%
%HeatTransImplicit.m (Неявная схема)
clear all;close all;
```

```

x0=0;x1=pi/2;h=pi/8;
T=1;tau=0.2;% устойчива при любых h и tau
x=x0:h:x1;t=0:tau:T;
Nx=length(x);Nt=length(t);
u=zeros(Nx,Nt);ff=u;Ut=u;
%Заполнение f и точного решения
for k=1:Nt
    for i=1:Nx

ff(i,k)=f(x,i,t,k);%4*t(k)+3*sin(x(i));
        Ut(i,k)=ut(x,i,t,k);
    end%i
end%k
%Заполняем нач.и граничн.условия,
% Начальные условия по t:
u(1:Nx,1)=u0(x);
%Граничные слева
u(1,1:Nt)=u1(t);
%Граничные справа
u(end,1:Nt)=ur(t);
%Второй и следующие слои u по времени:
for k=2:Nt
    %одномерная краевая по коорд. x
    %матрица A 3-х диаг. для внутр.точек
    A=zeros(Nx-2,Nx-2);b=zeros(Nx-2,1);
    A(1,1)=2+h^2/tau;A(1,2)=-1;
    A(end,end-1)=-1;A(end,end)=2+h^2/tau;
    b(1)=h^2/tau*u(2,k-1)+h^2*ff(2,k-1)+u(1,k);
    b(end)=h^2/tau*u(Nx-1,k-1)+h^2*...
        ff(Nx-1,k-1)+u(end,k);
    for i=2:length(A)-1
        A(i,i-1)=-1;A(i,i)=(2+h^2/tau);
        A(i,i+1)=-1;
        b(i)=h^2/tau*u(i+1,k-1)+h^2*ff(i+1,k-1);
    end
end

```

```

        end%i (координата)
        y=A\b;
        u(2:Nx-1,k)=y;
    end%k (время)
    % Выведем некоторые u(x,t):
    Mid=length(t)/2;
    figure('name',...
    ['t=',num2str(t([1,Mid,length(t)]))],...
    'color','w');
    for tt =[1,Mid,length(t)] plot(x,u(:,tt),'k--
    ',x,Ut(:,tt),'k-',...
    x,abs(u(1:Nx,tt)-Ut(:,tt)),'k:');
    hold on;
    end
    xlabel('\it x', 'FontSize', 14);
    ylabel('\it t', 'FontSize', 14);
    title('Неявная схема','FontSize', 14);
    hl = legend('Численное решение',...
    'Точное решение','Погрешность',4);
    grid on;hold off;
    % Выведем u(x,t) как поверхности
    % [XX,TT]=meshgrid(0:tau:T,x0:h:x1);
    % Z1=Ut;
    % prt('Точное решение','Ut',TT,XX, Z1);
    % Z2=u;
    % prt('Численное решение','u',TT,XX, Z2);
    % prt('Погрешность','| Ut-u |',...
    TT,XX, abs(Z1-Z2));
    %%%%%%%%%%%%%%%
    %HeatTrImplGenCond.m (Неявная схема)
    clear all;close all;
    x0=0;x1=pi/2;h=pi/8;
    T=1;tau=0.2;% устойчива при любых h и tau
    x=x0:h:x1;t=0:tau:T;

```

```

Nx=length(x);Nt=length(t);
u=zeros(Nx,Nt);ff=u;Ut=u;
%Заполнение f и точного решения
for k=1:Nt
    for i=1:Nx
        ff(i,k)=f(x,i,t,k);
        %4*t(k)+3*sin(x(i));
        Ut(i,k)=ut(x,i,t,k);
    end%i
end%k
% Начальные условия по t:
u(1:Nx,1)=u0(x);
%Граничные слева
u(1,1:Nt)=u1(t);
%Граничные справа
u(end,1:Nt)=ur(t);
%Второй и следующие слои u по времени:
for k=2:Nt
    %одномерная краевая по коорд. x
    %матрица A 3-х диаг. для внутр.точек
    A=zeros(Nx-2,Nx-2);b=zeros(Nx-2,1);
    A(1,1)=2+h^2/tau;A(1,2)=-1;
    A(end,end-1)=-1;A(end,end)=2+h^2/tau;
    b(1)=h^2/tau*u(2,k-1)+h^2*ff(2,k-1)...
    +u(1,k);
    b(end)=h^2/tau*u(Nx-1,k-1)+h^2*...
    ff(Nx-1,k-1)+u(end,k);
    for i=2:length(A)-1
        A(i,i-1)=-1;A(i,i)=(2+h^2/tau);
        A(i,i+1)=-1;
        b(i)=h^2/tau*u(i+1,k-1)...
        +h^2*ff(i+1,k-1);
    end%i (координата)
    y=A\b;

```

```

        u(2:Nx-1,k)=y;
    end%k(время)
% Выведем некоторые u(x,t):
Mid=length(t)/2;
    figure('name',...
    ['t=',num2str(t([1,Mid,length(t)]))],...
    'color','w');
    for tt =[1,Mid,length(t)] plot(x,u(:,tt),'k--
',x,Ut(:,tt),'k-',x,...
    abs(u(1:Nx,tt)-Ut(:,tt)),'k:');hold on;
end
xlabel('\it x','FontSize',14);
ylabel('\it t','FontSize',14);
title('Неявная схема','FontSize',14);
hl = legend('Численное решение',...
    'Точное решение','Погрешность',4);
grid on;hold off;
% Выведем u(x,t) как поверхности
% [XX,TT]=meshgrid(0:tau:T,x0:h:x1);
% Z1=Ut;
% prt('Точное решение','Ut',TT,XX,Z1);
% Z2=u;
% prt('Численное решение','u',TT,XX,Z2);
% prt('Погрешность','| Ut-u |',TT,XX,abs(Z1-
Z2));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=u0(x)
%z=zeros(length(x));
z=3*sin(x);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=ur(t)
z=zeros(length(t));
z=3+2*t.^2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```



```

function z=ut(x,i,t,k)
z=3*sin(x(i))+2*t(k)^2;
%z=x.^3/6+5*x/6;
%z=x.^3/6+x/2;
%z=x.^4/12+x.^2/2+x;
%z=3*sin(x)+2*t.^2;
%%%%%%%%%

```

§6. Колебания струны

Задача Дирихле с однородными граничными условиями.

Будем рассматривать струну как гибкую упругую нить. Понятие гибкости заключается в том, что возникающие в струне напряжения всегда направлены по касательной к ее мгновенному профилю. Струна не сопротивляется изгибу. Пусть $u(x, t)$ отклонение точки x в момент t от горизонтальной оси (O, x) . Если концы струны $x = 0$ и $x = l$ закреплены, то должны выполняться граничные условия

$$u(0, t) = 0; u(l, t) = 0; \quad (6.1)$$

Колебания струны зависят от её начальной формы и скорости каждой её точки в начальный момент $t = 0$.

$$u(x, 0) = \varphi(x); \quad \left. \frac{\partial u(x, t)}{\partial t} \right|_{t=0} = v(x) \quad (6.2)$$

Условия (6.1), (6.2) определяют решение уравнения колебаний струны

$$\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2} + f(x, t); \quad D = (0, l) \times (0, T) \quad (6.3)$$

Уравнение (6.3), начальные (6.2) и граничные (6.1) условия составляют постановку задачи.

Заменим точное решение (6.1-6.3) его приближенным значением в узлах сетки $u(x, t) \approx y(x_i, t_k) = y_i^k; i = \overline{2, Nx-1}; k = \overline{2, Nt}$.

Явная схема МКР

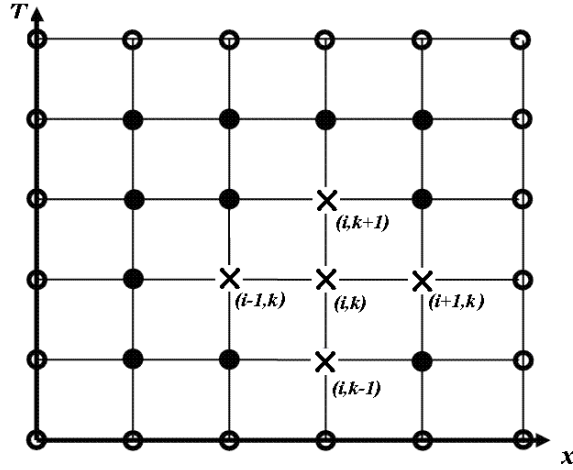


Рисунок 6.1 Пятиточечный шаблон

На границах y_i^1 и y_i^{Nx} решение (см. (6.1)) задано. Разностная схема имеет вид

$$\frac{y_i^{k+1} - 2y_i^k + y_i^{k-1}}{\tau^2} = \frac{y_{i-1}^k - 2y_i^k + y_{i+1}^k}{h^2} + f_i^k.$$

Порядок аппроксимации для внутренних узлов $O(h^2, \tau^2)$. Первый слой по времени ($k=1$) – начальные условия (6.2)

$$y_i^1 = \varphi(x_i);.$$

Второй слой по времени вычисляем, используя известную начальную скорость $v_i = \left. \frac{\partial u}{\partial t} \right|_{t=0}$ с порядком аппроксимации $O(h^2, \tau)$:

$$y_i^2 = y_i^1 + \tau v_i. \quad (6.4)$$

Далее y_i^{k+1} для $k = \overline{2, Nt}$ вычисляются по явной схеме

$$y_i^{k+1} = 2y_i^k - y_i^{k-1} + \frac{\tau^2}{h^2} (y_{i-1}^k - 2y_i^k + y_{i+1}^k) + \tau^2 f_i^k$$

обеспечивающей порядок аппроксимации $O(h^2, \tau)$. Условие устойчивости этой схемы $\tau/h^2 < 0.5$.

Пример 1. Свободные колебания $f(x, t) = 0$.

$$\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2}; \quad D = (0, \pi) \times (0, T) \quad (6.4.1)$$

$$u(x,0) = \varphi(x) = \sin(nx); \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = 0; \quad (6.4.2)$$

$$u(0,t) = 0; \quad u(\pi,t) = 0; \quad (6.4.3)$$

Задача имеет точное решение (см. рис 6.2):

$$u(x,t) = \sin(nx) \cos(\omega \cdot t). \quad (6.4.4)$$

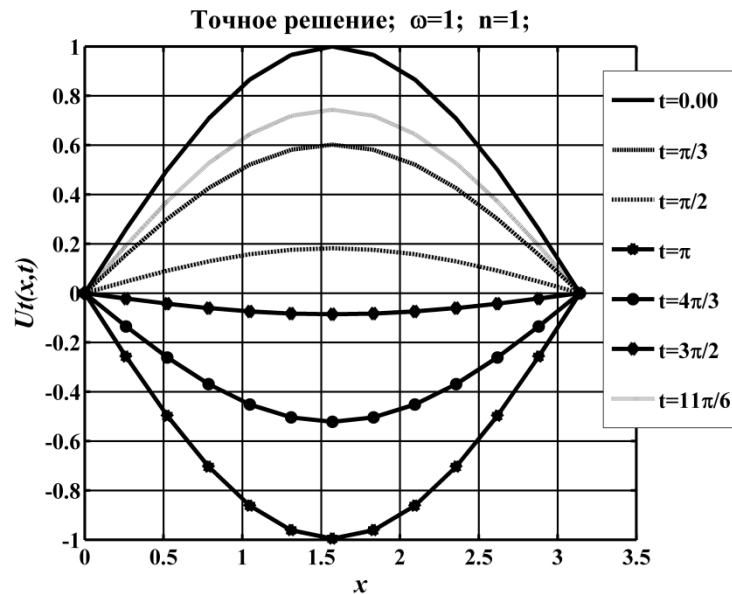


Рисунок 6.2. Первая форма колебаний. (Example1_FreeOscilations.m)

Вторая форма свободных колебаний имеет место при $n = 2$.

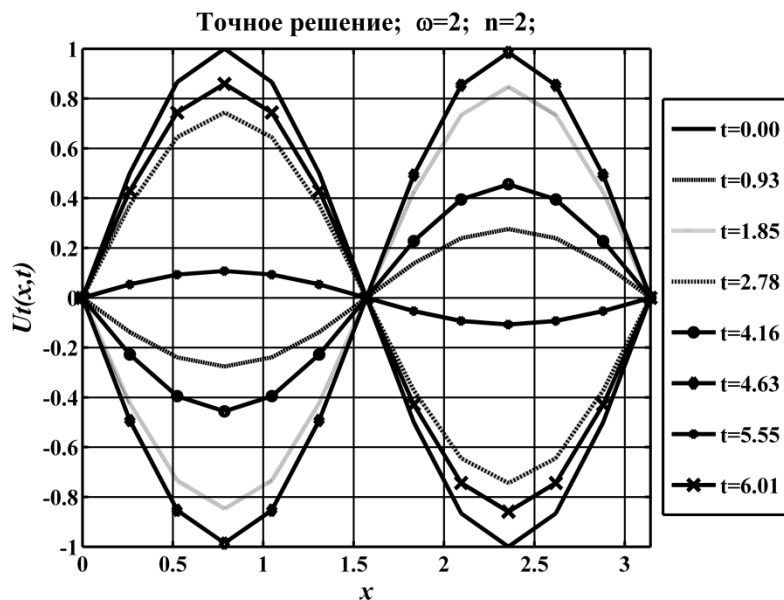


Рисунок 6.3 Вторая форма колебаний. (stringExplicit.m)

Решим теперь поставленную задачу методом конечных разностей. Сравним полученное решение u_i^k с точным $u_T(x_i, t_k)$. Результаты работы программы `stringExplicit.m` при $T = 2\pi$ (рис. 6.4).

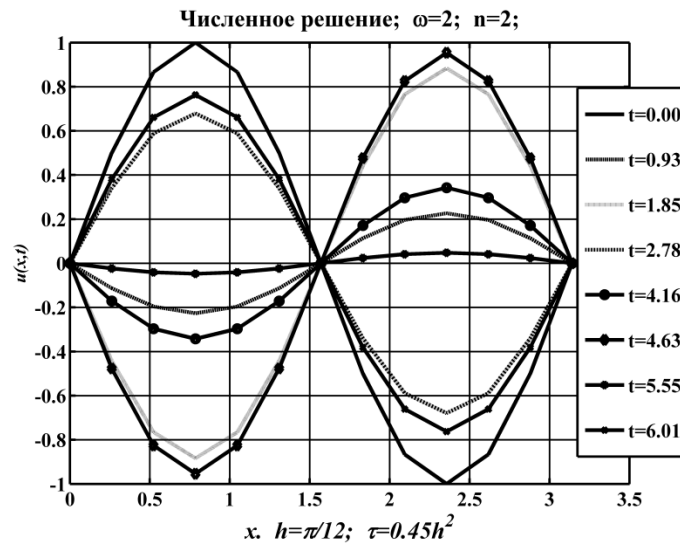


Рисунок 6.4. Численное решение. (`stringExplicit.m`)

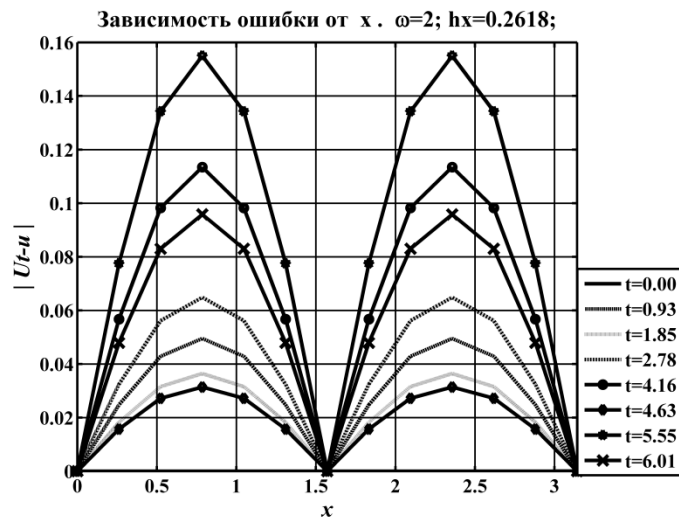


Рисунок 6.5. Наибольшая ошибка: $\max_t |u_T - u|$

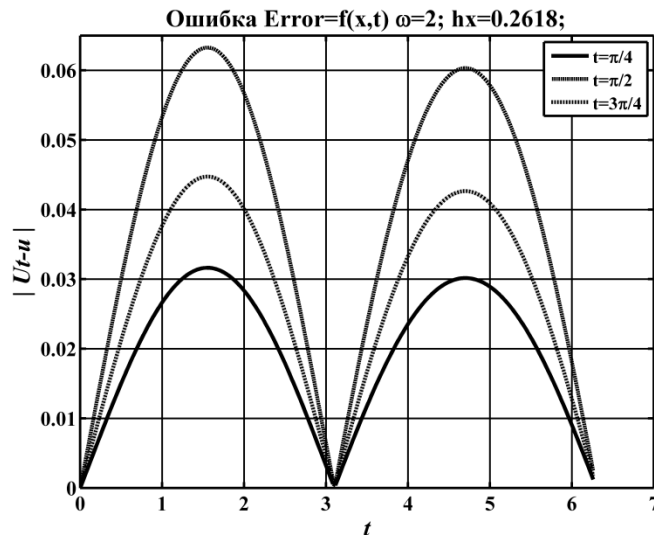


Рисунок 6.6 Погрешность, как функция времени

Получим зависимость погрешности $Err(h) = \max_{i,k} |u_i^k - u_T(x_i, t_k)|$ численного решения от шага h . Пусть $D = (0, \pi) \times (0, \pi/6)$. (Листинг ErrStringShort.m).

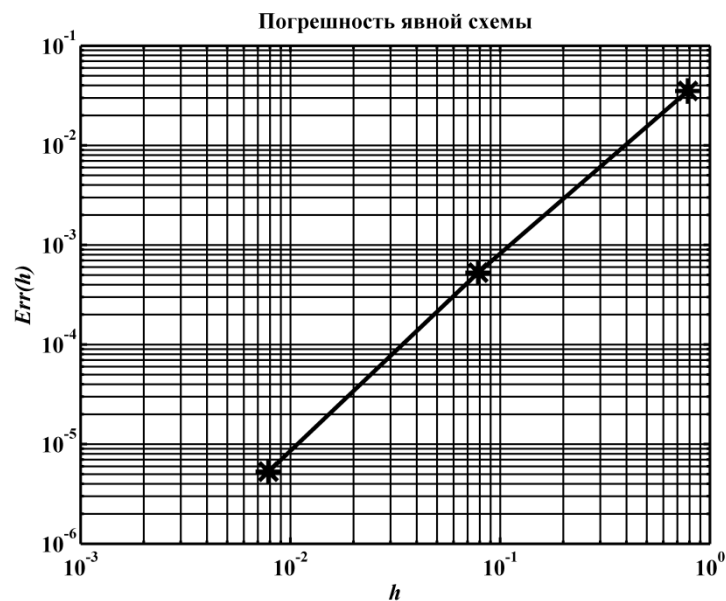


Рисунок 6.7. Зависимость погрешности вычислений от шага h

Неустойчивая явная разностная схема.

Возьмем $h = \pi/12$; $\tau = \pi/6$ в параметрах stringExplicit.m. Это нарушит условие устойчивости. Неустойчивость схемы проявляется не сразу, поэтому увеличим T до $3\pi/2$.

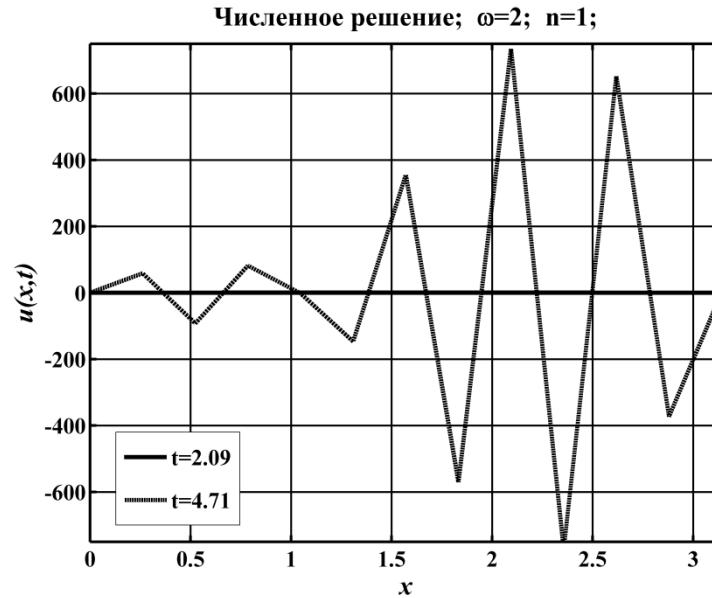


Рисунок 6.8. Неустойчивость явной схемы. $T = 4.7124$

Пример 2. Вынужденные колебания струны. (Листинг StringExplicitForcedOscillation.m)

Решим задачу

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2} + (n^2 - \omega^2) \sin(nx) \cdot \cos(\omega t); \quad D = (0, \pi) \times (0, T). \quad (6.5)$$

$$u(x,0) = \varphi(x) = \sin(nx); \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = 0; \quad (6.6)$$

$$u(0,t) = 0; \quad u(\pi,t) = 0; \quad (6.7)$$

Точное решение задачи (6.5) – (6.7) $u(x,t) = \sin(nx) \cos(\omega \cdot t)$.

Здесь $f(x,t) = (n^2 - \omega^2) \sin(nx) \cdot \cos(\omega t)$; $n \neq \omega$. Рассмотрим первую форму ($n=1$) колебаний при $\omega=2$. Полное колебание совершается за $t = \pi$ единиц времени.

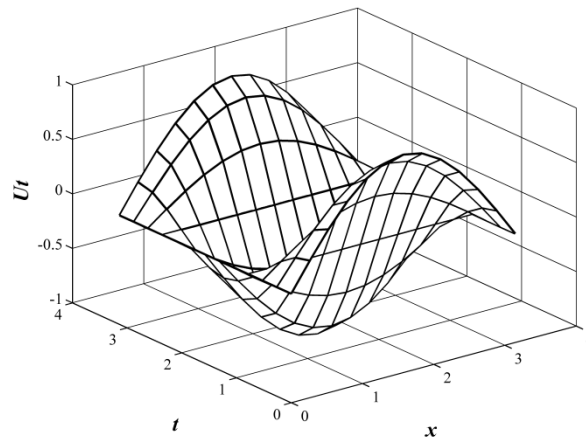


Рисунок 6.9. Численное решение на сетке 13x101

максимальная погрешность Err = 0.0644

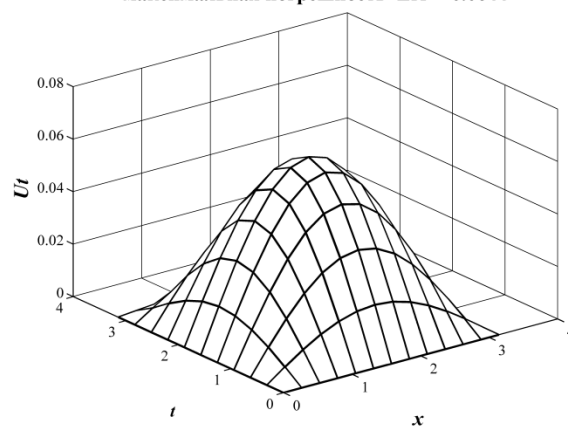


Рисунок 6.10. Погрешность численного решения на сетке 13x101

Пример 3. Резонанс. (Листинг RezonansString.m)

Зададим нагрузку $f(x,t) = (n^2 - \omega^2) \sin(nx) \cdot \cos(\omega t)$; $n=1$, и будем менять $\omega \in [0,2]$ с шагом $h_\omega = 0.1$. Увеличим для наглядности T до $24 \cdot \pi$.

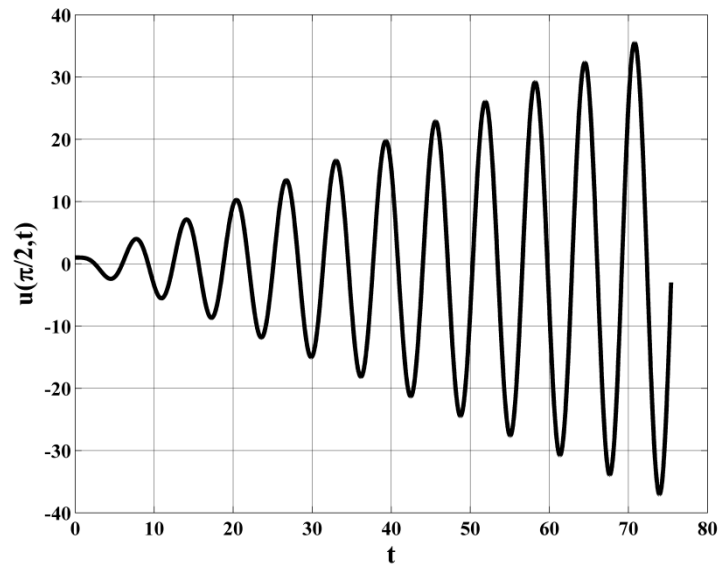


Рисунок 6.11. Движение средней точки струны

Построим график зависимости амплитуды средней точки струны от ω . Получим классическую резонансную кривую.

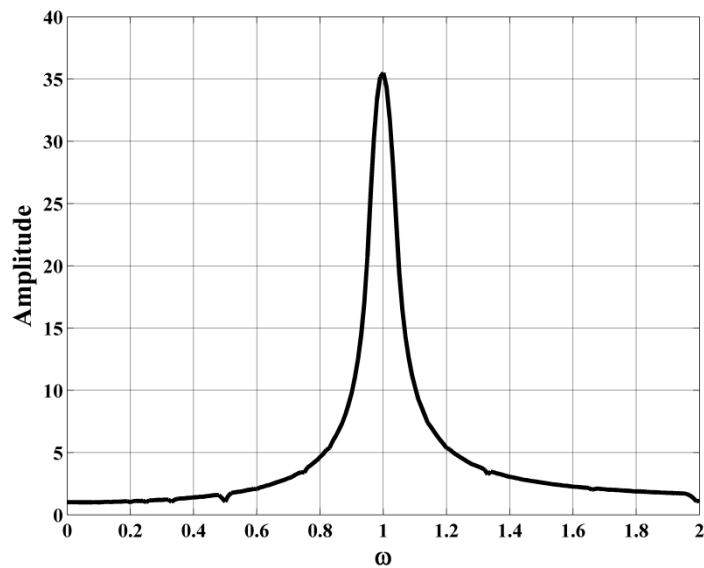


Рисунок 6.12. Резонансная кривая

Резонанс наблюдается при $\omega = 1$. Длина струны (6.5) $l = \pi$. Угловая частота $\omega = \omega_1 = \pi / l$ называется основной, остальные резонансные частоты $\omega_k = k\omega_1$, $k = 2, 3, \dots$ называются частотами обертонов.

Схема с порядком аппроксимации $O(\tau^2 + h^2)$

Будем вычислять решение на втором слое по времени, используя вместо (6.4):

$$\frac{1}{\tau}(y_i^2 - y_i^1) \approx \frac{\partial u(x_{i,0})}{\partial t} + \frac{\tau}{2} \frac{\partial^2 u(x_{i,0})}{\partial t^2}.$$

Выражение для $\frac{\partial^2 u(x_{i,0})}{\partial t^2}$ подставим из уравнения (6.3). Получим

$$y_i^2 = y_i^1 + \tau v_i + \frac{\tau^2}{2} \left(\frac{d^2 \varphi(x)}{dx^2} + f_i^1 \right); \quad 2 \leq i \leq N_x - 1, \quad (6.8)$$

где $\varphi''(x)$ при необходимости заменяется конечной разностью. Далее вычисления проводятся по шаблону «крест» (рис. 6.1).

Можно показать [3], что использование (6.8) и шаблона «крест» сохраняет аппроксимацию $O(\tau^2 + h^2)$ в явной разностной схеме. Необходимым и достаточным условием устойчивости схемы является условие Куранта $\tau < h$.

Задача Дирихле с неоднородными граничными условиями.

Откажемся от условия жесткого закрепления концов струны. Вместо (6.1) рассмотрим задачу (6.2), (6.3) с граничными условиями вида

$$u(0, t) = \alpha(t), \quad u(l, t) = \beta(t). \quad (6.9)$$

Неявная схема МКР.

Для решения задачи (6.1.1) – (6.3) может быть использована [3] безусловно устойчивая неявная схема с шаблоном (6.13). Рассмотрим два разностных оператора

$$\Lambda_x(i, k, h) = \frac{1}{h^2}(y_{i-1}^k - 2u_i^k + u_{i+1}^k); \quad \Lambda_t(i, k, \tau) = \frac{1}{\tau^2}(y_i^{k-1} - 2y_i^k + y_i^{k+1}).$$

$\Lambda_t(i, k, \tau)$ аппроксимирует левую часть (6.3) с порядком $O(\tau^2)$.

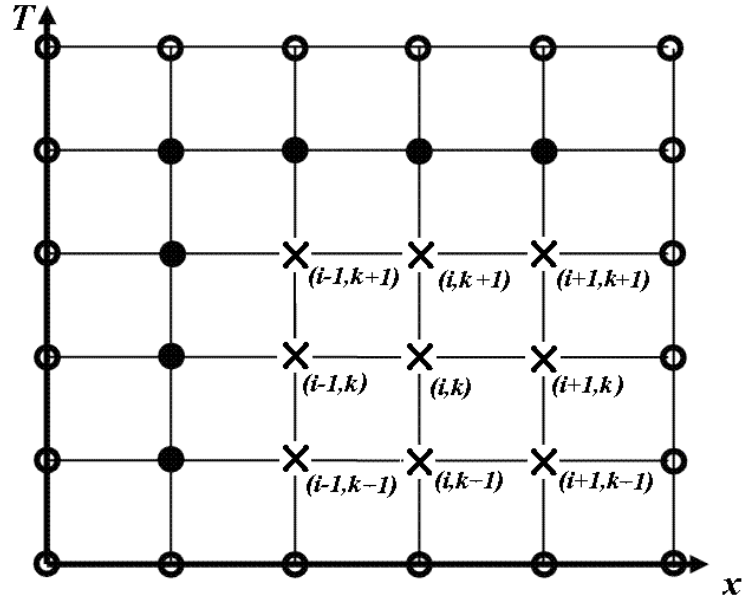


Рисунок 6.13. Шаблон неявной схемы

Производную в правой части (6.3) заменим взвешенной суммой вторых конечных разностей для слоев $(k+1), k, (k-1)$.

$$\frac{\partial^2 u(x, t)}{\partial x^2} \approx \sigma \Lambda_x(i, k+1, h) + (1-2\sigma) \Lambda_x(i, k, h) + \sigma \Lambda_x(i, k-1, h).$$

Получим аппроксимацию уравнения (6.3) с порядком $O(\tau^2 + h^2)$.

$$\Lambda_t(i, k, \tau) = \sigma \Lambda_x(i, k+1, h) + (1-2\sigma) \Lambda_x(i, k, h) + \sigma \Lambda_x(i, k-1, h) + f_i^{k+1} \quad (6.10)$$

При $1/4 \leq \sigma \leq 1/2$ схема (6.10) безусловно устойчива. При $\sigma = 0$ переходит а схему «крест».

Обозначим $\tau/h = \gamma$. Приведя в (6.10) подобные $(k+1)$ -го слоя, получим для внутренних $(2 \leq i \leq N_x - 1)$ точек $N_x - 2$ уравнения

$$\sigma \gamma^2 y_{i-1}^{k+1} - (1+2\sigma \gamma^2) y_i^{k+1} + \sigma \gamma^2 y_{i+1}^{k+1} = F_i^{k+1}; \quad k = \overline{2, N_t - 1},$$

Здесь

$$F_i^{k+1} = h^2 \left\{ y_i^{k-1} - 2y_i^k - \tau^2 \left[(1-2\sigma) \Lambda_x(i, k, h) - \sigma \Lambda_x(i, k-1, h) - f_i^{k+1} \right] \right\}.$$

Недостающие два уравнения дают граничные условия (6.9):

$$y_1^{k+1} = \alpha^{k+1} \text{ и } y_{N_x}^{k+1} = \beta^{k+1}. \text{ Получена СЛАУ } Ay = F, \text{ где}$$

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \sigma\gamma^2 & -(1+2\sigma\gamma^2) & \sigma\gamma^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma\gamma^2 & -(1+2\sigma\gamma^2) & \sigma\gamma^2 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \sigma\gamma^2 & -(1+2\sigma\gamma^2) & \sigma\gamma^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma\gamma^2 & -(1+2\sigma\gamma^2) & \sigma\gamma^2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$y = \begin{bmatrix} y_1^{k+1} & y_2^{k+1} & \dots & y_{N_x-1}^{k+1} & y_{N_x}^{k+1} \end{bmatrix}^T, \quad F = \begin{bmatrix} \alpha^{k+1} & F_2^{k+1} & \dots & F_{N_x-1}^{k+1} & \beta^{k+1} \end{bmatrix}^T$$

Пример 5.4. Для проверки устойчивости схемы (6.10), решим задачу примера 1 с теми значениями параметров $h = \pi/12$ и $\tau = \pi/6$, которые вызывали неустойчивость схемы «крест» (рис. 6.8).

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2}; \quad D = (0, \pi) \times (0, T)$$

$$u(x,0) = \varphi(x) = \sin(nx); \quad \varphi''(x) = -n^2 \sin(nx); \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = 0;$$

$$u(0,t) = 0; \quad u(\pi,t) = 0;$$

Точное решение: $u_t(x,t) = \sin(nx) \cos(\omega \cdot t)$ (рис 6.3).

Полученный результат (рис. 6.14) соответствует точному решению задачи (рис. 6.3), это показывает, что неявная схема (6.10) устойчива (Листинг StringImplicit.m).

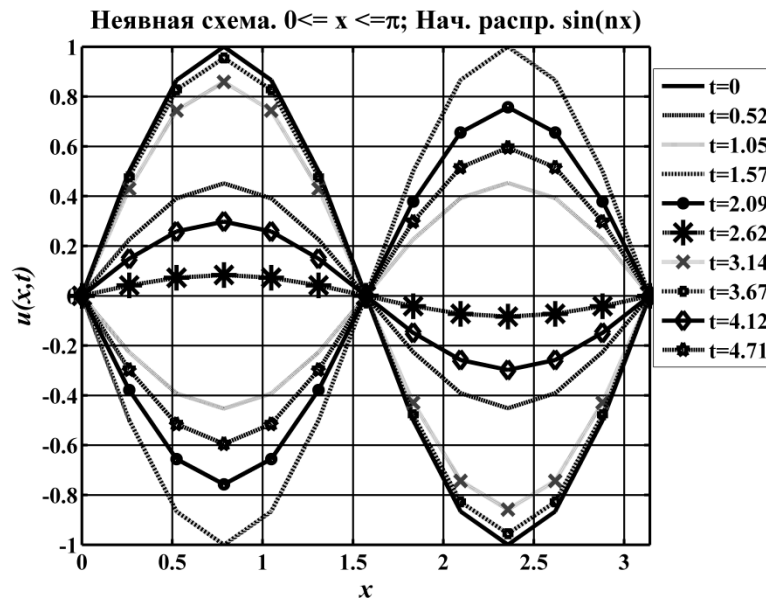


Рисунок 6.14. Устойчивое решение задачи Примера 1

Точность неявной схемы.

Воспользуемся точным решением задачи **примера 4** для получения зависимости $Err(h) = \max_{x,t} |u_T(x,t) - u(x,t)|_h$.

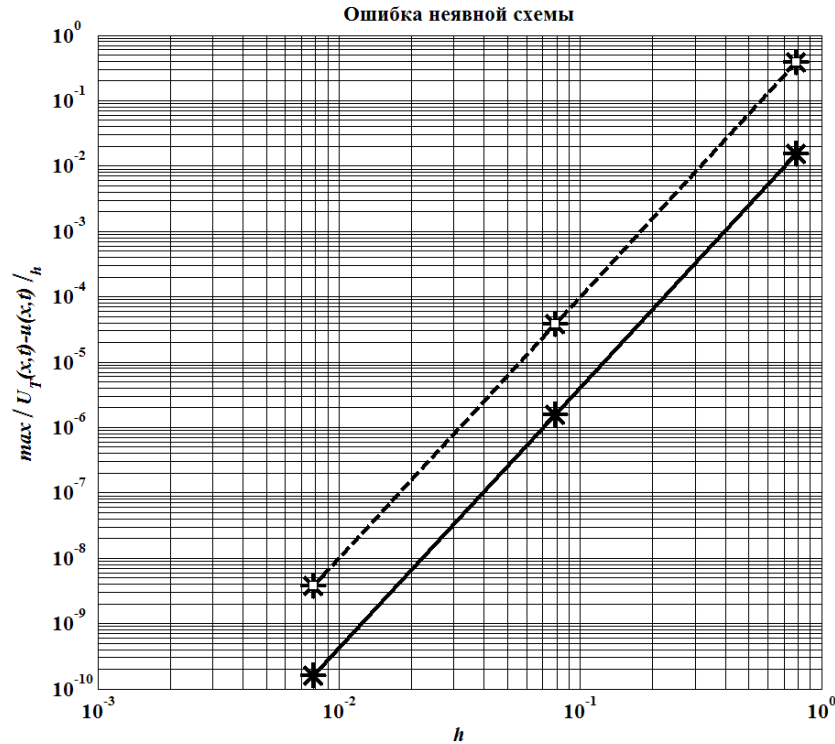


Рисунок 6.15. $Err(h) = \max_{x,t} |u_T(x,t) - u(x,t)|_h$. Пунктиром показана за-

висимость $y = h^4$. (StringImplAccur.m)

Пример 5. Бегущая волна.

Рассмотрим струну без внешней нагрузки [$f(x,t) = 0$].

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2}; \quad D = (0,l) \times (0,T)$$

$$u(x,0) = \varphi(x) = 0; \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = 0;$$

Пусть левый конец струны закреплен, а правый в момент $t = 0$ начинает движение по закону $\beta(t) = A_0 \sin(\omega t)$.

$$u(0,t) = 0; \quad u(l,t) = \beta(t);$$

В данном разделе рассматриваются модели волновых процессов, распространяющихся с единичной скоростью. Поэтому волна достигнет левого конца струны за l единиц времени. Задав время моделирова-

ния $T < l$, получим бегущую по полубесконечной струне волну. (Листинг RunningWave.m).

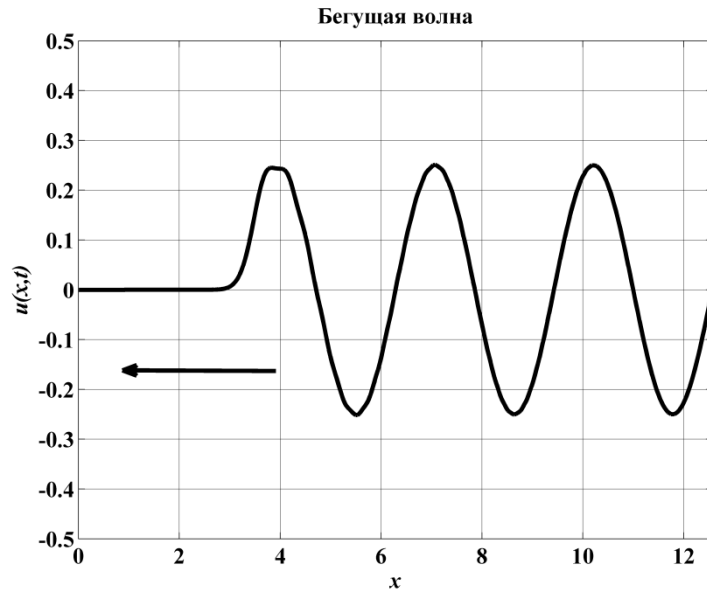


Рисунок 6.16. Бегущая волна $l = 4\pi$, $\omega = 2$, $\lambda = \pi$, $A_0 = 0.25$.

Пример 6. Стоячие волны.

В условиях примера 5 увеличим время моделирования так, чтобы произошло отражение бегущей волны от левого закрепленного конца струны. При отражении волна меняет фазу на π . Отраженная волна налагается на падающую. Происходит сложение двух гармонических колебаний

$$\xi = \xi_1 + \xi_2 = A_0 \{ \sin \omega[t - (l - x)] - \sin \omega[t - (l + x)] \} = 2A_0 \sin \omega x \cos(t - l)$$

Амплитуда колебаний $A = 2A_0 \sin \omega x$ обращается в нуль в точке закрепления, затем в узлах, следующих через $\Delta = \pi / \omega = \lambda / 2$. Здесь λ - длина волны. При заданной частоте ω важно, чтобы в длине струны l укладывалось точно целое число полуволн $\lambda / 2$. В середине между узлами амплитуда A достигает экстремума. Такие значения x называются пучностями. В стоячей волне узлы неподвижны, хотя все остальные точки струны участвуют в колебании (рис 6.17).

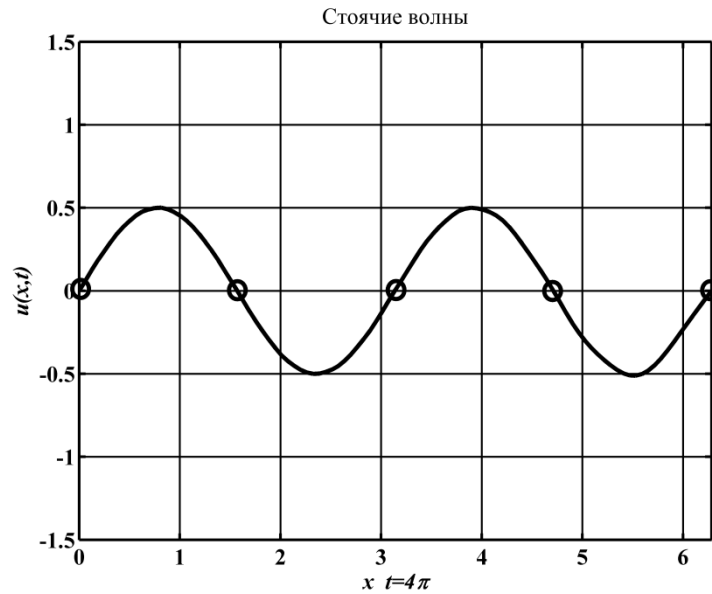


Рисунок 6.17. $l = 2\pi$, $\omega = 2$, $\lambda = \pi$, $A_0 = 0.25$. (RunningWave.m)

Продолжение процесса при $t > 4\pi$ приводит к сложению отраженной волны, возвратившейся к моменту $t = 4\pi$ в точку $x = l$ с возбуждающим колебанием $\beta(t) = A_0 \sin(\omega t)$. Теперь падающая волна имеет амплитуду $A_1 = 2A_0$. После отражения от левого закрепления начнется образование стоячих волн с амплитудой $A = 2A_1 = 4A_0$. В момент $t = 8\pi$ отраженная волна приходит в точку $x = l$ и движение струны представляет собой стоячую волну с амплитудой $A = 2A_1 = 4A_0$ (рис. 6.18). (Листинг RunningWave.m)

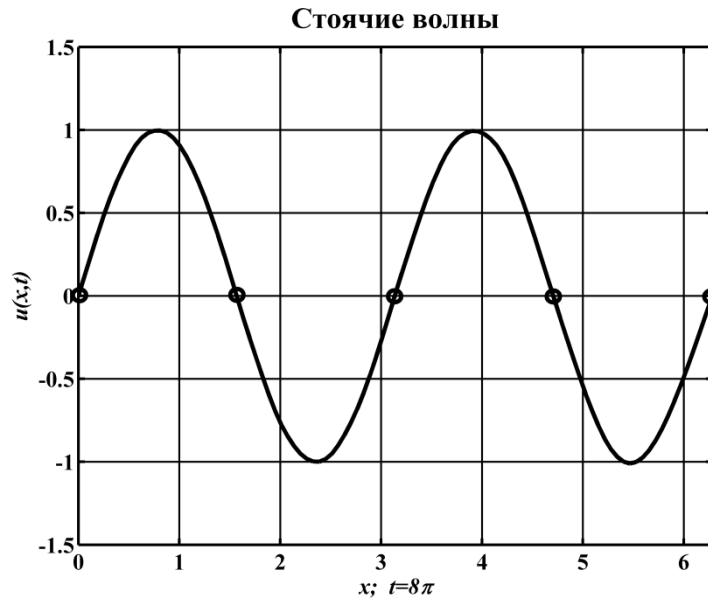


Рисунок 6.18. Стоячая волна $t = 8\pi$, $\omega = 2$, $\lambda = \pi$, $A_0 = 0.25$, $A = 4A_0$.

Далее рассмотренный цикл повторится через каждые $\Delta t = 4\pi$ с увеличением (на A_0) амплитуды падающих (и отраженных) волн из-за подкачки энергии в $x = l$. Амплитуда получаемых стоячих волн вдвое больше амплитуд движущихся.

Если поставить граничное условие

$$\beta(t) = \begin{cases} A_0 \sin(\omega t), & 0 \leq t \leq 4\pi \\ 0, & t > 4\pi \end{cases} \quad \text{где } \omega = 2; A_0 = 0.25,$$

получим стоячую волну $\xi = 2A_0 \sin \omega x \cos(t - l)$ с амплитудой $A_1 = 2A_0$ (рис. 6.17.) и для $t > 4\pi$.

♦ Задачи

1	Явная схема для вынужденных колебаний. Получить численное решение при $h = \pi/12$, $\tau = 0.01\pi$. $\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2} + f(x, t); \quad D = (0, \pi) \times (0, \pi)$ $u(x, 0) = \sin x, \quad u'(x, 0) = 0, \quad f(x, t) = \sin(nx) \cdot \cos(\omega t).$
2	Стоячие волны. Получить численное решение задачи: $\frac{\partial^2 u(x, t)}{\partial t^2} - \frac{\partial^2 u(x, t)}{\partial x^2} = 0; \quad D = (0, 2\pi) \times (0, 2\pi)$

	$u(x,0) = 0; \left. \frac{\partial u}{\partial t} \right _{t=0} = 0; u(0,t) = 0; u(2\pi,t) = \psi(t); \psi(t) = A_0 \sin(\omega \cdot t).$
3	Стоячие волны. Решить задачу 2 для $D = (0,5) \times (0,8\pi)$
4	Стоячие волны. Решить задачу 2 при синфазном колебании концов струны $\psi(t) = A_0 \sin(\omega \cdot t), u(0,t) = \psi(t), u(l,t) = \psi(t).$ $\{\xi = 2A_0 \cos \omega(x - l/2) \sin \omega(t - l/2)\}$
5	Стоячие волны. Решить задачу 2 при противофазном колебании концов струны $\psi(t) = A_0 \sin(\omega \cdot t), u(0,t) = \psi(t), u(l,t) = -\psi(t).$ $\{\xi = 2A_0 \sin \omega(x - l/2) \cos \omega(t - l/2)\}$
6	Нулевые граничные условия и начальная скорость струны. Длина струны $l = 5$. $u(x,0) = \begin{cases} \exp(-1/[4x(x-1)]), & x \in [0,1] \\ 0, & x \notin [0,1] \end{cases}.$ Вывести анимированный график решения.
7	Нулевые граничные условия и начальная скорость струны. Длина струны $l = 5$. Распределение начального отклонения нормальное $N(\sigma, \mu)$. $\sigma = 0.15, \mu = 2.5$. Вывести анимированный график решения.
8	Нулевые граничные условия и начальная скорость струны. $D(x,t) = (-l,l) \times (0,T), l = 5$. $u(x,0) = \begin{cases} P_5(-x), & x \in [-1,0] \\ P_5(x), & x \in [0,1] \\ x \notin [-1,1] \end{cases}, \text{ где } P_5(x) = \sum_{i=1}^6 a_i x^{i-1},$ $a = [1 \ 0 \ 0 \ -10 \ 15 \ -6];$ Вывести анимированный график решения.
9	Нулевые граничные условия и начальная скорость струны. $D(x,t) = (-l,l) \times (0,T)$. $u(x,0) = \begin{cases} P_9(-x), & x \in [-1,0] \\ P_9(x), & x \in [0,1] \\ 0, & x \notin [-1,1] \end{cases}, \text{ где } P_9(x) = \sum_{i=1}^{10} a_i x^{i-1};$

	$a = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -126 & 420 & -540 & 315 & -70 \end{bmatrix}$ Вывести анимированный график решения.
10	Нулевые граничные условия и начальная скорость струны. $D(x,t) = (-l,l) \times (0,T). \quad u(x,0) = \begin{cases} x+1, & x \in [-1,0] \\ 1-x, & x \in [0,1] \\ 0, & x \notin [-1,1] \end{cases}$ Вывести анимированный график решения.
11	Нулевые граничные условия и начальная скорость струны. $D(x,t) = (-l,l) \times (0,T). \quad u(x,0) = \begin{cases} 1, & x \in [-1,1] \\ 0, & x \notin [-1,1] \end{cases}$ Вывести анимированный график решения.
12	Нулевые граничные условия и начальная скорость струны. $D(x,t) = (0,l) \times (0,T), \quad l = 2. \quad u(x,0) = \begin{cases} x, & x \in [0, 0.5] \\ 1-x, & x \in [0.5, 1.5] \\ x-2, & x \in [1.5, 2] \end{cases}$ Вывести анимированный график решения.

Листинги к теме 6. Колебания струны.

```
function z=Alpha(t,k)
    Nt=length(t); zz=zeros(1,Nt);
    z=zz(k);
    %omega=2;A=0.25;
    %z=A*sin(omega*t(k));
    %%%%%%%%%%
function z=Beta(t,k)
    Nt=length(t);
    zz=zeros(1,Nt);
    z=zz(k);
    omega=2;A=0.5;
    z=-A*sin(omega*t(k));
```

```

    if t(k)<=4*pi
        z=A*sin(omega*t(k));
    else
        z=0;
    end
    %%%%%%%%%%
function y=f(x,t,i,k)
y=0;
    %%%%%%%%%%
function z=FF(x,t,u,i,k,h,tau,Sigma)
gamma=tau/h;
    z=u(i,k-1)-2*u(i,k)-...
        gamma^2*(1-2*Sigma)*LAMBDAx(u,i,k)-...
        Sigma*gamma^2*LAMBDAx(u,i,k-1)-...
        tau^2*f(x,t,i,k+1);
    %%%%%%%%%%
function z=LAMBDAx(u,i,k)
z=(u(i-1,k)-2*u(i,k)+u(i+1,k));
    %%%%%%%%%%5
%stringExplicit.m - явная схема
% tau/h^2<0.5
clear all;close all;
x0=0;x1=pi;h=pi/12;x=x0:h:x1;Nx=length(x);
t0=0;t1=4*pi;tau=0.04;
t=t0:tau:t1;Nt=length(t);
f=zeros(Nx,Nt);
u=zeros(Nx,Nt);% будущее решение
Omeg=1;
%заполнение f и точного решения
for i=1:Nx
    for k=1:Nt
        f(i,k)=sin(x(i))*cos(Omeg*t(k));
    end%k
end%i

```

```

%Заносим 2 начальных условия по t
u(:,1)=sin(x(:));v(:,1)=0;
%Заносим граничные условия
u(1,:)=0;u(end,:)=0;
%явная схема i -> x k -> t
%зная производную по t на 1-слое из н.у
%находим "u" на 2-м
%используя левую конечную разность
u(:,2)=u(:,1)+tau*v(:,1);
for k=2:Nt
    for i=2:Nx-1
        u(i,k+1)=2*u(i,k)-u(i,k-1)+...
            (tau/h)^2*...
            (u(i-1,k)-2*u(i,k)+u(i+1,k))...
            +tau^2*f(i,k);
    end%i
end%k
htt=round(length(t)/12);
tt=1:htt:Nt;
figure('name',' Резонанс');
for j=1:length(tt)
    %figure('name',['t= ',num2str(t(tt(j)))]);
    plot(x,u(1:Nx,tt(j)));hold on;
end%j
grid on;title('\omega=1');
xlabel('x');ylabel('u(x,t)');
%%%%%%%%%%%%
% RunningWave.m Бегущая волна - неявная
%Внимание: здесь m-функция LAMBDAx(u,i,k)
% в FF(x,t,u,i,k,h,tau,Sigma)
%не имеет h в знаменателе *****
clear all;close all;
x0 = 0; x1=2*pi; h=pi/50;
x=x0:h:x1;Nx=length(x);

```

```

t0=0;t1=8*pi;tau=h/2;
t=t0:tau:t1;Nt=length(t);
gamma=tau/h;Sigma=1/3;
nn=2;Omega=2;
u=zeros(Nx,Nt);% будущее решение
v=zeros(Nx,1);%Начальная скорость струны
d2fi=zeros(Nx,1);y=zeros(Nx,1);
b=zeros(Nx,1);
u(1:Nx,1)=zeros(Nx,1);%sin(nn*x);
% НАЧАЛЬНОЕ РАСПРЕДЕЛЕНИЕ u(x,0)=SIN(nn*x)
d2fi(1:Nx,1)=zeros(Nx,1);
%-nn^2*sin(nn*x);%u''(x,0)=-sin(x)
%зная производную по t на 1-слое из н.у
%находим "u" на 2-м,
%используя разложение в ряд Тейлора
% до tau^2 включительно см. (**) стр.47
u(1:Nx,2)=u(1:Nx,1)+tau*v(1:Nx,1)+...
tau^2/2*d2fi(1:Nx,1);
%Заносим однородные граничные условия
for k=1:Nt
    u(1,k)=Alpha(t,k);
    u(end,k)=Beta(t,k);%sin(4*t)
end
%явная схема по x: (i -> x k -> t)
%матрица A 3-х диаг. для внутр.точек по x:
V=zeros(Nx-1,1);
V(:,1)=Sigma*gamma^2;
W=zeros(Nx,1);
W(:,1)=-(1+2*Sigma*gamma^2);
A=diag(W,0)+diag(V,1)+diag(V,-1);
A(1,1)=1;A(1,2)=0;
A(end,end)=1;A(end,end-1)=0;
figure('name','Численное
решение','color','w');%2

```

```

for k=2:Nt-1
    for i=2:Nx-1
        b(i)=FF(x,t,u,i,k,h,tau,Sigma);
    end%i (координата)
    b(1)=Alpha(t,k+1);b(end)=Beta(t,k+1);
    y=A\b;
    % Вычислено решение на k+1 слое (по t),
    % заполняем k+1 -й столбец u
    u(:,k+1)=y;
    %if mod(k,5)==0%вывод через 5
        hpu=plot(x,u(1:Nx,k+1),'r-
', 'LineWidth',3);
        xlim([x0,x1]);ylim([-1.5,1.5]);
title(...
'Численное решение \beta(t)=-0.5*\sin(\omega*t)');
    xlabel(['x' ' t=' num2str(t(k))],...
'fontname','Times New Roman',...
'fontsize',14,'fontweight','b');
ylabel(...
'u(x,t)', 'fontname','Times New Roman',...
'fontsize',14,'fontweight','b');
    grid on;
    drawnow;
    %end
end%k
% figure('name',...
'Колебания струны','color','w');%1
plot(x,u(1:Nx,1:25:Nt),'LineWidth',1);
xlim([0 x1]);
% xlabel('x','fontname','Times New Roman',...
% 'fontsize',14,'fontweight','b');
% ylabel('Ut(x,t)', 'fontname',...
%'Times New Roman',...
% 'fontsize',14,'fontweight','b');

```

```

% title(...)
%'Неявная схема.  $0 \leq x \leq \pi$ ;
% Нач.распр. $\sin(nx)$  ',...
%      'fontname','Times New Roman',...
%      'fontsize',14,'fontweight','b');
% grid on;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%stringExplicit.m - явная схема
clear all;close all;
x0=0;x1=pi;h=pi/12;x=x0:h:x1;%x-строка
Nx=length(x);t0=0;t1=2*pi;
tau=0.45*h^2;%tau/h^2<0.5
t=t0:tau:t1;Nt=length(t);
n=2;%See n at (6.4.2),(6.4.4)
u=zeros(Nx,Nt);% будущее решение
Ut=u;f=u;Omeg=1;% Ut=sin(n*x).*cos(Omeg*t)
%заполнение f и точного решения
for i=1:Nx
    for k=1:Nt
        Ut(i,k)=sin(n*x(i)).*cos(Omeg*t(k));
        %f(i,k)=(n^2-
Omeg^2)*sin(n*x(i)).*cos(Omeg*t(k));
    end%k
end%i
%Прореживаем график по t.
%Хотелось бы равномерно разместить кривые:
figure('color','w');%1
plot(x,Ut(:,1:15:end),'-*');
grid on;ylim([-1 1]);
title(...)
['Точное решение'\omega=' num2str(Omeg)...
'; n=' num2str(n) ';' ]);
xlabel('x');ylabel('Ut(x,t)');
%Заносим 2 начальных условия по t

```

```

u(:,1)=sin(n*x(:));v(:,1)=0;
%Заносим граничные условия
u(1,:)=0;u(end,:)=0;
%явная схема i ->x k ->t
%зная производную по t на 1-слое из н.у
%находим "u" на 2-м
%используя левую конечную разность
u(:,2)=u(:,1)+tau*v(:,1);
for k=2:Nt-1
    for i=2:Nx-1
        u(i,k+1)=2*u(i,k)-u(i,k-1)+...
            (tau/h)^2*...
            (u(i-1,k)-2*u(i,k)+u(i+1,k))...
            +tau^2*f(i,k);
    end%i
end%k
figure('color','w');
plot(x,u(:,1:15:end),'-*');
grid on;ylim([-1 1]); %2
title(...
['Численное решение;' '\omega=' num2str(Omeg) ...
    ';' n=' num2str(n) ';' ]);
xlabel('x');ylabel('u(x,t)');
%Deviation
Dev=abs(Ut-u);
figure('color','w');
plot(x,Dev(:,1:15:end),'-*');
grid on;xlim([0 pi]); %3
title(['Ошибка при t_e_n_d;',...
    ' \omega=' num2str(Omeg) ';' n=' ,...
    num2str(n) ';' hx=' num2str(h) ';' ]);
xlabel('x');ylabel('| Ut-u |');
figure('color','w');%4
plot(t,Dev(round(length(x)/4),:),'g',...

```

```

'linewidth',2);hold on;
plot(t,Dev(round(3*length(x)/4),:),'b',...
'linewidth',2);grid on;
title(['Error=f(x,t) ' '\omega='...
      num2str(Omeg) ' ; n=',...
num2str(n) ' ; hx=' num2str(h) ' ;']);
xlabel('t');ylabel('| Ut-u |');
%В центре струны-неподвижная точка
figure('color','w');
plot(t,Dev(round(length(x)/2),:),...
'r-+', 'linewidth',2);
grid on;xlim([0 t1]);
title(['Ошибка в центре струны;...
      ' '\omega='...
      num2str(Omeg) ' ; n=' , num2str(n) ' ; hx='
num2str(h) ' ; tau=;', num2str(tau)]);
xlabel('t');ylabel('| Ut-u |');
%%%%%%%%%%%%%
%StringExplicitForcedOscillation.m
%- явная схема tau/h^2<0.5
% Точное решение
clear all;close all;
x0=0;
x1=5;
%x1=2*pi;
%h=pi/24;
h=0.05;
x=x0:h:x1;Nx=length(x);
t0=0;
%t1=3*pi;
t1=8*pi;
%tau=pi/6;
tau=0.45*h^2;
t=t0:tau:t1;Nt=length(t);

```



```

nn=1;%See n at equation (1)
u=zeros(Nx,Nt);% будущее решение
Ut=u;f=u;Dev=u;
Omeg=1;%(n^2-Omeg^2)=1-4=-3;
Ut=sin(x).*cos(2*t)
%Заполнение f; точного решения не знаю
for i=1:Nx
    for k=1:Nt
        %Ut(i,k)=sin(x(i)).*cos(Omeg*t(k));
        f(i,k)=(nn^2-
Omeg^2)*sin(nn*x(i))*cos(Omeg*t(k));
    end%k
end%i
%Прореживаем график по t.
%Хотелось бы равномерно разместить кривые:
%figure('color','w');%1
%plot(x,Ut(:,1:5:end),'k-*');grid on;ylim([-1
1]);
%title(['Точное решение; ' '\omega='
num2str(Omeg)...
% ' ; n=' num2str(nn) ' ;']);
xlabel('x');ylabel('Ut(x,t)');
%Заносим 2 начальных условия по t
%u(:,1)=0;
% [y]=BegNormDistr(Sigma,Mju,x)
Sigma=0.15;Mu=2.5;
u(:,1)=BegNormDistr(Sigma,Mu,x);
uMax=max(max(u));
v(:,1)=0;
%Заносим граничные условия
u(1,:)=0;
%u(end,:)=sin(Omeg*t);
u(end,:)=0;
%явная схема i ->x k ->t

```

```

%зная производную по t на 1-слое из n.y
%находим "u" на 2-м
%используя левую конечную разность
u(:,2)=u(:,1)+tau*v(:,1);
figure('color','w');%1
for k=2:Nt-1
    for i=2:Nx-1
        u(i,k+1)=2*u(i,k)-u(i,k-1)+...
            (tau/h)^2*...
            (u(i-1,k)-2*u(i,k)+u(i+1,k))+tau^2*f(i,k);
    end%i
    plot(x,u(:,k),'k-','LineWidth',3);
    grid on;ylim([-uMax uMax]);
    tStr=num2str(t(k-1));
%только для вывода нулевого момента на plot
    tStrLen=length(tStr);
    tStr=tStr(1:tStrLen-3);
    title(...
['Численное Решение;' '\omega=' num2str(Omeg)...
    ';' n=' num2str(nn) ';' ]);
    xlabel(['x; t=' tStr]);ylabel('u(x,t)');
    pause;
end%k
%Deviation
Dev=abs(Ut-u);
figure('color','w');%3
plot(x,Dev(:,Nt));grid on;xlim([0 pi]);
title(['Ошибка при t_e_n_d; ' ...
'\omega=' num2str(Omeg) '];
hx=' num2str(h) ';' ]);
xlabel('x');ylabel('| Ut-u |');
% Сделать зависимость Dev от времени
figure('color','w');%4
yy=zeros(3,length(t));

```

```

xx=[round(length(x)/4) round(length(x)/2) ...
round(3*length(x)/4)];
yy(:, :)=Dev(xx(:, :));
plot(t,yy(1,:), 'r');hold on;ylim([0 0.065]);
plot(t,yy(2,:), 'g');hold on;ylim([0 0.065]);
plot(t,yy(3,:), 'b');grid on;ylim([0 0.065]);
title(['Ошибка Error=f(x,t) ' '\omega=' ...
num2str(Omeg) '; hx=' num2str(h) ' ;']);
xlabel('t');ylabel('| Ut-u |');
%legend(['x=' , num2str(xx(1)) 'x='
num2str(xx(2)) , ...
% 'x=' num2str(xx(3))]);
%%%%%%%%%%%%
% StringImplAccur.m Свободные колебания
% (Пример 1) - неявная
% Внимание: здесь m-функция LAMBDAx(u,i,k)
% в FF(x,t,u,i,k,h,tau,Sigma)
% не имеет h в знаменателе *****
clear all;close all;
tic;
x0 = 0; x1=pi;%/2;
H=pi/4*logspace(0,-2,3);
Err=zeros(1,length(H));
for Hk=1:length(H)
h=H(Hk);x=x0:h:x1;Nx=length(x);
t0=0;t1=pi/2;%2;%4;
tau=h;t=t0:tau:t1;Nt=length(t);
gamma=tau/h;Sigma=1/3;
nn=1;Omega=1;
% свободные колебания f(x,t)=0;
% See n at equation (6.3)
Ut=ut(x,t,nn,Omega);
% Есть точное решение, используем.
u=zeros(Nx,Nt);% будущее решение

```

```

v=zeros(Nx,1);%Начальная скорость струны
d2fi=zeros(Nx,1);y=zeros(Nx,1);
b=zeros(Nx,1);
u(1:Nx,1)=sin(nn*x);
% НАЧАЛЬНОЕ РАСПРЕДЕЛЕНИЕ u(x,0)=SIN(nn*x)
d2fi(1:Nx,1)=-nn^2*sin(nn*x);%u''(x,0)=-sin(x)
%зная производную по t на 1-слое из н.у
%находим "u" на 2-м,
%используя разложение в ряд Тейлора
%до tau^2 включительно см. (**) стр.47
u(1:Nx,2)=u(1:Nx,1)+tau*v(1:Nx,1)+tau^2/2*d2fi
(1:Nx,1);
%Заносим однородные граничные условия
for k=1:Nt
    u(1,k)=Alpha(t,k);u(end,k)=Beta(t,k);
end
%неявная схема по x: (i -> x k -> t)
%матрица A 3-х диаг. для внутр.точек по x:
V=zeros(Nx-1,1);V(:,1)=Sigma*gamma^2;
W=zeros(Nx,1);W(:,1)=-(1+2*Sigma*gamma^2);
A=diag(W,0)+diag(V,1)+diag(V,-1);
A(1,1)=1;A(1,2)=0;A(end,end)=1;A(end,end-1)=0;
for k=2:Nt-1
    for i=2:Nx-1
        b(i)=FF(x,t,u,i,k,h,tau,Sigma);
    end%i (координата)
    b(1)=Alpha(t,k+1);b(end)=Beta(t,k+1);
    y=A\b;
    % Вычислено решение на k+1 слое (по t),
    % заполняем k+1 -й столбец u
    u(:,k+1)=y;
end%k
Err(Hk)=max(max(Ut-u));
end%Hk

```

```

figure('name','Ошибка неявной схемы
h0=pi/4',...
'color','w');%1
loglog(H,Err);
xlabel('h','fontname','Times New Roman',...
'fontsize',16,'fontweight','b');
ylabel('|Ut(x,t)-u(x,t)|','fontname',...
'Times New Roman',...
'fontsize',16,'fontweight','b');
title('Ошибка неявной схемы',...
'fontname','Times New Roman',...
'fontsize',16,'fontweight','b');
grid on;
toc;
%%%%%%%%%
% StringImplicit.m Свободные колебания
% (Пример 1) - неявная
%Внимание: здесь m-функция LAMBDAx(u,i,k) в
% FF(x,t,u,i,k,h,tau,Sigma)
%не имеет h в знаменателе *****
clear all;close all;
x0 = 0; x1=pi; h=pi/12;x=x0:h:x1;Nx=length(x);
t0=0;t1=pi/2;tau=h;t=t0:tau:t1;Nt=length(t);
gamma=tau/h;Sigma=1/3;
nn=2;Omega=2;
% свободные колебания f(x,t)=0;
% See n at equation (6.3)
Ut=ut(x,t,nn,Omega);
% Есть точное решение, используем.
%%%%%%%%% ВЫВОД ТОЧНОГО РЕШЕНИЯ
figure('name',...
'Колебания струны','color','w');%1
plot(x,Ut(1:Nx,1:Nt),'LineWidth',1);

```

```

xlim([x0,x1]);ylim([-1,1]);
xlabel('x','fontname','Times New Roman',...
        'fontsize',14,'fontweight','b');
ylabel('Ut(x,t)','fontname',...
        'Times New Roman',...
        'fontsize',14,'fontweight','b');
title(...
'Tочное решение.  $0 \leq x \leq \pi$ ; Нач.распр.sin(nx)',...
        'fontname','Times New Roman',...
        'fontsize',14,'fontweight','b');

grid on;
%%%%%%%%%%
u=zeros(Nx,Nt);% будущее решение
v=zeros(Nx,1);%Начальная скорость струны
d2fi=zeros(Nx,1);y=zeros(Nx,1);b=zeros(Nx,1);
u(1:Nx,1)=sin(nn*x);
% НАЧАЛЬНОЕ РАСПРЕДЕЛЕНИЕ u(x,0)=SIN(nn*x)
d2fi(1:Nx,1)=-nn^2*sin(nn*x);%u'(x,0)=-sin(x)
%зная производную по t на 1-слое из n.y
%находим "u" на 2-м,
%используя разложение в ряд Тейлора до tau^2
% включительно см. (**) стр.47
u(1:Nx,2)=u(1:Nx,1)+tau*v(1:Nx,1)+...
tau^2/2*d2fi(1:Nx,1);
%Заносим однородные граничные условия
for k=1:Nt
    u(1,k)=Alpha(t,k);u(end,k)=Beta(t,k);
end
%неявная схема по x: (i -> x k -> t)
%матрица A 3-х диаг. для внутр.точек по x:
V=zeros(Nx-1,1);V(:,1)=Sigma*gamma^2;
W=zeros(Nx,1);W(:,1)=-(1+2*Sigma*gamma^2);
A=diag(W,0)+diag(V,1)+diag(V,-1);
A(1,1)=1;A(1,2)=0;A(end,end)=1;A(end,end-1)=0;

```

```

% figure('name',...
'Численное решение','color','w');%2
% ВЫВОД АНИМАЦИИ

for k=2:Nt-1
    for i=2:Nx-1
        b(i)=FF(x,t,u,i,k,h,tau,Sigma);
    end%i (координата)
    b(1)=Alpha(t,k+1);b(end)=Beta(t,k+1);
    y=A\b;
    % Вычислено решение на k+1 слое (по t),
заполняем k+1 -й столбец u
    u(:,k+1)=y;
    %%%%%%%%% ВЫВОД АНИМАЦИИ %%%%%%%%%
%hpu=plot(x,u(1:Nx,k+1),'k-','LineWidth',1);
%         xlim([x0,x1]);ylim([-1,1]);grid on
%         title('Численное решение');
%xlabel('x','fontname','Times New Roman',...
%         'fontsize',14,'fontweight','b');
%ylabel('u(x,t)','fontname',...
%'Times New Roman',...
%         'fontsize',14,'fontweight','b');
%         drawnow;
    %%%%%%%%% КОНЕЦ ВЫВОДА АНИМАЦИИ
%%%%%%%%
end%k
figure('name','Колебания
струны','color','w');%1
plot(x,u(1:Nx,1:Nt),'LineWidth',1);
xlim([x0,x1]);ylim([-1,1]);
xlabel('x','fontname','Times New Roman',...
        'fontsize',14,'fontweight','b');
ylabel('Ut(x,t)','fontname','Times New Roman',...
        'fontsize',14,'fontweight','b');

```

```

title('Неявная схема.  $0 \leq x \leq \pi$ ;  

      Нач. расп.  $\sin(nx)$  ', ...  

      'fontname', 'Times New Roman', ...  

      'fontsize', 14, 'fontweight', 'b');  

grid on;  

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%5  

%Задача Дирихле с одног ГУ  

%ErrStringShort   явная схема  $\tau/h < 1/2$   

tic;  

clear all; close all; x0=0; x1=pi; t0=0; t1=pi/6;  

H=pi/6*logspace(0,-  

2,3); Err=zeros(1,length(H));  

K=1  

while K<=length(H)  

h=H(1,K); x=x0:h:x1; Nx=length(x);  

tau=0.3*h^2; t=t0:tau:t1; Nt=length(t);  

Ut=zeros(1,Nx); uNew=Ut; v=Ut;  

Omeg=1; n=1;  

%Free oscillation !!!!!!!!!!!!!!!!!!!!!  

%Заносим 2 начальных условия по t  

uOld=sin(n*x); u=uOld+tau*v;  

%Заносим граничные условия  

uOld(1)=0; uOld(end)=0; u(1)=0; u(end)=0;  

uNew(1)=0; uNew(end)=0;  

k=2;  

while k<Nt  

for i=2:Nx-1  

    uNew(i)=...  

    2*u(i)-uOld(i)+(tau/h)^2*(u(i-1)-...  

    2*u(i)+u(i+1));  

    Ut(i)=sin(n*x(i)).*cos(Omeg*t(k+1));  

    %***** Погрешность в точке  

%*****  

    if abs(uNew(i)-Ut(i))>Err(K)

```



```

        Err(K)=abs(uNew(i)-Ut(i));
    end
end%i
uOld=u;u=uNew;
k=k+1;
end%while k
%disp(...)
['h=' num2str(h) ' Err=' num2str(Err(1))];
K=K+1
end%while K
% save 'H.dat' H -ascii;
% save 'Errors.dat' Err -ascii;

figure('name', 'Погрешность','color','w');
loglog(H,Err);grid on;
disp(['t1=' num2str(t1) ' h=' num2str(h) '
tau=' num2str(tau) ]);
toc;
%%%%%%%%%
%stringExplicit.m - явная схема tau/h^2<0.5
clear all;close all;
x0=0;x1=pi;h=pi/12;x=x0:h:x1;%x-строка
Nx=length(x);t0=0;t1=2*pi;
% Интервал по времени (0,2*pi)
tau=0.45*h^2;%Условия устойчивости выполнены
t=t0:tau:t1;Nt=length(t);
n=1;%См. n в (6.4.2, 6.4.4)
u=zeros(Nx,Nt);% Описание будущего решения
Ut=u;f=u;
%правая часть f=0 (свободные колебания)
Dev=u;% Описание будущей погрешности
Оmeg=1;
% См. Омега в ( 6.4.4); Ut=sin(x).*cos(t)
%заполнение f и точного решения

```

```

for i=1:Nx
    for k=1:Nt
        Ut(i,k)=sin(n*x(i)).*cos(Omeg*t(k));
        %f(i,k)=(n^2-
Omeg^2)*sin(n*x(i))*cos(Omeg*t(k));
    end%k
end%i
figure('color','w');%1
plot(x,Ut(:,1:15:end),'-*');
%Прореживаем график по t.
%Хотелось бы равномерно разместить кривые:
grid on;ylim([-1 1]); xlim([0 x1]);
title(...
['Точное решение;' '\omega=' num2str(Omeg)...
'; n=' num2str(n) ';' ]);
xlabel('x');ylabel('Ut(x,t)');
%Заносим 2 начальных условия по t
u(:,1)=sin(n*x(:));v(:,1)=0;
%Заносим граничные условия
u(1,:)=0;u(end,:)=0;
%явная схема i ->x k ->t
%зная производную по t на 1-слое из н.у
%находим "u" на 2-м
%используя левую конечную разность
u(:,2)=u(:,1)+tau*v(:,1);
for k=2:Nt-1
    for i=2:Nx-1
        u(i,k+1)=2*u(i,k)-u(i,k-1)+...
            (tau/h)^2*(u(i-1,k)-
2*u(i,k)+u(i+1,k))+tau^2*f(i,k);
    end%i
end%k
figure('color','w');plot(x,u(:,1:15:end));
grid on;ylim([-1 1]); xlim([0 x1]);

```

```

title(...
['Численное решение;...
' '\omega=' num2str(Omeg) ...
'; n=' num2str(n) ';' ] );
xlabel('x'); ylabel('u(x,t)');
Dev=abs(Ut-u); %Deviation (Уклонение)
figure('color','w');
plot(x,Dev(:,1:15:end)); grid on; xlim([0 x1]);
title(...
['Ошибка при t_e_n_d; ' '\omega='
num2str(Omeg) ' ';
hx=' num2str(h) ';' ] );
xlabel('x'); ylabel('| Ut-u |');
% Сделать зависимость Dev от времени
figure('color','w');
yy=zeros(3,length(t));
xx=...
[round(length(x)/4) round(length(x)/2) round(3*length(x)/4)];
yy(:, :) = Dev(xx(:, :), :);
plot(t,yy(1,:), 'r'); hold on; xlim([0 t1]);
plot(t,yy(2,:), 'g'); hold
on; plot(t,yy(3,:), 'b'); grid on;
title(['Ошибка Error(x,t) ' '\omega=' ...
num2str(Omeg) ' '; hx=' num2str(h) ';' ] );
xlabel('t'); ylabel('| Ut-u |');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%stringExplicit.m - явная схема
clear all; close all;
x0=0; x1=pi; h=pi/12; x=x0:h:x1; %x-строка
Nx=length(x); t0=0; t1=2*pi;
tau=0.45*h^2; %tau/h^2<0.5
t=t0:tau:t1; Nt=length(t);
n=2; %See n at (6.4.2), (6.4.4)
u=zeros(Nx,Nt); % будущее решение

```

```

Ut=u;f=u;Omeg=1;% Ut=sin(n*x).*cos(Omeg*t)
%заполнение f и точного решения
for i=1:Nx
    for k=1:Nt
        Ut(i,k)=sin(n*x(i)).*cos(Omeg*t(k));
        %f(i,k)=(n^2-
%Omeg^2)*sin(n*x(i))*cos(Omeg*t(k));
    end%k
end%i
%Прореживаем график по t.
%Хотелось бы равномерно разместить кривые:
figure('color','w');%1
plot(x,Ut(:,1:15:end),'-*');grid on;ylim([-1
1]);
title(...
['Точное решение; '...
'\omega=' num2str(Omeg)...
'; n=' num2str(n) ';' ]);
xlabel('x');ylabel('Ut(x,t)');
%Заносим 2 начальных условия по t
u(:,1)=sin(n*x(:));v(:,1)=0;
%Заносим граничные условия
u(1,:)=0;u(end,:)=0;
%явная схема i ->x k ->t
%зная производную по t на 1-слое из н.у
%находим "u" на 2-м
%используя левую конечную разность
u(:,2)=u(:,1)+tau*v(:,1);
for k=2:Nt-1
    for i=2:Nx-1
        u(i,k+1)=2*u(i,k)-u(i,k-1)+...
            (tau/h)^2*(u(i-1,k)-...
2*u(i,k)+u(i+1,k))+tau^2*f(i,k);
    end%i

```

```

end%k
figure('color','w');
plot(x,u(:,1:15:end),'-*');grid on;
ylim([-1 1]); %2
title(['Численное решение; ' ...
      '\omega=' num2str(Omeg) ...
      '; n=' num2str(n) ';' ]);
xlabel('x');ylabel('u(x,t)');
%Deviation
Dev=abs(Ut-u);
figure('color','w');
plot(x,Dev(:,1:15:end),'-*');
grid on;xlim([0 pi]); %3
title(...
['Ошибка при t_e_n_d; ' ...
  '\omega=' num2str(Omeg) ...
  '; n=',num2str(n) '; hx=' num2str(h) ';' ]);
xlabel('x');ylabel('| Ut-u |');
figure('color','w');%4
plot(t,Dev(round(length(x)/4),:),'g',...
'linewidth',2);hold on;
plot(t,Dev(round(3*length(x)/4),:),'b',...
'linewidth',2);grid on;
title(['Error=f(x,t) ' '\omega=' ...
      num2str(Omeg) '; n=',num2str(n) ...
      '; hx=' num2str(h) ';' ]);
xlabel('t');ylabel('| Ut-u |');
%В центре струны-неподвижная точка
figure('color','w');
plot(t,Dev(round(length(x)/2),:),'r-+',...
'linewidth',2);
grid on;xlim([0 t1]);
title(...
['Ошибка в центре струны; ' '\omega=' ...

```

```

        num2str(Omeg) ' ; n=' , ...
num2str(n) ' ; hx=' num2str(h) ...
' ; tau=;' , num2str(tau) ] ) ;
xlabel('t') ; ylabel(' | Ut-u | ' ) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%stringExplicit.m - явная схема
clear all ; close all ;
x0=0 ; x1=pi ; h=pi/12 ; x=x0:h:x1 ; %x-строка
Nx=length(x) ; t0=0 ; t1=2*pi ;
tau=0.45*h^2 ; %tau/h^2<0.5
t=t0:tau:t1 ; Nt=length(t) ;
n=2 ; %See n at (6.4.2) , (6.4.4)
u=zeros(Nx,Nt) ; % будущее решение
Ut=u ; f=u ; Omeg=1 ; % Ut=sin(n*x) .* cos(Omeg*t)
%заполнение f и точного решения
for i=1:Nx
    for k=1:Nt
        Ut(i,k)=sin(n*x(i)) .* cos(Omeg*t(k)) ;
        %f(i,k)=(n^2-
%Omeg^2)*sin(n*x(i)) *cos(Omeg*t(k)) ;
    end%k
end%i
%Прореживаем график по t.
%Хотелось бы равномерно разместить кривые:
figure('color','w') ; %1
plot(x,Ut(:,1:15:end),'-*') ;
grid on ; ylim([-1 1]) ;
title(['Точное решение ; ' '\omega='
num2str(Omeg) ...
' ; n=' num2str(n) ' ; ']) ;
xlabel('x') ; ylabel('Ut(x,t)') ;
%Заносим 2 начальных условия по t
u(:,1)=sin(n*x(:)) ; v(:,1)=0 ;
%Заносим граничные условия

```

```

u(1,:)=0;u(end,:)=0;
%явная схема i -> x k -> t
%зная производную по t на 1-слое из n.y
%находим "u" на 2-м
%используя левую конечную разность
u(:,2)=u(:,1)+tau*v(:,1);
for k=2:Nt-1
    for i=2:Nx-1
        u(i,k+1)=2*u(i,k)-u(i,k-1)+...
            (tau/h)^2*(u(i-1,k)-
2*u(i,k)+u(i+1,k))+tau^2*f(i,k);
    end%i
end%k
figure('color','w');plot(x,u(:,1:15:end),'-
*');grid on;ylim([-1 1]); %2
title(...)
['Численное решение; ' ...
'\omega=' num2str(Omeg)...
'; n=' num2str(n) ';' ]);
xlabel('x');ylabel('u(x,t)');
%Deviation
Dev=abs(Ut-u);
figure('color','w');
plot(x,Dev(:,1:15:end),'-*');grid on;xlim([0
pi]); %3
title(['Ошибка при t_e_n_d; ' '\omega='
num2str(Omeg) ' ; n=',num2str(n)...
'; hx=' num2str(h) ';' ]);
xlabel('x');ylabel('| Ut-u |');
figure('color','w');%4
plot(t,Dev(round(length(x)/4),:),'g',...
'linewidth',2);hold on;
plot(t,Dev(round(3*length(x)/4),:),'b',...
'linewidth',2);grid on;

```

```

title(['Error=f(x,t) ' '\omega='...
      num2str(Omeg) ' ; n=',num2str(n)...
      ' ; hx=' num2str(h) ' ;']);
xlabel('t');ylabel('| Ut-u |');
%В центре струны-неподвижная точка
figure('color','w');
plot(t,Dev(round(length(x)/2),:),...
'r-+', 'linewidth',2);
grid on;xlim([0 t1]);
title(['Ошибка в центре струны;...
      ' '\omega='...
      num2str(Omeg) ' ; n=',num2str(n)...
      ' ; hx=' num2str(h) ' ; tau=;',num2str(tau)]);
xlabel('t');ylabel('| Ut-u |');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%stringExplicitNonStability.m - явная схема
clear all;close all;
x0=0;x1=pi;h=pi/12;x=x0:h:x1;%x-строка
Nx=length(x);t0=0;t1=3*pi/2;
tau=pi/3;%tau=0.45*h^2;%tau/h^2<0.5
disp(tau/h^2);
t=t0:tau:t1;Nt=length(t);
n=1;%See n at (6.4.2), (6.4.4)
u=zeros(Nx,Nt);% будущее решение
Ut=u;f=u;Omeg=2;% Ut=sin(n*x).*cos(Omeg*t)
%заполнение f и точного решения
for i=1:Nx
    for k=1:Nt
        Ut(i,k)=sin(n*x(i)).*cos(Omeg*t(k));
        %f(i,k)=(n^2-
Omeg^2)*sin(n*x(i)).*cos(Omeg*t(k));
    end%k
end%i
%Прореживаем график по t.

```



```

%Хотелось бы равномерно разместить кривые:
figure('color','w');%1
plot(x,Ut(:,1:2:end),'-*');grid on;
ylim([-1 1]);
title(...)
['Точное решение; ' ...
'\omega=' num2str(Omeg)...
'; n=' num2str(n) ';'']);
xlabel('x');ylabel('Ut(x,t)');
%Заносим 2 начальных условия по t
u(:,1)=sin(n*x(:));v(:,1)=0;
%Заносим граничные условия
u(1,:)=0;u(end,:)=0;
%явная схема i ->x k ->t
%зная производную по t на 1-слое из н.у
%находим "u" на 2-м
%используя левую конечную разность
u(:,2)=u(:,1)+tau*v(:,1);
for k=2:Nt-1
    for i=2:Nx-1
        u(i,k+1)=2*u(i,k)-u(i,k-1)+...
            (tau/h)^2*(u(i-1,k)-...
2*u(i,k)+u(i+1,k))+tau^2*f(i,k);
    end%i
end%k
figure('color','w');
plot(x,u(:,1:2:end),'-*');grid on;
%ylim([-1 1]); %2
title(...)
['Численное решение; ' ...
'\omega=' num2str(Omeg)...
'; n=' num2str(n) ';'']);
xlabel('x');ylabel('u(x,t)');
%Deviation

```

```

Dev=abs(Ut-u);
figure('color','w');
plot(x,Dev(:,1:2:end),'-*');grid on;
xlim([0 pi]); %3
title(['Ошибка при t_e_n_d; ' '\omega='...
      num2str(Omeg) ' ; n=',num2str(n) ' ;...
      hx=' num2str(h) ' ;']);
xlabel('x');ylabel('| Ut-u |');
figure('color','w');%4
plot(t,Dev(round(length(x)/4),:),'g',...
      'linewidth',2);hold on;
plot(t,Dev(round(3*length(x)/4),:),'b',...
      'linewidth',2);grid on;
title(['Error=f(x,t) ' '\omega='...
      num2str(Omeg) ' ; n=',num2str(n) ' ;...
      hx=' num2str(h) ' ;']);
xlabel('t');ylabel('| Ut-u |');
%В центре струны-неподвижная точка
figure('color','w');
plot(t,Dev(round(length(x)/2),:),...
      'r-+', 'linewidth',2);
grid on;xlim([0 t1]);
title(['Ошибка в центре струны;...
      ' '\omega='...
      num2str(Omeg) ' ; n=',num2str(n) ' ; hx='
num2str(h) ' ; tau=;',num2str(tau)]);
xlabel('t');ylabel('| Ut-u |');
%%%%%%%%%%%%

```

§7. Задача Дирихле для уравнения Пуассона

Найти функцию $u = u(x, y)$, удовлетворяющую в области D уравнению

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y) \quad (7.1)$$

и принимающую значения $g(s)$ на границе Γ области D .

Подобная модель может применяться для описания установившегося течения жидкости, стационарных тепловых полей, процессов теплопередачи с внутренними источниками тепла, деформации упругих мембран и т.д.

Рассмотрим численное решение этой задачи в прямоугольной области $D = (a, b) \times (c, d)$. Введем в D ортогональную сетку D_h .

$$\begin{cases} D_h = \{(x_i, y_k)\}; & h_x = (b - a)/(N_x - 1); h_y = (d - c)/(N_y - 1); \\ x_i = a + (i - 1)h_x; y_k = c + (k - 1)h_y; & 1 \leq i \leq N_x; 1 \leq k \leq N_y, \end{cases}$$

N_x и N_y — число узлов по каждой координате. Пусть $h_x = h_y = h$.

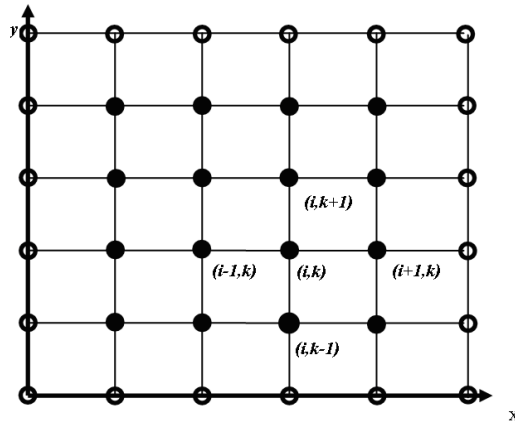


Рисунок 7.1. Ортогональная сетка

Аппроксимируем уравнение (7.1) во внутренних точках, используя пятиточечный шаблон (см. рис. 7.1):

$$\frac{u_{i-1,k} - 2u_{i,k} + u_{i+1,k}}{h^2} + \frac{u_{i,k-1} - 2u_{i,k} + u_{i,k+1}}{h^2} = f_{i,k}, \quad i = \overline{2, N_x - 1}, \quad k = \overline{2, N_y - 1},$$

где $f_{ik} = f(x_i, y_k)$. Приводя подобные слагаемые, приходим к системе линейных алгебраических уравнений относительно неизвестных $u_{i,k}$:

$$u_{i-1,k} + u_{i+1,k} - 4u_{i,k} + u_{i,k-1} + u_{i,k+1} = h^2 f_{i,k}.$$

Положим $N_x = N_y = 6$ и введем одномерную нумерацию неизвестных $u_{i,k}$ (см. рис. 7.2). Получим матрицу и вектор правой части этой системы.

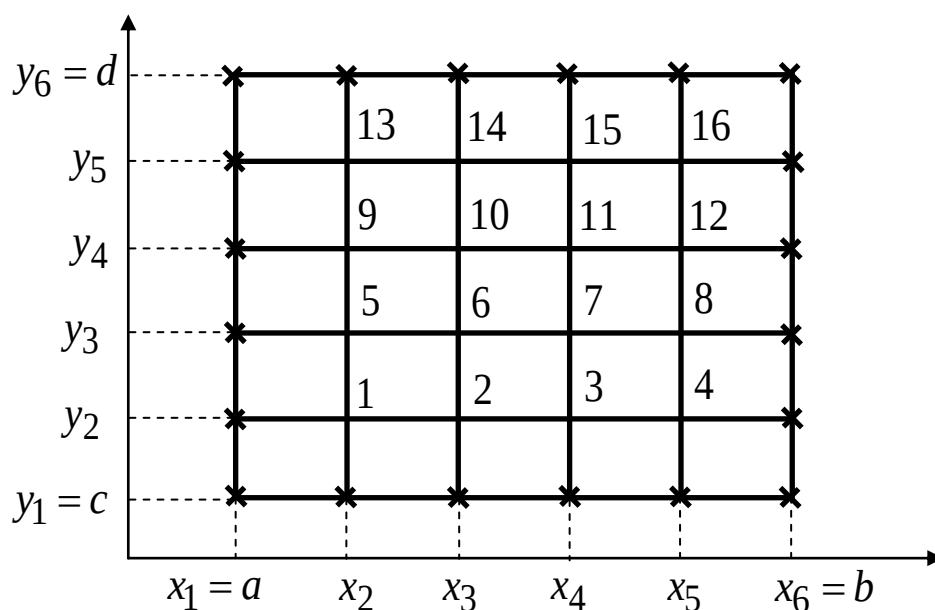


Рисунок 7.2. Одномерная нумерация узлов

Запишем систему для одномерной нумерации неизвестных, придерживаясь пятиточечного шаблона и учитывая граничные условия. В граничных узлах, помеченных на рис. 7.2 знаком « $\times$ », заданы граничные условия $g_{i,k}$, которые переносятся в правую часть системы. Получим (нулевые элементы матрицы не приводятся):

$$A = \begin{bmatrix} \begin{array}{cccc|cccc} -4 & 1 & & & 1 & & & \\ 1 & -4 & 1 & & & 1 & & \\ & 1 & -4 & 1 & & & 1 & \\ & & 1 & -4 & & & & 1 \\ & & & 1 & -4 & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} \\ \hline \begin{array}{cccc|cccc} 1 & & & & -4 & 1 & & 1 \\ & 1 & & & 1 & -4 & 1 & \\ & & 1 & & & 1 & -4 & 1 \\ & & & 1 & & 1 & -4 & \\ & & & & 1 & -4 & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} \\ \hline \begin{array}{cccc|cccc} & & & & 1 & & & \\ & & & & & 1 & & \\ & & & & & & 1 & \\ & & & & & & & 1 \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} \\ \hline \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} & \begin{array}{cccc|cccc} & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{array} \end{bmatrix}$$

Рисунок 7.3. Значения элементов $A_{(n-2) \times (n-2)}$

Теперь решаем системы $Au = F$, где $f_{i,k}$ перенумерована, как и $u_{i,k}$.

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \\ u_{10} \\ u_{11} \\ u_{12} \\ u_{13} \\ u_{14} \\ u_{15} \\ u_{16} \end{bmatrix}; \quad F = \begin{bmatrix} h^2 f_1 - g_{1,2} - g_{2,2} \\ h^2 f_2 - g_{3,2} \\ h^2 f_3 - g_{4,1} \\ h^2 f_4 - g_{6,2} - g_{5,1} \\ h^2 f_5 - g_{1,3} \\ h^2 f_8 \\ h^2 f_7 \\ h^2 f_8 - g_{6,3} \\ h^2 f_9 - g_{1,4} \\ h^2 f_{10} \\ h^2 f_{11} \\ h^2 f_{12} - g_{6,4} \\ h^2 f_{13} - g_{1,5} - g_{2,6} \\ h^2 f_{14} - g_{3,6} \\ h^2 f_{15} - g_{4,6} \\ h^2 f_{16} - g_{6,5} - g_{5,6} \end{bmatrix};$$

Рисунок 7.4. Значения элементов $u_{(n-2) \times 1}$ и $F_{(n-2) \times 1}$

Пример 1. Решить задачу (листинг PoissonTask.m)

$$\begin{cases} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -10\pi^2 \cdot \sin(\pi \cdot x) \cdot \sin(3\pi \cdot y); \\ (x, y) \in D = (0, 1) \times (0, 1); \quad u|_{\Gamma} = 0; \end{cases} \quad (7.3)$$

Точное решение $u_T = \sin(\pi \cdot x) \cdot \sin(3\pi \cdot y)$;

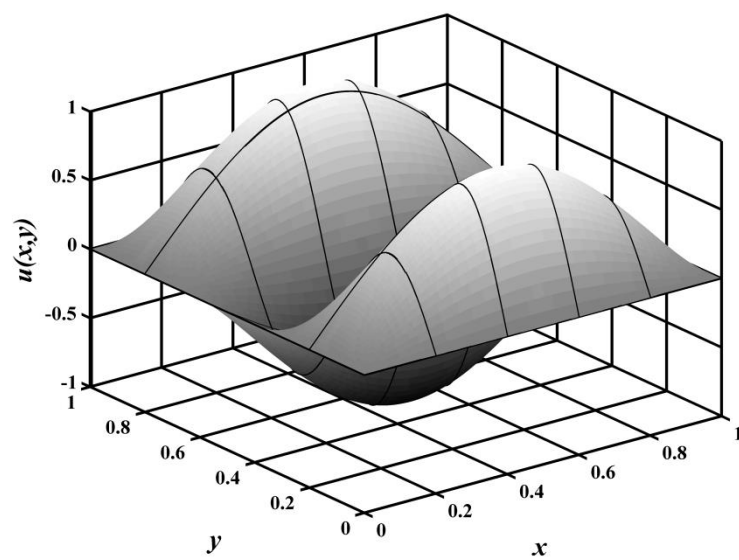


Рисунок 7.5. Точное решение на сетке 6×6

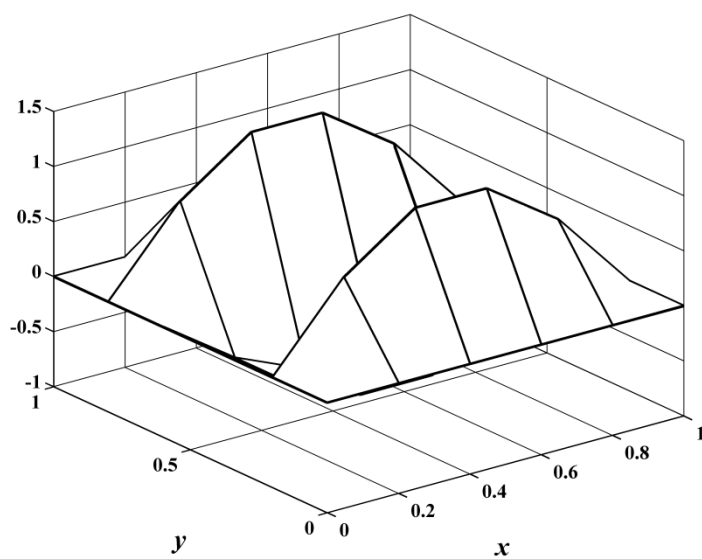


Рисунок 7.6. Численное решение на сетке 6×6

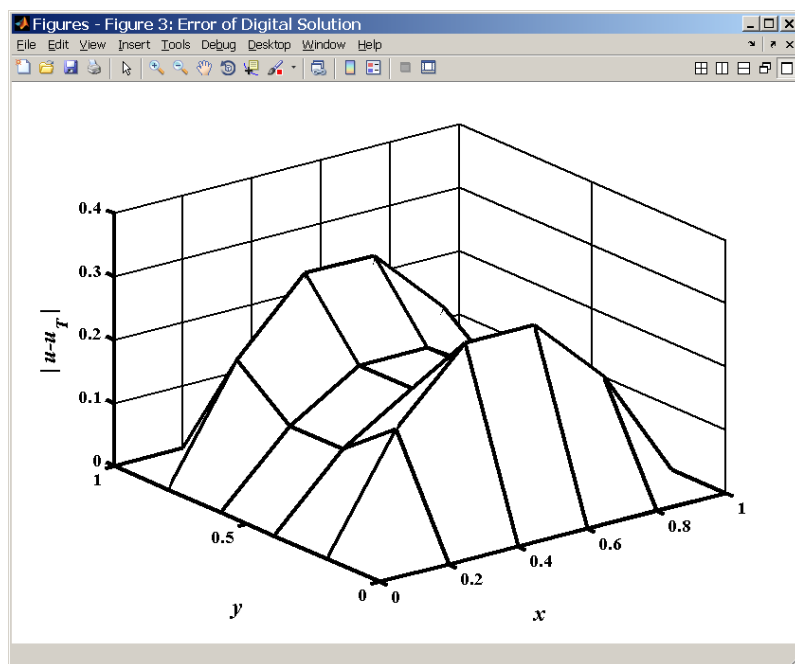


Рисунок 7.7. Погрешность численного решения на сетке 6×6
 Погрешность метода конечных разностей в рассматриваемой задаче пропорциональна h^2 .

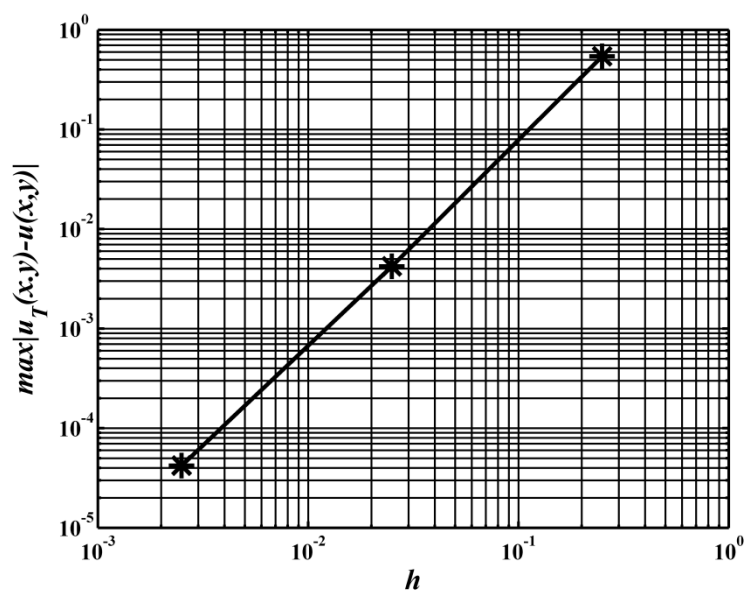


Рисунок 7.8. Ошибка, как функция h



Задачи

Решите задачу Дирихле для уравнения Пуассона с нулевыми граничными условиями.

№	Уравнение
1	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = -10$
2	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = 2(\cos(x + y) - y \cdot \sin(x + y))$
3	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = e^{-x} + e^{-y}$
4	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = x^2 + 3y$
5	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = -(x^2 + y^2)$
6	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = -100(x^2 + y^2) \sin xy$
7	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = -(n^2 + m^2) \pi^2 \sin n \pi x \cos m \pi y$
8	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = -10(x + y) e^{xy}$
9	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = e^x + e^y$
10	$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = -10(\sin \pi x - 2 \cos \pi y^2)$

Листинги к теме 7. Задача Дирихле для уравнения Пуассона.

```
function b=bCalc(x,y,h)
n=length(x);b=zeros((n-2)*(n-2),1);k=1;
for i=2:n-1
    for j=2:n-1
        b(k,1)=
10*(h*pi)^2*sin(pi*x(i))*sin(3*pi*y(j));
        k=k+1;
    end%j
end%i
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Err=ExactComparison(x,y,uu,n)
k=1;Err=0;
for i=2:n-1
    for j=2:n-1
        ErrNew=abs(ut_ik(x,i,y,j)-uu(k));
        if ErrNew>Err
            Err=ErrNew;
        end%if
        k=k+1;
    end%j
end%i
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%5
function [f b]=fbCalc(x,y,h)
n=length(x);
n2=(n-2)*(n-2);
f=zeros(n2,1);b=f;
k=1;
for i=2:n-1
    for j=2:n-1
        f(k,1)= - 10*pi^2*sin(pi*x(i))*...
            sin(3*pi*y(j));
        % f(k,1)= - 10;
        b(k,1)=h^2*f(k,1); k=k+1;
```

```

        end%j
    end%i
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %PoissonTask.m
    clear all;close all;
    x0=0;x1=1;y0=0;y1=1;n=22;
    h=(x1-x0)/(n-1);x=x0:h:x1;y=y0:h:y1;
    %u=zeros(n,n);f=zeros((n-2)*(n-2),1);
    %заполнение точного решения
    Ut=ut(x,y);
    % Выведем u(x,t) как поверхности
    [XX,YY]=meshgrid(y0:h:y1,x0:h:x1); ZZ=Ut;
    %прореживаем графики с шагом hh
    hh=2;
    XX1=XX(1:hh:end,1:hh:end);YY1=YY(1:hh:end,1:hh
:end);ZZ1=ZZ(1:hh:end,1:hh:end);
    %prt(StrName,StrZ,StrT,Y1, X1, Z1)
    prt('Точное           решение','u_T(x,y)',['Сетка
',num2str(n),' \times ',num2str(n)],YY1, XX1, ZZ1);
    % Граничные условия по x: u занулена вся, с
    границей.
    %Блоки для Матрицы СЛАУ
    nn=n-2;
    I = speye(nn,nn);
    E = sparse(2:nn,1:nn-1,1,nn,nn);
    D = E+E'-2*I;
    A = kron(D,I)+kron(I,D);
    %заполнение f и b для внутренних узлов по
    строкам
    [f b]=fbCalc(x,y,h);
    %решение СЛАУ
    uu=A\b;
    %свернем uu в двумерный массив u
    u=ReDim(uu,n);ZZ=u;ZZ2=ZZ(1:hh:end,1:hh:end);

```

```

prt('Численное решение','u(x,y)',...
['Сетка ',num2str(n),' \times ',num2str(n)],...
YY1,XX1,ZZ1);
prt(' Погрешность ',...
'|u_T(x,y)-u(x,y)|',...
['Сетка ',num2str(n),' \times ',num2str(n)],...
YY1,XX1, ZZ2-ZZ1);
%%%%%%%%%%
%PoissonTaskErrorOnly.m
clear all;close all;tic;
x0=0;x1=1;y0=0;y1=1;
H=0.25*logspace(0,-2,3);
ERR=zeros(1,length(H));
for k=1:length(H)
tic; h=H(k);
n=(x1-x0)/h+1;x=x0:h:x1;y=y0:h:y1;
%b=zeros((n-2)*(n-2),1);
%Блоки для Матрицы СЛАУ
nn=n-2;
I = speye(nn,nn);
E = sparse(2:nn,1:nn-1,1,nn,nn);
D = E+E'-2*I;
A = kron(D,I)+kron(I,D);
disp(['n=',num2str(n),...
' Блоки и Матрица СЛАУ вычислены']);
%заполнение f и b для
%внутренних узлов по строкам
b=bCalc(x,y,h);
disp(...
'заполнение b для внутрени. узлов проведено');
%решение СЛАУ
uu=A\b;
disp('решение СЛАУ получено');
%Вычисление ошибки

```

```

ERR(k)=ExactComparison(x,y,u,n);
disp('Вычисление ошибки завершено');
%Вывод результата
% format long;
% ERR(1,k)=h
% ERR(2,k)=Err
toc;
% ERR(3,k)=tElapsed;
% ERR
k
end
loglog(H,ERR);grid on;
% save 'err02.dat' ERR -ascii
% figure('color','w');
loglog(ERR(1,:),ERR(2,:),ERR(1,:),ERR(3,:));
grid on;
%%%%%%%%%%%%%
%PoissonTaskKronOnly.m
clear all;close all;
x0=0;x1=1;y0=0;y1=1;n=60;
h=(x1-x0)/(n-1);x=x0:h:x1;y=y0:h:y1;
%u=zeros(n,n);
%f=zeros((n-2)*(n-2),1);%uu=f;b=f;
%заполнение точного решения
Ut=ut(x,y);
% Выведем Ut(x,t) как поверхность
[XX,YY]=meshgrid(y0:h:y1,x0:h:x1); ZZ=Ut;
% prt('Точное решение',...
['Сетка ',num2str(n),' \times ',num2str(n)],...
YY, XX, ZZ);
% Граничные условия по x:
% u занулена вся, с границей.
%Блоки для Матрицы СЛАУ
nn=n-2;

```

```

I = speye(nn,nn) ;
E = sparse(2:nn,1:nn-1,1,nn,nn) ;
D = E+E'-2*I;
A = kron(D,I)+kron(I,D) ;
%заполнение f и b для внутренних узлов по
строкам
[f b]=fbCalc(x,y,h) ;
%решение СЛАУ
uu=A\b;
%свернем uu в двумерный массив u
u=ReDim(uu,n) ;Z1=u;
% Выведем u(x,t) и погрешность как поверхности
%      prt('Численное      решение' , ['Сетка
%',num2str(n) , '\times' , num2str(n)] ,YY,XX, Z1) ;
% prt(' Пгрешность ' ,...
%['Сетка ' ,num2str(n) , '\times' , num2str(n)] ,...
%YY,XX, ZZ-Z1) ;
figure('name','Errors') ;ZZ=Ut-u;
surf1(YY, XX, ZZ) ;
shading interp;
colormap gray;
%%%%%%%%%%%%
%PoissonTask.m
clear all;close all;
x0=0;x1=1;y0=0;y1=1;n=6;h=(x1-x0)/(n-1) ;
x=x0:h:x1;y=y0:h:y1;
u=zeros(n,n) ;Ut=u;
f=zeros((n-2)*(n-2),1) ;b=f;
%заполнение f и точного решения
for i=1:n
    for j=1:n
        Ut(i,j)=sin(pi*x(i))*sin(3*pi*y(j)) ;
        %Ut(i,j)=100*sin(x(i)*y(j)) ;
        %Ut(i,j)=sin(pi*x(i))*cos(pi*y(j)) ;

```

```

        %Ut(i,j)=exp(x(i)*y(j));
        %Ut(i,j)=exp(-x(i))+exp(-y(j));
        %Ut(i,j)=6*(x(i)+2*y(i)^2);
    end%k
end%i
% Выведем Ut(x,t) как поверхность
[XX,YY]=meshgrid(y0:h:y1,x0:h:x1);
figure('Name','Точное решение','color','w');
mesh(XX,YY,Ut);colormap('gray')
% Граничные условия по x: u занулена вся,
% с границей.
%Блоки для Матрицы СЛАУ
D=diag(ones((n-2),1));
T= (- 4)*diag(ones((n-2),1))+...
    diag(ones((n-3),1),-1)+...
    diag(ones((n-3),1),1);
Z=zeros((n-2),(n-2));
A=[ T D Z Z;D T D Z;Z D T D;Z Z D T];
%заполнение f для внутренних узлов по строкам
%в одномерной нумерации узлов
k=1;
for i=2:n-1
    for j=2:n-1
        f(k)=
10*pi^2*sin(pi*x(i))*sin(3*pi*y(j));
        %f(k)=-
100*(x(i)^2+y(j)^2)*sin(x(i)*y(j));
        %f(k)=-
2*pi^2*sin(pi*x(i))*cos(pi*y(j));
        %f(k)=-10*(x(i)+y(i))*exp(x(i)*y(j));
        %f(k)=exp(x(i))+exp(y(j));
        %f(k)=-10*(sin(pi*x(i))-...
%2*cos(pi*y(i)^2));
        b(k)=h^2*f(k); k=k+1;
    end
end

```

```

        end%j
    end%i
    %решение СЛАУ
    uu=A\b;
    %свернем одномерное решение uu
    % в двумерный массив u
    k=1;
    for i=2:n-1
        for j=2:n-1
            u(i,j)=uu(k);
            k=k+1;
        end%j
    end%i
    figure('Name','Digital Solution','color','w');
    mesh(XX,YY,u);zlim([-1 1]);
    %Вывод ошибки в граф. окно.
    figure('Name',...
        'Error of Digital Solution','color','w');
    mesh(XX,YY,abs(u-Ut));
    %%%%%%%%%%
    function prt(StrName,StrZ,StrT,Y1, X1, Z1)
    %StrName Name figure - string
    %StrZ - xlabel - string
    %StrT - title - string
    figure('Name',StrName,'Color','w');
    ha = axes;
    hm = mesh(Y1, X1, Z1);
    colormap gray
    set(hm, 'LineWidth', 2, 'EdgeColor', 'k')
    set(ha, 'FontSize', 16, 'LineWidth', 2,...
        'GridLineStyle', '-', ...
        'FontName', 'Times New Roman')
    hx = xlabel('{\it x}');
    set(hx, 'FontSize', 16,'FontName',...

```

```

'Times New Roman');
hy = ylabel('{\it y}');
set(hy, 'FontSize', 16, 'FontName', ...
'Times New Roman');
hz = zlabel(StrZ);
set(hz, 'FontSize', 16, 'FontName', ...
'Times New Roman');
htt = title(StrT);
set(htt, 'FontSize', 16, 'FontName', ...
'Times New Roman');
%%%%%%%%%%%%
function u=ReDim(uu,n)
u=zeros(n,n);
k=1;
for i=2:n-1
    for j=2:n-1
        u(i,j)=uu(k);
        k=k+1;
    end%j
end%i
%%%%%%%%%%%%
%Ris_7_5_Acc_Solut_Poisson.m
clear all;close all;
x0=0;x1=1;y0=0;y1=1;n=60;
h=(x1-x0)/(n-1);x=x0:h:x1;y=y0:h:y1;
%u=zeros(n,n);
%f=zeros((n-2)*(n-2),1);%uu=f;b=f;
%заполнение точного решения
Ut=ut(x,y);
% Выведем Ut(x,t) как поверхность
[XX,YY]=meshgrid(y0:h:y1,x0:h:x1); ZZ=Ut;
%mesh(YY, XX, ZZ);
figure;
surf1(YY, XX, ZZ);

```



```

shading interp;
colormap gray;
hold on;%shading faceted;
%figure;
Y=YY(1:10:end,1:10:end) ;
X=XX(1:10:end,1:10:end) ;
Z=ZZ(1:10:end,1:10:end) ;
mesh(Y, X, Z) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=ut(x,y)
n=length(x) ;
z=zeros(n) ;
for i=1:n
    for j=1:n
        z(i,j)=sin(pi*x(i))*sin(3*pi*y(j)) ;
    end%j
end%i
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function z=ut_ik(x,i,y,j)
z=sin(pi*x(i))*sin(3*pi*y(j)) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Литература

1. А. Н. Тихонов, А. А. Самарский. Уравнения математической физики. 6-е изд., М., Изд-во МГУ, 1999 г., 798 с.
2. Л. Ф. Шампайн, И. Гладел, С. Томпсон. Решение обыкновенных дифференциальных уравнений с использованием MATLAB. Учебное пособие. Перевод с английского И. А. Макарова. СПб.: Издательство «Лань», 2009. – 304 с., ил.
3. Н. Н. Калиткин. Численные методы. Под редакцией А. А. Самарского. М., Наука, Главная редакция физико-математической литературы. 1978, 512 с., ил.