



Министерство образования  
Республики Беларусь

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

---

---

Кафедра «Теоретическая механика»

# ЧИСЛЕННЫЕ МЕТОДЫ

*Лабораторный практикум*

Часть 2

**МАТЕМАТИЧЕСКАЯ ОБРАБОТКА  
РЕЗУЛЬТАТОВ ЭКСПЕРИМЕНТАЛЬНЫХ  
ДАННЫХ**

Минск  
БНТУ  
2010

Министерство образования Республики Беларусь  
БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ

---

Кафедра «Теоретическая механика»

# ЧИСЛЕННЫЕ МЕТОДЫ

*Лабораторный практикум  
для студентов специальности  
1-55 01 03 «Компьютерная мехатроника»*

В 2 частях

Часть 2

## МАТЕМАТИЧЕСКАЯ ОБРАБОТКА РЕЗУЛЬТАТОВ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ

М и н с к  
БНТУ  
2010

УДК 519.6 (076.5)

ББК 22.193я7

Ч 67

С о с т а в и т е л и:

*И.С. Куликов, Ю.А. Довга*

Р е ц е н з е н т ы:

профессор, доктор физ.-мат. наук *А.В. Чигарев*;

доцент, кандидат физ.-мат. наук *М.Г. Ботогова*

Ч 67 Численные методы: лабораторный практикум для студентов специальности 1-55 01 03 «Компьютерная мехатроника»: В 2. ч. / сост.: И.С. Куликов, Ю.А. Довга. – Минск: БНТУ, 2010. – Ч. 2: Математическая обработка результатов экспериментальных данных. – 60 с.

ISBN 978-985-525-187-4 (Ч. 2).

Издание представляет собой сборник заданий для лабораторных работ по дисциплине «Численные методы» для студентов специальности 1-55 01 03 «Компьютерная мехатроника» с использованием СКМ Mathematica. В текст включено краткое изложение основных теоретических сведений, знание которых необходимо для сознательного решения задач.

УДК 519.6 (076.5)

ББК 22.193я7

ISBN 978-985-525-187-4 (Ч. 2)

ISBN 978-985-525-188-1

© БНТУ, 2010

## ИНТЕРПОЛЯЦИЯ И ПРЕДСКАЗАНИЕ

**Цель:** определение значений функций с помощью интерполяционных многочленов Лагранжа и Ньютона.

### *Теоретические сведения*

Простейшая задача интерполяции заключается в следующем. Для заданных  $n + 1$  точек  $x_i = x_0, x_1, \dots, x_n$ , которые называются узлами интерполяции, и значений в этих точках некоторой функции  $f(x_i) = y_0, y_1, \dots, y_n$  построить полином  $\varphi(x)$  (интерполяционный полином) степени  $n$  вида

$$\varphi(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0, \quad (7.1)$$

принимающий в узлах  $x_i$  те же значения  $y_i$ , что и функция  $f(x_i)$ :

$$\varphi(x_0) = y_0, \varphi(x_1) = y_1, \dots, \varphi(x_n) = y_n, \quad i = 1, \dots, n. \quad (7.2)$$

Для каждого набора точек имеется только один интерполяционный многочлен, степени не больше  $n$ . Однозначно определенный многочлен может быть представлен в различных видах. Рассмотрим интерполяционные многочлены Ньютона и Лагранжа.

Необходимость интерполяции функций в основном связана с двумя причинами:

1) функция  $f(x)$  имеет сложное аналитическое описание, вызывающее определенные трудности при его использовании (например,  $f(x)$  является спецфункцией: гамма-функцией, эллиптической функцией и др.);

2) аналитическое описание функции  $f(x)$  неизвестно, т.е.  $f(x)$  задана таблично. При этом необходимо иметь аналити-

ческое описание приближенно представляющее  $f(x)$  (например, для вычисления значений  $f(x)$  в произвольных точках, определения интегралов и производных от  $f(x)$  и т.п.).

### Глобальная интерполяция.

Простейшим видом глобальной интерполяции является параболическая интерполяция, когда используются выше описанные условия (7.2), для отыскания неизвестных  $n+1$  коэффициентов  $a_0, a_1, \dots, a_n$  выражения (7.1) получают систему из  $n+1$  уравнений:

$$\begin{cases} a_n x_0^n + a_{n-1} x_0^{n-1} + \dots + a_1 x_0 + a_0 = y_0, \\ a_n x_1^n + a_{n-1} x_1^{n-1} + \dots + a_1 x_1 + a_0 = y_1, \\ \dots \\ a_n x_n^n + a_{n-1} x_n^{n-1} + \dots + a_1 x_n + a_0 = y_n. \end{cases} \quad (3)$$

Формула *Лагранжа* является наиболее общей, может применяться к таким узлам интерполяции, что расстояние между соседними узлами – не постоянная величина:

$$L_n(x) = \sum_{i=0}^n \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)} y_i. \quad (7.4)$$

Построение интерполяционного многочлена в форме *Ньютона* применяется главным образом для функций, заданных таблично с *равноотстоящими* значениями аргумента ( $h_i = x_{i+1} - x_i = \text{const}$ ). Введем предварительно понятие конечных разностей:

$$\Delta y_i = y_{i+1} - y_i, \quad (i = 0, 1, \dots, n-1)$$

$$\Delta^2 y_i = \Delta y_{i+1} - \Delta y_i, \quad (i = 0, 1, \dots, n-2)$$

$$\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i, \quad (i = 0, 1, \dots, n-k)$$

С учетом введенных обозначений, *первая интерполяционная формула Ньютона* имеет вид  $t = \frac{x - x_0}{h}$ ,

$$P_{n1}(x) = P_{n1}(x_0 + th) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_0 + \dots + \frac{t(t-1)\dots(t-n+1)}{n!}\Delta^n y_0. \quad (7.5)$$

Вторая интерполяционная формула имеет вид  $t = \frac{x - x_n}{h}$ ,

$$P_{n2}(x) = P_{n2}(x_n + th) = y_n + t\Delta y_{n-1} + \frac{t(t-1)}{2!}\Delta^2 y_{n-2} + \dots + \frac{t(t-1)\dots(t-n+1)}{n!}\Delta^n y_0. \quad (7.6)$$

Формулу (7.5) удобно использовать, если точка находится вблизи начала таблицы, а формулу (7.6) – если точка находится вблизи конца таблицы.

Однако интерполяция при большом числе узлов приводит к необходимости работать с многочленами высокой степени, что неприемлемо как с точки зрения вычислений, так и из-за склонности таких многочленов к осцилляции (колебаниям) между узлами сетки. Поэтому на практике часто используют интерполяцию кусочными многочленами (или локальную интерполяцию).

### **Локальная интерполяция**

При большом количестве узлов интерполяции сильно возрастает степень интерполяционных многочленов, что делает их неудобными для проведения вычислений.

Высокой степени многочленов можно избежать, разбив отрезок интерполирования на несколько частей, с построением в каждой части своего интерполяционного полинома. Такой метод называется *интерполяцией сплайнами*. Наиболее распространенным является построение на каждом отрезке  $[x_i, x_{i+1}]$ ,

$i = 0, \dots, n-1$  кубической функции. При этом сплайн – кусочная функция, на каждом отрезке заданная кубической функцией, является кусочно-непрерывной вместе со своими первой и второй производной.

### Полиномиальная интерполяция и аппроксимация средствами Mathematica

Для решения задач интерполяции и аппроксимации функций, заданных рядом узловых точек, в Mathematica используются следующие функции:

1) **InterpolatingFunction[range, table]** возвращает интерполирующую функцию, позволяющую вычислять промежуточные значения в заданном диапазоне range для таблицы table;

2) **InterpolatingPolynomial[data, var]** возвращает полином (степенной многочлен) по переменной var, значения которого в узловых точках точно совпадают с данными из списка data. Он может иметь форму  $\{\{x_1, f_1\}, \{x_2, f_2\}, \dots\}$  или  $\{f_1, f_2, \dots\}$  (во втором случае  $x_i$  принимают значения  $1, 2, \dots$ ). Вместо  $f_i$  может быть список  $\{f_i, df_i, ddf_i, \dots\}$ , указывающий значения производных в точках  $x_i$ ;

3) **Interpolation[data]** конструирует объект InterpolatingFunction.

**InterpolationOrder** – опция функции Interpolation, указывающая степень подходящего полинома. При ее значении, равном 1б, осуществляется кусочно-линейная интерполяция. Целое значение, большее единицы, задает степень глобальной полиномиальной интерполяции.

```

In[10]:= data = Table[{x, x2 + 1}, {x, 1, 5}]
Out[10]= {{1, 2}, {2, 5}, {3, 10}, {4, 17}, {5, 26}}

funi = Interpolation[data]

Out[11]= InterpolatingFunction[{{1, 5}}, <>]

In[12]:= {funi[1.5], funi[3], funi[4.5]}
Out[12]= {3.25, 10, 21.25}

```

Рис. 7.1. Вычисление промежуточных значений функции

**SplineFit[points, type]** – функция пакета **NumericalMath`SplineFit`**, позволяющая производить интерполяцию сплайнами. *Type* может принимать значения **Cubic**, **Bezier** и **CompositeBezier**.

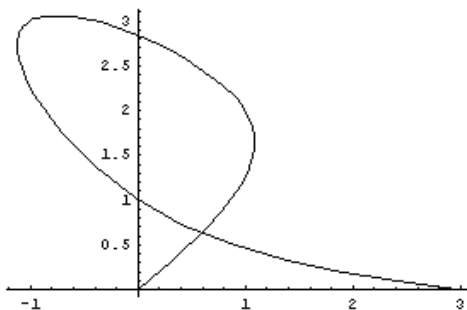
```

In[1]:= << NumericalMath`SplineFit`
pts = {{0, 0}, {1, 2}, {-1, 3}, {0, 1}, {3, 0}};
spline = SplineFit[pts, Cubic]
spline[1.4]
ParametricPlot[spline[u], {u, 0, 4}, PlotRange -> All,
Compiled -> False]

Out[3]= SplineFunction[Cubic, {0., 4.}, <>]

Out[4]= {0.265143, 2.70171}

```



Out[5]= - Graphics -

Рис. 7.2. Интерполяция сплайнами в Mathematica



■ Полномочная аппроксимация с визуализацией

```
In[1]:= data := {{0, .9}, {2, 8.1}, {3, 17}, {4, 33}};
p[x_] = Collect[InterpolatingPolynomial[data, x], x]
```

```
Out[2]= 0.9 + 2.74167 x - 0.4625 x2 + 0.445833 x3
```

```
In[3]:= g1 := Plot[p[x], {x, 0, 4}];
g2 := ListPlot[data, PlotStyle -> {PointSize[0.02]}];
Show[g1, g2, PlotRange -> {0, 35}];
```

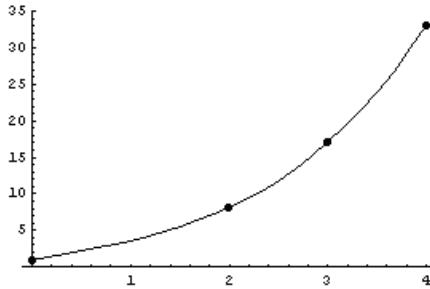


Рис.3. Построение интерполяционной функции

### Задания к лабораторной работе

**Задание 1.** Вычислить значения заданной функции  $y_i = f(x_i)$  в узлах интерполяции  $x_i = a + hi$ , где  $h = \frac{b-a}{10}$ ,  $i = 0, 1, \dots, 10$  на отрезке  $[a, b]$ .

#### Варианты заданий

№ варианта	$f(x)$	$[a, b]$	№ варианта	$f(x)$	$[a, b]$
1	$\sin(x^2)$	$[0, 2]$	9	$x \cos(x + \ln(1+x))$	$[1, 5]$
2	$\cos(x^2)$	$[0, 2]$	10	$10 \ln \frac{2x}{1+x}$	$[1, 5]$

3	$e^{\sin(x)}$	[0,5]	11	$\sin x^2 \cdot e^{-\left(\frac{x}{2}\right)^2}$	[0,3]
4	$\frac{1}{0.5+x^2}$	[0,2]	12	$\cos(x+\cos^3 x)$	[0,2]
5	$e^{-(x+\sin x)}$	[2,5]	13	$\cos(x+e^{\cos x})$	[3,6]
6	$\frac{1}{1+e^{-x}}$	[0,4]	14	$\cos(2x+x^2)$	[0,1]
7	$\sin(x+e^{\sin x})$	[0,3]	15	$e^{\cos x} \cos x^2$	[0,2]
8	$e^{-(x+\frac{1}{x})}$	[1,3]	16	$\sin(x+e^{x+2})$	[1,2]

**Задание 2.** По вычисленной таблице  $(x_i, y_i)$  провести параболическую интерполяцию.

Для нахождения коэффициентов искомого полинома (7.1) необходимо составить систему линейных алгебраических уравнений (7.3). Решить систему с помощью встроенной функции **Solve[]**. Построить график интерполяционного многочлена и отметить на нем узловые точки  $(x_i, y_i)$ .

**Задание 3.** Для вычисления табличной функции составить формулу интерполяционного многочлена Лагранжа, используя операторы суммирования и перемножения по дискретному аргументу, а также функцию **if []**. Построить график интерполяционного многочлена и отметить на нем узловые точки  $(x_i, y_i)$ .

**Задание 4.** Провести интерполирование с помощью первой и второй интерполяционных формул Ньютона. Построить график интерполяционного многочлена и отметить на нем узловые точки  $(x_i, y_i)$ .

**Задание 5.** Провести линейную интерполяцию заданной функции средствами Mathematica. Построить график полученной функции и отметить узловые точки  $(x_i, y_i)$ .

**Задание 6.** Провести кубическую сплайн-интерполяцию заданной функции средствами пакета Mathematica. Построить график полученной функции.

Вычислить значения заданной функции  $y_i = f(x_i)$  в точках  $x_i = a + i/10$ , где  $i = 0, 1, \dots, 10(b - a)$  на отрезке  $[a, b]$ . Выполнить предсказание (экстраполяцию) полученного вектора в последующих 10 точках по последним 7 значениям функции.

Отобразить графически имеющиеся данные, предсказанные данные и истинный вид функции  $f(x)$ .

### *Лабораторная работа № 8*

## **АППРОКСИМАЦИЯ ФУНКЦИЙ. МЕТОД НАИМЕНЬШИХ КВАДРАТОВ**

**Цель:** изучить способы аппроксимации функций.

### *Теоретические сведения*

Аппроксимация функций заключается в приближенной замене заданной функции  $f(x)$  некоторой функцией  $\varphi(x)$  так, чтобы отклонение функции  $\varphi(x)$  от  $f(x)$  в заданной области было наименьшим. Функция  $\varphi(x)$  при этом называется аппроксимирующей. Значащую цифру числа  $a^*$  называют верной, если абсолютная погрешность числа не превосходит единицы разряда, соответствующего этой цифре.

Рассмотрим принципиально другой, нежели интерполирование, подход к приближению функции заданной таблицей своих значений  $f(x_i) = y_i$  в точках  $x_i, i = 0, 1, 2, \dots, n$ . Будем искать приближение в виде полинома степени  $m$

$$P_m(x) = a_0 + a_1x + a_2x^2 + \dots + a_mx^m,$$

причем такого, чтобы значения этого полинома в табличных точках  $x_i$  не слишком сильно отличались от значений функции  $y_i$ ,

т.е. разности  $|P_m(x_i) - y_i|$  должны быть малы. Для оценки степени приближения полинома  $P_m(x)$  к точкам  $y_i$  используется метод наименьших квадратов, согласно которому наилучшее приближение достигается при минимуме суммы квадратов отклонений полинома от значений функции

$$\Delta(a_0, a_1, a_2, \dots, a_m) = \sum_{i=0}^m (P_m(x_i) - y_i)^2 \rightarrow \min.$$

Таким образом, не требуется, чтобы аппроксимирующая функция проходила через все заданные точки, что особенно важно при аппроксимации данных, заведомо содержащих погрешности.

Очевидно, что при  $m=n$  решением такой задачи является интерполяционный многочлен, так как в табличных точках  $x_i$  его значения  $P_m(x)$  равны значениям функции  $y_i$  и минимум квадратов разностей будет равен нулю  $\Delta(a_0, a_1, a_2, \dots, a_m) = 0$ .

При  $m < n$  поставленная задача будет иметь единственное решение, которое можно найти, приравняв частные производные  $\frac{\partial \Delta(a_0, a_1, a_2, \dots, a_m)}{\partial a_i}$  к нулю. При  $m > n$  поставленная задача

будет иметь множество решений, для поиска одного из них можно задать произвольно  $m-n-1$  коэффициентов  $a_i$ , а остальные найти, решив задачу интерполяции.

Остановимся подробнее на случае  $m < n$ , т.е. когда число табличных точек больше числа параметров аппроксимационной функции. Приравняв нулю частные производные функции  $\Delta(a_0, a_1, a_2, \dots, a_m)$ , получим систему уравнений

$$\frac{\partial \sum_{i=0}^m (a_0 + a_1 x_i + a_2 x_i^2 + \dots + a_m x_i^m - y_i)^2}{\partial a_k}, \quad k = 0, 1, 2, \dots, m$$

или после преобразований

$$a_0 \sum_{i=0}^n x_i^k + a_1 \sum_{i=0}^n x_i^{k+1} + \dots + a_m \sum_{i=0}^n x_i^{k+m} = a_0 \sum_{i=0}^n y_i \cdot x_i^k,$$

$$k = 0, 1, 2, \dots, m.$$

Таким образом, для определения коэффициентов  $a_k$  получаем систему линейных уравнений с симметрической матрицей коэффициентов

$$\begin{bmatrix} n+1 & \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \dots & \sum_{i=0}^n x_i^m \\ \sum_{i=0}^n x_i & \sum_{i=0}^n x_i^2 & \sum_{i=0}^n x_i^3 & \dots & \sum_{i=0}^n x_i^{m+1} \\ \dots & \dots & \dots & \dots & \dots \\ \sum_{i=0}^n x_i^m & \sum_{i=0}^n x_i^{m+1} & \sum_{i=0}^n x_i^{m+2} & \dots & \sum_{i=0}^n x_i^{m+m} \end{bmatrix}$$

и столбцом свободных членов

$$\begin{bmatrix} \sum_{i=0}^n y_i \\ \sum_{i=0}^n y_i x_i \\ \vdots \\ \sum_{i=0}^n y_i x_i^k \end{bmatrix},$$

элементы которых вычисляются по табличным данным. Найденный из решения этой системы полином называется

среднеквадратичным приближением функции заданной таблицей.

Основная сфера применения таких полиномов – обработка экспериментальных данных (построение эмпирических формул). Дело в том, что интерполяционный полином, построенный по значениям функции, полученным с помощью эксперимента, будет испытывать сильное влияние «экспериментального шума», к тому же при интерполировании узлы интерполяции не могут повторяться, т.е. нельзя использовать результаты повторных экспериментов при одинаковых условиях. Среднеквадратичный же полином сглаживает шумы и позволяет использовать результаты многократных экспериментов.

В ряде случаев аппроксимирующую функцию ищут не в виде полинома, а в виде некоторой другой зависимости. Если применить к этой зависимости метод наименьших квадратов, то получим систему нелинейных уравнений, решить которую весьма затруднительно. Чтобы облегчить решение можно произвести замену переменных таким образом, чтобы линеаризовать функцию  $y(x)$ . Ниже приведены замены переменных для некоторых часто используемых нелинейных функций.

Функция	Замена переменных
$y = a \cdot b^x$	$q = \lg y, z = \lg x, p = \lg a, s = \lg b$
$y = a \cdot x^b$	$q = \lg y, z = \lg x, p = a, s = \lg b$
$y = a + \frac{b}{x}$	$q = y, z = \frac{1}{x}, p = a, s = b$
$y = \frac{1}{ax + b}$	$q = \frac{1}{y}, z = x, p = a, s = b$
$y = \frac{x}{ax + b}$	$q = \frac{x}{y}, z = x, p = a, s = b$

Если значения  $x$  или  $y$  отрицательны или нулевые, то логарифмирование и деление на ноль выполнить невозможно, по-

этому следует сдвинуть шкалу  $x$  или  $y$  в положительную область.

## Нелинейная регрессия

Рассмотрим наиболее простые случаи *нелинейной* регрессии: гиперболу, экспоненту и параболу. При нахождении коэффициентов гиперболы и экспоненты используют прием приведения нелинейной регрессионной зависимости к линейному виду. Это позволяет использовать для вычисления коэффициентов функций регрессии вышеприведенные формулы.

## Регрессия в Mathematica

В **Mathematica** для решения задач регрессии используется функция ядра **Fit**: **Fit[data, funs, vars]**. Эта функция ищет приближение для списка данных *data* наименьших квадратов в виде линейной комбинации функций *funs* переменных *vars*.

```
In[12]= data = {{0, 5}, {1, 2}, {3, 5}, {4, -2}, {5, 9}, {8, 11}};
dataplot = ListPlot[data, PlotStyle -> PointSize[0.02]];
Fit[data, x + 2, x]
plotline = Plot[%, {x, -5, 15}];
Show[dataplot, plotline];
```

```
Out[14]= 0.90583 (2 + x)
```

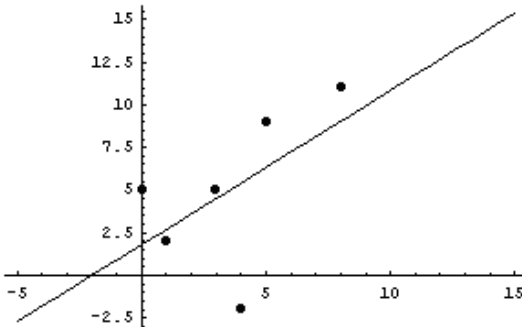


Рис. 8.1. Пример использования функции Fit[]

**FindFit[data, expr, pars, vars]** находит численные значения параметров **pars**, при которых выражение **expr** дает лучшее приближение **fit** данных **data** функциями переменных **vars**.

```
In[32]:= data = {{0, 5}, {1, 2}, {3, 5}, {4, -2}, {5, 9}, {8, 11}};
f2 = FindFit[data, c + d x + e x^2, {c, d, e}, x]
Plot[c + d x + e x^2 /. f2, {x, 0, 8}, AxesOrigin -> {0, 0},
PlotRange -> {Automatic, {-2, 11}},
Epilog -> Prepend[Point /@ data, PointSize[0.02]]];
```

```
Out[33]= {c -> 4.15845, d -> -1.25067, e -> 0.272291}
```

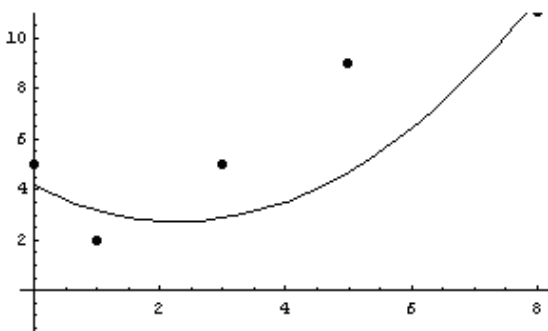


Рис. 8.2. Приближение данных квадратичной функцией

### Задания к лабораторной работе

**Задание 1.** Создайте таблицу экспериментальных данных:

$$x_i = a + hi, \quad i = 0, 1, \dots, 10, \quad h = \frac{b-a}{10} \quad \text{на отрезке } [a, b].$$

Варианты задания.

№ варианта	$y_i$										$[a, b]$	
1	2,86	2,21	2,96	3,27	3,58	3,76	3,93	3,67	3,9	3,64	4,09	[0, 1]



2	1,14	1,02	1,64	1,64	1,96	2,17	2,64	3,25	3,47	3,86	3,36	[-1,1]
3	4,7	4,64	4,57	4,45	4,4	4,43	4,27	4,37	4,42	4,5	4,62	[2, 4]

Окончание таблицы

№ варианта	$y_i$											$[a, b]$
4	0,43	0,99	2,07	2,54	1,67	1,29	1,24	0,66	0,43	0,35	0,7	[2, 4]
5	1,55	1,97	1,29	0,94	0,88	0,09	0,02	0,84	0,81	0,09	0,15	[1, 4]
6	3,24	1,72	1,95	2,77	2,47	0,97	1,75	1,55	0,12	0,7	1,19	[0, 4]
7	2,56	1,92	2,85	2,94	2,39	2,16	2,51	2,1	1,77	2,28	1,7	[-1,2]
8	1,77	0,92	2,21	1,5	3,21	3,46	3,7	4,02	4,36	4,82	4,03	[-1,3]
9	1,53	0,45	1,68	0,12	0,68	2,36	2,58	2,53	3,45	2,7	2,82	[4, 8]
10	2,5	3,9	3,54	4,63	3,87	5,25	4,83	3,24	3,08	3	4,7	[0, 5]
11	2,95	3,38	2,71	2,37	2,29	2,75	2,76	2,74	2,57	2,4	2,99	[1, 5]
12	-0,23	-0,03	-0,98	-0,97	-0,43	-0,91	-0,27	-0,19	0,88	1,06	0,72	[2,4]
13	2,36	0,03	-0,38	-1,33	0,25	-1,36	0,95	3,16	4,03	4,92	4,2	[0, 2]
14	3,82	4,07	3,53	4,83	5,53	5,04	5,09	5,87	5,53	4,72	4,73	[3, 4]
15	2,35	2,16	2,39	2,39	2,18	2,09	2,44	2,56	3,35	3,22	3,65	[-3,4]
16	-1,53	0,45	1,68	-0,12	0,68	2,36	2,58	-2,53	-3,45	2,7	-2,82	[0, 5]

**Задание 2.** Аппроксимировать многочленами 2-й и 6-й степени по методу наименьших квадратов функцию, заданную таблицей значений  $x_i$  и  $y_i$ , и сравнить качество приближений. Построить графики приближений и отметить узловые точки  $(x_i, y_i)$ .

**Задание 3.** Для приведенных в таблице экспериментальных данных  $(x_i, y_i)$  провести линейную регрессию с помощью встроенной функции *Fit* пакета Mathematica. Построить график приближения и отметить на нем узловые точки.

**Задание 4.** Аппроксимировать данные полиномом 4-ой степени, используя возможности пакета Mathematica. Отобразить графически результаты аппроксимации.

**Задание 5.** Аппроксимировать экспериментальные данные из таблицы линейной комбинацией функции:

$$f(x) = a_1 f_1(x) + a_2 f_2(x) + a_3 f_3(x).$$

Отобразить графически узловые точки и результаты проведенной регрессии общего вида.

Варианты задания.

№ варианта	$f_1(x)$	$f_2(x)$	$f_3(x)$
1	$e^x$	$\frac{1}{\sqrt{1+2\cos^2 x}}$	$\sin x$
2	$\frac{1}{1+x^2}$	$e^x$	$\sin(3x)$
3	$\frac{1}{1+x^2}$	$e^{\sin x}$	$x$
4	$\arctg x$	$\ln(\ln(x))$	$\sin x$
5	$e^{\frac{-x}{2}}$	$\frac{1}{x}$	$e^{-x}$
6	$\frac{1+x}{2+x}$	$\cos \frac{x}{10}$	$\cos x$
7	$\frac{1}{1+e^{x^2}}$	$\sqrt{1+x^2}$	$\cos x$
8	$\cos \frac{x}{2}$	$2 - \cos x$	$\sin \frac{x}{2}$
9	$\frac{1}{1+e^x}$	$\arctg \sqrt{x}$	$\sin(3x)$
10	$\ln(x+5)$	$\sqrt{1+x}$	$\sin x$
11	$\frac{1}{x}$	$\sqrt{1+x}$	$\frac{1}{x^2}$
12	$\cos x$	$\frac{1}{1+x+x^2}$	$\frac{1}{1+x}$
13	$e^x$	$\cos(4x)$	$e^{\frac{-x}{2}}$
14	$\sqrt{1+e^{-x}}$	$e^{\frac{x}{3}}$	$\sin^2(3x)$
15	$\frac{1}{1+x+x^2}$	$\cos \frac{x}{10}$	$\sin \frac{x}{10}$
16	$3^x$	$\frac{1}{x}$	$x$

```

In[1]:= << Statistics`LinearRegression`
In[2]:= data = {{0.055, 90}, {0.091, 97}, {0.138, 107}, {0.167, 124}, {0.182, 142}, {0.211, 150},
               {0.232, 172}, {0.248, 189}, {0.284, 209}, {0.351, 253}};
In[3]:= Regress[data, {1, Sin[x], Cos[x]}, x]
Out[3]= {ParameterTable → 1
          Estimate      SE      TStat      PValue
          Sin[x]      187.447  113.485  1.65173  0.142571
          Cos[x]     -2014.2  547.473 -3.67909  0.00786913
          RSquared → 0.985544, AdjustedRSquared → 0.981414, EstimatedVariance → 51.5957,
          ANOVATable → Model  2  24622.9  12311.5  238.614  3.63219 × 10-7
                       Error  7  361.17  51.5957
                       Total  9  24984.1
In[4]:= func = Fit[data, {1, Sin[x], Cos[x]}, x]
Out[4]= 2085.59 - 2014.2 Cos[x] + 187.447 Sin[x]

```

Рис. 8.3. Приближение данных комбинацией функций

**Задание 6.** Аппроксимировать экспериментальные данные из таблицы функцией вида  $f(x) = e^{u_0 + u_1 x + u_2 x^2}$

```

In[5]:= << Statistics`NonlinearFit`
In[6]:= data = {{1.0, 1.0, .126}, {2.0, 1.0, .219},
                {1.0, 2.0, .076}, {2.0, 2.0, .126}, {1.1, .0, .186}};
In[7]:= NonlinearFit[data,
                    theta1 theta3 x1 / (1 + theta1 x1 + theta2 x2), {x1, x2},
                    {theta1, theta2, theta3}]
Out[7]= 
$$\frac{2.44277 x_1}{1 + 3.13151 x_1 + 15.1594 x_2}$$


```

Рис. 8.4. Приближение данных функциями нескольких переменных

**Задание 7.** Выполнить сглаживание экспериментальных данных с помощью встроенных функций пакета Mathematica. Изобразить полученные результаты.

```

In[1]:= <<Statistics`DataSmoothing`

In[2]:= data = {{a, b}, {c, d}, {e, f}, {g, h}};
        data1 = {7., 4., 5., 3., 8., 5., 2., 3., 6., 8.};

In[4]:= MovingAverage[data, 3]
        MovingAverage[data1, 2]

Out[4]= {{1/3 (a + c + e), 1/3 (b + d + f)},
         {1/3 (c + e + g), 1/3 (d + f + h)}}

Out[5]= {5.5, 4.5, 4., 5.5, 6.5, 3.5, 2.5, 4.5, 7.}

In[6]:= MovingMedian[data1, 3]

Out[6]= {5., 4., 5., 5., 5., 3., 3., 6.}

```

Рис.8. Сглаживание данных средствами пакета Mathematica

### *Лабораторная работа № 9.*

## **ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ И ДИФФЕРЕНЦИРОВАНИЕ.**

**Цель:** изучить методы численного дифференцирования и численного нахождения определенных интегралов.

### *Теоретические сведения.*

#### **Численное интегрирование.**

Необходимость вычисления значений определенных интегралов при моделировании возникает достаточно часто. Формула

Ньютона-Лейбница  $I^* = \int_a^b f(x)dx = F(b) - F(a)$  имеет

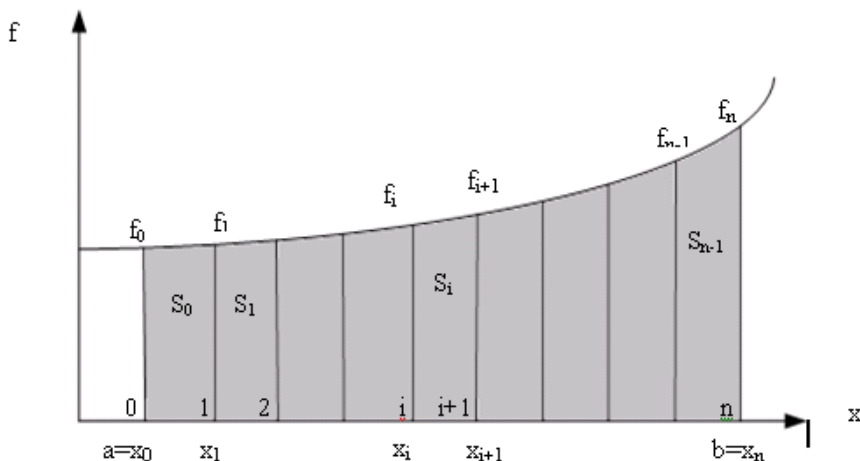
ограниченное применение:

- во-первых, не позволяет вычислить интегралы от таблично заданной подынтегральной функции  $f(x)$ ;
- во-вторых, не всякая подынтегральная функция имеет первообразную  $F(x)$ .

Численные методы интегрирования универсальны: позволяют вычислить значение определенного интеграла непосредственно по значениям подынтегральной функции  $f(x)$ , независимо от способа ее задания или вида аналитического выражения.

### Квадратурные формулы

**Геометрический смысл** определенного интеграла (рис. 9.1) – площадь криволинейной трапеции, ограниченной осью  $OX$ , кривой  $f(x)$ , и прямыми  $x=a$  и  $x=b$ .



Численные методы интегрирования основаны на различных способах оценки этой площади, поэтому полученные формулы численного интегрирования называются квадратурными (формулами вычисления площади).

Рассмотрим получение и применение простейших формул.

Отрезок  $[a, b]$  делят на  $n$  необязательно равных частей – элементарных отрезков. Принято такое деление отрезка называть сеткой, а точки  $x_0, x_1, \dots, x_n$  – узлами сетки.

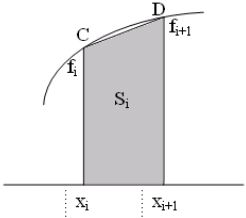
Если сетка равномерная, то  $h = \frac{b-a}{n}$  – шаг сетки, при интегрировании – шаг интегрирования, а координата  $i$ -го узла вычисляется по формуле

$$x_i = a + i \cdot h, i = \overline{0, n}.$$

Полная площадь криволинейной трапеции состоит из  $n$  элементарных криволинейных трапеций – элементарных площадей:

$$I^* = \sum_{i=0}^{n-1} S_i.$$

Примеры простейших формул для часто используемой равномерной сетки		
Оценка элементарной площади $S_i$ левым прямоугольником	$I^* = \sum_{i=0}^{n-1} S_i \approx \sum_{i=0}^{n-1} f_i \cdot h =$ $= h \sum_{i=0}^{n-1} f_i = h \sum_{i=0}^{n-1} f(x_i) = h \sum_{i=0}^{n-1} f(a + i \cdot h)$	
Оценка элементарной площади $S_i$ правым прямоугольником	$I^* = \sum_{i=0}^{n-1} S_i \approx \sum_{i=0}^{n-1} f_{i+1} \cdot h =$ $= h \sum_{i=1}^n f_i = h \sum_{i=1}^n f(x_i) = h \sum_{i=1}^n f(a + i \cdot h)$	
Оценка элементарной площади $S_i$ центральным прямоугольником	$I^* = \sum_{i=0}^{n-1} S_i \approx \sum_{i=0}^{n-1} \tilde{f}_i \cdot h =$ $= h \sum_{i=0}^{n-1} \tilde{f}_i = h \sum_{i=0}^{n-1} f(x_i + \frac{h}{2}) = h \sum_{i=0}^{n-1} f(a + i \cdot h + \frac{h}{2})$	

Формула трапеций	$\Gamma^* = \sum_{i=0}^{n-1} S_i \approx \sum_{i=0}^{n-1} \frac{f_i + f_{i+1}}{2} \cdot h =$ $= h \cdot \left( \frac{f_0 + f_n}{2} + \sum_{i=1}^{n-1} f_i \right) =$ $= h \cdot \left( \frac{f(a) + f(b)}{2} + \sum_{i=1}^{n-1} f(a + i \cdot h) \right)$	
Формула Симпсона (парабол)	$S_i \approx \frac{h}{6} \cdot (f_i + 4\tilde{f}_i + f_{i+1}) =$ $= \frac{h}{6} \cdot \left( f(x_i) + 4f\left(x_i + \frac{h}{2}\right) + f(x_{i+1}) \right)$ $\Gamma^* \approx \sum_{i=0}^{n-1} \frac{h}{6} (f_i + 4\tilde{f}_i + f_{i+1}) =$ $= \frac{h}{3} \cdot \left[ \frac{f_n - f_0}{2} + \sum_{i=0}^{n-1} (f_i + 2\tilde{f}_i) \right] =$ $= \frac{h}{3} \cdot \left[ \frac{f(b) - f(a)}{2} + \sum_{i=0}^{n-1} \left( f(a + ih) + 2f\left(a + ih + \frac{h}{2}\right) \right) \right]$	На каждом элементарном отрезке подынтегральная функция $f(x)$ заменяется квадратичной параболой, построенной по трем точкам: концам элементарного отрезка, $(x_i, f_i)$ , $(x_{i+1}, f_{i+1})$ и его середине $x_i + \frac{h}{2}$ , $\tilde{f}_i$
Формула Ньютона (грех восьмых)	$\Gamma^* \approx \frac{3h}{8} (f_0 + 3(f_1 + f_2 + f_4 + f_5 + \dots + f_{n-2} + f_{n-1}) + 2(f_3 + f_6 + \dots + f_{n-3} + f_n))$	На каждом элементарном отрезке подынтегральная функция $f(x)$ заменяется кубическим интерполяционным многочленом, построенным по четырем точкам

**Метод неопределенных коэффициентов** состоит в вычислении определенного интеграла  $I = \int_a^b f(x) dx$  с помощью

формулы  $I^* = \int_a^b \varphi(x) dx = \sum_{i=0}^n A_i \cdot y_i$ ,  $I^* \approx I$ , коэффициенты  $A_i$

которой находятся в результате решения следующей системы

уравнений: 
$$\begin{cases} I_0 = A_0 + A_1 + \dots + A_n \\ I_1 = A_0 x_0 + A_1 x_1 + \dots + A_n x_n \\ \dots \\ I_1 = A_0 x_0^n + A_1 x_1^n + \dots + A_n x_n^n \end{cases}$$

где  $I_k = \int_a^b x^k dx = \frac{b^{k+1} - a^{k+1}}{k+1}$ ,  $k = 0, 1, \dots, n$ .

## Метод Монте-Карло

Во многих задачах исходные данные носят случайный характер, поэтому для их решения должен применяться статистико-вероятностный подход. На основе такого подхода и построен метод статистических испытаний, называемый также методом Монте-Карло.

Пусть  $\eta$  – равномерно распределенная на отрезке  $[a, b]$  случайная величина. Тогда  $\int_a^b f(x)dx = \frac{b-a}{n} \sum_{i=0}^n f(\eta_i)$ .

Random[Real, {0, x}, n]	Возвращает равномерно распределенное вещественное случайное число между 0 и $x$ с точностью до $n$ знаков после запятой
Mean[list]	Возвращает среднее арифметическое значение элементов массива list

**Численное дифференцирование** аналитически или таблично заданной функции  $f(x)$  на отрезке  $[a, b]$  в точке  $x = X$  заключается в замене  $f(x)$  интерполяционным полиномом  $\varphi(x)$ , производные  $\frac{d^m \varphi(x)}{d x^m} \approx \frac{d^m f(x)}{d x^m}$  которого можно найти аналитически с помощью соответствующих формул:

$$\frac{d^m f(x)}{d x^m} \approx \frac{d^m \varphi(x)}{d x^m} = \sum_{i=0}^n B_i \cdot y_i.$$

Метод неопределенных коэффициентов предполагает использование в качестве интерполяционного многочлена



$\varphi(x)$  полинома степени  $k = n$ :  $\varphi(x) = (X - x_i)^k$ . Коэффициенты  $B_i$  находятся в результате решения следующей системы уравнений:

$$\begin{cases} D_0 = B_0 + B_1 + \dots + B_n; \\ D_1 = B_0 x_0 + B_1 x_1 + \dots + B_n x_n; \\ \dots \dots \dots \\ D_n = B_0 x_0^n + B_1 x_1^n + \dots + B_n x_n^n, \end{cases}$$

где  $D_k = (X^k)' = k \cdot X^{k-1}$ ,  $k = 0, 1, \dots, n$ .

### Интегрирование и дифференцирование встроенными средствами Mathematica

Функция	
$D[f, x]$	Возвращает частную производную функции $f$ по переменной $x$
$D[f, \{x, n\}]$	Возвращает частную производную $n$ -го порядка функции $f$ по переменной $x$
$D[f, x_1, x_2, \dots]$	Возвращает смешанную производную
$Dt[f, x]$	Возвращает обобщенную производную функции $f$ по переменной $x$
$Dt[f]$	Возвращает полный дифференциал $f$

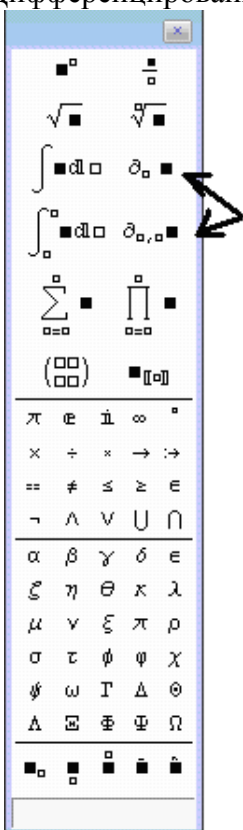
Для вычисления производных более высоких порядков возможно последовательное применение функции  $D$ .

Функция	
$Integrate[f, x]$	Возвращает первообразную (неопределенный интеграл) функции $f$ по переменной $x$
$Integrate[f, \{x, x_{min}, x_{max}\}]$	Возвращает значение определенного интеграла от функции $f$ по переменной $x$ с пределами от $x_{min}$ до $x_{max}$

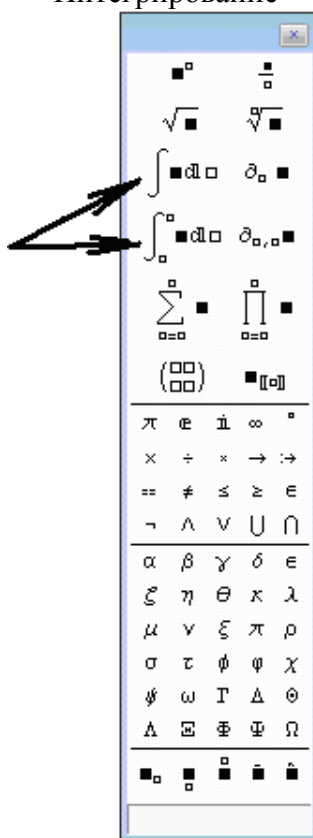
$Integrate [f, \{x, x\ min, x\ max\}, \{y, y\ min, y\ max\}, \dots]$	Возвращает значениекратного интеграла от функции $f$ с пределами от $xmin$ до $xmax$ по переменной $x$ , от $ymin$ до $ymax$ по переменной $y$ и т.д. (кратность реально не ограничена).
$NIntegrate [f, \{x, x\ min, x\ max\}]$	Возвращает численное приближение интеграла от функции $f$ по переменной $x$ в пределах от $xmin$ до $xmax$ .

Для вычисления производных и интегралов можно также пользоваться панелью инструментов BasicInput.

### Дифференцирование



### Интегрирование



Вычисление производных:

Производная от степенного многочлена

```
In[4]:= D[a x^2 + b x + c, x]
```

```
Out[4]= b + 2 a x
```

Пятая производная от  $x^n$

```
In[5]:= D[x^n, {x, 5}]
```

```
Out[5]= (-4 + n) (-3 + n) (-2 + n) (-1 + n) n x^{-5+n}
```

Частная производная от функции  $\frac{\partial^3 \sin(x+y)}{\partial x \partial y \partial x}$

```
In[6]:= D[Sin[x + y], x, y, x]
```

```
Out[6]= -Cos[x + y]
```

Обобщенная производная от  $\sin(x+y)$  по переменной  $x$

```
In[7]:= Dt[Sin[x + y], x]
```

```
Out[7]= Cos[x + y] (1 + Dt[y, x])
```

Рис. 9.1. Вычисление производных средствами Mathematica

### *Задания к лабораторной работе*

**Задание 1.** Определить функцию  $f(x)$  таблично, вычислив значения в точках

$$x_i = a + hi, \quad i = 0, 1, \dots, 8, \quad h = \frac{b-a}{8} \text{ на отрезке } [a, b].$$

Варианты задания.

№	$f(x)$	$[a, b]$	$[c, d]$
1	$\frac{1}{\operatorname{tg}(2x) + 1}$	[0.4, 0.8]	[2, 2.1]
2	$\frac{\cos(3x)}{(1 - \cos(3x))^2}$	[0.8, 1.6]	[-1, -0.9]
3	$\frac{1}{x\sqrt{x^3+4}}$	[0.18, 0.98]	[0.5, 0.6]
4	$\frac{\sin(x)}{1 + \sin(x)}$	[0.8, 1.6]	[2, 2.1]
5	$x^{2\lg(x+2)}$	[0, 0.4]	[1.5, 1.6]
6	$x^{2\operatorname{arctg}\frac{x}{3}}$	[0.8, 1.6]	[1, 1.1]
7	$e^{2x} \cdot \sin(3x)$	[0.4, 1.2]	[2, 2.1]
8	$\frac{\operatorname{ctg}(2x)}{(\sin(2x))^2}$	[0.8, 1.2]	[1, 1.1]
9	$(x+1)\sin x$	[1, 5]	[1, 1.1]
10	$5x + x\lg(x)$	[0.2, 1]	[1.3, 1.4]
11	$(2x+3)\sin x$	[0.4, 1.2]	[0.5, 0.6]
12	$\frac{\cos x}{2x+5}$	[0.4, 1.2]	[1, 1.1]
13	$\frac{1}{1+x+x^2}$	[0, 4]	[2, 2.1]
14	$\frac{1+x}{2+x}$	[0.4, 0.8]	[1.5, 1.6]
15	$\sqrt{1+e^{-x}}$	[0.4, 1.2]	[0.5, 0.6]

**Задание 2.** Вычислить интеграл  $\int_a^b f(x)dx$  :

- с помощью встроенного оператора интегрирования;
- по формулам прямоугольников;
- по формуле Симпсона;

• методом неопределенных коэффициентов для численного интегрирования.

**Задание 3.** Вычислить интеграл  $\int_a^b f(x)dx$ : методом Мон-

те-Карло. Для этого необходимо:

• определить диапазон случайных чисел, например  $j = 0 \dots 1000$ ;

• определить с помощью функции `Random[Real, {0, x}]` равномерно распределенную случайную величину  $\eta_j$  на отрезке интегрирования  $[a, b]$ ;

• создать вектор  $F_j = f(\eta_j)$ ;

• с помощью функции `Mean[list]` вычислить интеграл.

**Задание 4.** Найдите первообразную аналитически заданной функции  $f(x)$ .

**Задание 5.** Вычислить значения первой и второй производных функции  $f(x)$  в точке  $X = c$ :

• с помощью операторов дифференцирования пакета `Mathematica`;

• методом неопределенных коэффициентов для численного дифференцирования. Определить функцию  $f(x)$  таблично, вычислив значения  $y_i = f(x_i)$  в точках  $x_i = c + hi$ ,  $i = 0, 1, \dots, 10$ ,  $h = 0.01$  на отрезке  $[c, d]$ .

**Задание 6.** Вычислить символьное значение первой и второй частных производных функции  $f(x)$  по переменной  $x$ .

## РЕШЕНИЕ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ И ИХ СИСТЕМ

**Цель:** изучить методы решения ОДУ и их систем.

### *Теоретические сведения*

$F(x, y, y', y'', \dots, y^{(n)}) = 0$  – обыкновенное дифференциальное уравнение (зависимость только от  $x$ ).

Задача Коши для обыкновенного дифференциального уравнения: определить частные решения, используя начальные условия.

Заменой переменной любое уравнение  $n$ -го порядка свести к системе  $n$  уравнений первого порядка:

$$\begin{cases} y' = y_1 \\ y'' = y_2 \\ \dots\dots\dots \\ y'_{n-1} = f(x, y, y_1, y_2, \dots, y_{n-1}) \end{cases}$$

Решение обыкновенных дифференциальных уравнений широко применяется на практике научно-технических расчетов. Хотя линейные ОДУ могут иметь решения в виде специальных функций, многие физические системы нелинейные и описываются нелинейными ОДУ, не имеющими аналитического решения. В этом случае приходится использовать численные методы решения.

### **Сущность численных методов**

На отрезке решения  $[x_0, x_n]$  выбирается некоторое множество точек, называемое сеткой:

$$x_0 < x_1 < x_2 < \dots < x_n.$$

В полученных точках вычисляются приближенные значения  $y_1, y_2, y_3, \dots, y_n$  решения задачи Коши. Задача сводится к построению

нию процедуры вычисления по известному приближенному значению решения  $y_k$  в точке  $x_k$  следующего приближения  $y_{k+1}$  в точке  $x_{k+1}$ . В результате начиная с  $x_0$ , последовательно вычисляются таблицу приближенных значений решения в узлах сетки.  $h = x_{k+1} - x_k$  – шаг сетки. Если  $h = \text{const}$ , то  $x_k = x_0 + kh$ ,  $k = 0, 1, 2, \dots$

Для оценки погрешности метода на одном шаге сетки разлагают точное решение в ряд Тейлора в окрестностях узла  $x_k$ :

$$y(x_{k+1}) = y(x_k) + hy'(x_k) + \frac{h^2}{2} y''(x_k) + \frac{h^3}{6} y'''(x_k) + \dots$$

Численный метод решения дифференциального уравнения имеет порядок  $p$ , если его расчетные формулы согласуются с разложением в ряд Тейлора до членов порядка  $h^p$ .

### Одношаговые методы

Одношаговые методы – это методы, в которых для нахождения следующей точки кривой  $y = f(x)$  требуется информация лишь об одном предыдущем шаге.

Простейшим из одношаговых методов является метод Эйлера:

$$y_{i+1} = y_i + h f(x_i, y_i), \quad i = 0, 1, \dots, n-1.$$

Метод Эйлера имеет невысокую точность (порядка  $h$ ).

Для достижения более высокой точности используют метод Рунге-Кутты четвертого порядка:

$$y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

где  $k_1 = f(x_i, y_i)$ ,  $k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{hk_1}{2}\right)$ ,  $k_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{hk_2}{2}\right)$ ,  
 $k_4 = f(x_i + h, y_i + hk_3)$ .

### Многошаговые методы

В многошаговых методах для отыскания следующей точки кривой  $y = f(x)$  требуется информация более чем об одной из предыдущих точек.

Пусть найдены значения  $y_{i-3}, y_{i-2}, y_{i-1}, y_i$  в четырех последовательных точках. При этом имеются также вычисленные заранее значения правой части уравнения  $\frac{dy}{dx} = f(x, y)$ :  $f_{i-3}, f_{i-2}, f_{i-1}, f_i$ . Тогда схему Адамса можно представить в виде:

$$y_{i+1} = y_i + h f_i + \frac{h}{2} \Delta f_i + \frac{5h}{12} \Delta^2 f_i + \frac{3h}{8} \Delta^3 f_i, \quad i = 3, \dots, n-1,$$

где конечные разности в точке  $x_i$  имеют вид:

$$\begin{aligned} \Delta f_i &= f_i - f_{i-1}, \\ \Delta^2 f_i &= f_i - 2f_{i-1} + f_{i-2}, \\ \Delta^3 f_i &= f_i - 3f_{i-1} + 3f_{i-2} - f_{i-3}. \end{aligned}$$

## Решение задачи Коши средствами Mathematica

Решение дифференциальных уравнений в общем виде:

`DSolve[eqn, y, x]`

Решает дифференциальное уравнение eqn относительно функции y, с независимой переменной x

`In[1]:= r = DSolve[Y' [x] == a Y [x] + Y[x], Y[x], x]`

`f[x_] = Y[x] /. r[[1]]`

`Out[1]= {{Y[x] -> e^{\frac{1}{2} (a - \sqrt{4+a^2}) x} C[1] + e^{\frac{1}{2} (a + \sqrt{4+a^2}) x} C[2]}}`

`Out[2]= e^{\frac{1}{2} (a - \sqrt{4+a^2}) x} C[1] + e^{\frac{1}{2} (a + \sqrt{4+a^2}) x} C[2]`

Решение задачи Коши для дифференциального уравнения:

`In[4]:=`

`Clear[r, Y, x]`

`r = DSolve[{Y' [x] == a Y [x] + Y[x], Y[0] == 1, Y' [0] == 0}, Y[x], x]`

`f[x_] = Y[x] /. r[[1]]`

`Out[5]= {{Y[x] -> \frac{a e^{\frac{1}{2} (a - \sqrt{4+a^2}) x} + \sqrt{4+a^2} e^{\frac{1}{2} (a - \sqrt{4+a^2}) x} - a e^{\frac{1}{2} (a + \sqrt{4+a^2}) x} + \sqrt{4+a^2} e^{\frac{1}{2} (a + \sqrt{4+a^2}) x}}{2 \sqrt{4+a^2}}}}`

`Out[6]= \frac{a e^{\frac{1}{2} (a - \sqrt{4+a^2}) x} + \sqrt{4+a^2} e^{\frac{1}{2} (a - \sqrt{4+a^2}) x} - a e^{\frac{1}{2} (a + \sqrt{4+a^2}) x} + \sqrt{4+a^2} e^{\frac{1}{2} (a + \sqrt{4+a^2}) x}}{2 \sqrt{4+a^2}}`



## Метод Рунге-Кутта для систем дифференциальных уравнений

$$\begin{cases} y_1' = f_1(x, y_1, y_2) \\ y_2' = f_2(x, y_1, y_2) \end{cases} \text{ Начальные условия: } \begin{matrix} y_1(x_0) = y_{1,0} \\ y_2(x_0) = y_{2,0} \end{matrix}$$

Тогда 
$$y_{1,i+1} = y_{1,i} + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$y_{2,i+1} = y_{2,i} + \frac{h}{6}(l_1 + 2l_2 + 2l_3 + l_4)$$

где  $k_1 = f_1(x_i, y_{1,i}, y_{2,i})$

$$l_1 = f_2(x_i, y_{1,i}, y_{2,i})$$

$$k_2 = f_1\left(x_i + \frac{h}{2}, y_{1,i} + \frac{hk_1}{2}, y_{2,i} + \frac{hl_1}{2}\right)$$

$$l_2 = f_2\left(x_i + \frac{h}{2}, y_{1,i} + \frac{hk_1}{2}, y_{2,i} + \frac{hl_1}{2}\right)$$

$$k_3 = f_1\left(x_i + \frac{h}{2}, y_{1,i} + \frac{hk_2}{2}, y_{2,i} + \frac{hl_2}{2}\right)$$

$$l_3 = f_2\left(x_i + \frac{h}{2}, y_{1,i} + \frac{hk_2}{2}, y_{2,i} + \frac{hl_2}{2}\right)$$

$$k_4 = f_1(x_i + h, y_{1,i} + hk_3, y_{2,i} + hl_3)$$

$$l_4 = f_2(x_i + h, y_{1,i} + hk_3, y_{2,i} + hl_3)$$

### Задания к лабораторной работе

**Задание 1.** Решить задачу Коши:  $\frac{d y}{d x} = f(x, y), y(0) = 1$  с

шагом  $h = 0.1$  на отрезке  $[0, 1]$ :

- методом Эйлера;
- методом Рунге-Кутта (коэффициенты  $k_i$  задать, как функции от  $x$  и  $y$ );
- методом Адамса;
- используя функцию `DSolve[]`.

### Варианты задания

№	$f(x, y)$	№	$f(x, y)$	№	$f(x, y)$
1	$x + y$	6	$2y - 2 \cos 2x$	11	$2y - 3e^{-x}$
2	$2x^2 + 2y$	7	$y - e^{\frac{x}{2}} + 2$	12	$\frac{y}{2} - e^{-x}$
3	$e^x - 3y$	8	$3y - 2 \sin x$	13	$y + \frac{\cos x}{3}$
4	$y - \sin x$	9	$e^{2x} - y$	14	$y - 4x + 5$
5	$\frac{y}{3} - x^2$	10	$2 \sin x + y$	15	$2x - \frac{y}{3} - e^x$

**Задание 2.** Построить графики решений, полученных методами Эйлера, Рунге-Кутта, Адамса и с помощью функции *DSolve*[].

Вычислить в точке  $x=1$  относительную погрешность для каждого метода.

**Задание 3.** Построить график численного решения ОДУ с использованием функции *DSolve*[].

---

```

In[2]:= solution = NDSolve[{y'[x] ==  $\frac{1}{2 y[x]}$ , y[.01] == .1},
    y, {x, .01, 1}];
Plot[y[x] /. solution, {x, .01, 1}];

```

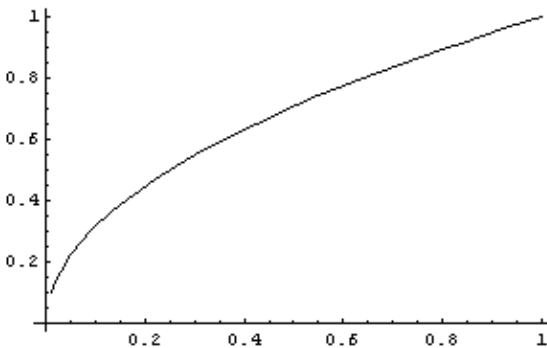


Рис. 10.1. Пример использования функции *NDSolve*[]

**Задание 4.** Решить задачу Коши для системы ОДУ при заданных начальных условиях на отрезке  $[0,2]$  с шагом  $h = 0.2$ . Решить с помощью функции  $DSolve[]$ . Построить графики функций  $u(t), v(t)$ .

Варианты задания

№	Система ОДУ	Начальные условия				№	Система ОДУ	Начальные условия			
		$u(0)$	$u'(0)$	$v(0)$	$v'(0)$			$u(0)$	$u'(0)$	$v(0)$	$v'(0)$
1	$\begin{cases} u'' = 2v + u \\ v'' = 4v - 2u \end{cases}$	1.5	1.5	1	1	9	$\begin{cases} u'' = \frac{1}{2} + v \\ v'' = 4 - u + t \end{cases}$	2	0	-1	1
2	$\begin{cases} u'' = -v + 3u \\ v'' = v - 2u \end{cases}$	-1	1	-1.5	3	$\begin{matrix} 1 \\ 0 \end{matrix}$	$\begin{cases} u'' = -v + t \\ v'' = v + 3u \end{cases}$	-1	2	-1.5	1
3	$\begin{cases} u'' = 2v - u \\ v'' = 4v + u \end{cases}$	1.5	1.5	1	1	$\begin{matrix} 1 \\ 1 \end{matrix}$	$\begin{cases} u'' = v - u - t \\ v'' = 2v + u \end{cases}$	1.5	1.5	-1	5
4	$\begin{cases} u'' = 5v \\ v'' = v + 2u + \end{cases}$	1	1.5	0	2	$\begin{matrix} 1 \\ 2 \end{matrix}$	$\begin{cases} u'' = 5v + t \\ v'' = 3v + u \end{cases}$	-1	1.5	0	2
5	$\begin{cases} u'' = 2v + u + \\ v'' = v + 2u - \end{cases}$	0.5	1.5	-1	2	$\begin{matrix} 1 \\ 3 \end{matrix}$	$\begin{cases} u'' = v + u \\ v'' = v + u - t \end{cases}$	-0.5	1	-1	0
6	$\begin{cases} u'' = 2v + u + \\ v'' = 4v \end{cases}$	0.5	2	1	2	$\begin{matrix} 1 \\ 4 \end{matrix}$	$\begin{cases} u'' = 2v - u \\ v'' = 4v + t \end{cases}$	0	-2	0	-1
7	$\begin{cases} u'' = -v + t \\ v'' = 5v - 7u \end{cases}$	5	5	-1	1	$\begin{matrix} 1 \\ 5 \end{matrix}$	$\begin{cases} u'' = v - 2t \\ v'' = v + 3u \end{cases}$	3	3	-1	1.5
8	$\begin{cases} u'' = v - 5u \\ v'' = 2v + u + \end{cases}$	1.5	1	3	1						

**Задание 5.** Решить задачу Коши для ОДУ 2-го порядка методом Рунге-Кутты. Определить близость полученного заданным методом решения к точному значению с помощью оценок:

$\beta_1 = \max |y_{Ti} - y_{Mi}|$  – линейная оценка,

$\beta_2 = \frac{\sqrt{\sum_{i=1}^n (y_{Ti} - y_{Mi})^2}}{\sqrt{\sum_{i=1}^n y_{Ti}^2}}$  – интегральная оценка,

где  $y_{Ti}$  – точное решение,  $y_{Mi}$  – полученное приближенное решение.

### Варианты задания

№	Дифференциальное уравнение	Начальные условия	Интервал интегрирования	Шаг	Точное решение
1	$y'' - 2y' + y = 0$	$y(2) = 1$ $y'(2) = 2$	[2; 4]	0.2	$y = (1 - 3x)e^{x-2}$
2	$y'' - 3y' + 2y - 2x + 3 = 0$	$y(0) = 1$ $y'(0) = 2$	[0; 2]	0.2	$y = e^x + x$
3	$y'' + y = 4e^x$	$y(0) = 4$ $y'(0) = -3$	[0; 1]	0.1	$y = 2 \cos(x) - 5 \sin(x) + 2e^x$
4	$x^2 y'' + xy' = 0$	$y(1) = 5$ $y'(1) = -1$	[1; 1.5]	0.05	$y = 5 - \ln(x)$
5	$y'' - 2y' = 2e^x$	$y(1) = -1$ $y'(1) = 0$	[1; 2]	0.1	$y = e^{2x-1} - 2e^x + e - 1$
6	$y'' + 4y = \cos(3x)$	$y(0) = 0.8$ $y'(0) = 2$	[0; 1]	0.1	$y = \cos(2x) + \sin(2x) - 0.2 \cos(3x)$
7	$y'' + 2y' + 2y = xe^x$	$y(0) = 0$ $y'(0) = 0$	[0; 1.5]	0.1	$y = e^{-x}(x - \sin(x))$
8	$(1 + x^2)y'' + (y')^2 + 1 = 0$	$y(0) = 1$ $y'(0) = 1$	[0; 0.5]	0.05	$y = 1 - x + 2 \ln(-1 + x)$
9	$y'' + 4y' + 4y = 0$	$y(0) = 1$ $y'(0) = -1$	[0; 1]	0.1	$y = (1 + x)e^{-2x}$

Окончание таблицы

10	$y'' - 3y' = e$	$y(0) = 2.2$ $y'(0) = 0.8$	[0; 0.2]	0.02	$y = 2 + 0.1(e^{3x} + e^{5x})$
11	$x^2 y'' - 2y = 0$	$y(1) = 0.83$ $y'(1) = 0.66$	[1; 2]	0.1	$y = 0.5x^2 + \frac{1}{3x}$
12	$y'' - 5y' + 6y = e^x$	$y(0) = 0$ $y'(0) = 0$	[0; 0.2]	0.02	$y = -e^{2x} + 0.5(e^{2x} + x^x)$
13	$y'' + y = 1 + e^x$	$y(0) = 2.5$ $y'(0) = 1.5$	[0; 1]	0.1	$y = \cos(x) + \sin(x) + \frac{e^x}{2} + 1$
14	$x^2 y'' + 2.5y'x - y = 0$	$y(1) = 2$ $y'(1) = 3.5$	[1; 2]	0.1	$y = 3\sqrt{x} - x^{-2}$
15	$y'' + y' = x^2 - x + 2$	$y(0) = 1$ $y'(0) = 0$	[0; 1]	0.1	$y = \cos(x) + \sin(x) + x^2 - x$

*Лабораторная работа №11*

**МЕТОД ДИФФЕРЕНЦИАЛЬНОЙ ПРОГОНКИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ КОШИ ДЛЯ ОДУ 2-ГО ПОРЯДКА**

**Цель:** Изучить метод дифференциальной прогонки.

*Теоретические сведения*

Дифференциальное уравнение и граничные условия записываются в виде

$$y''(x) + p(x)y'(x) - q(x)y(x) = f(x), \quad a \leq x \leq b \quad (11.1)$$

$$\alpha_0 y(a) + \beta_0 y'(a) = \gamma_0 \quad (11.2)$$

$$\alpha_1 y(b) + \beta_1 y'(b) = \gamma_1 \quad (11.3)$$

где  $p(x), q(x), f(x)$  – известные функции;  
 $\alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1$  – заданные постоянные.

**При условии**  $\beta_0 \neq 0$  вычислительная схема метода прогонки для задачи (11.1) – (11.3) может быть следующей:

1. Решаем задачу Коши относительно  $Z_1(x)$  и  $Z_2(x)$  для  $x \in [a; b]$ :

$$Z_1'(x) = -Z_1^2(x) - p(x)Z_1(x) + q(x), \quad Z_1(a) = -\frac{\alpha_0}{\beta_0} \quad (11.4)$$

$$Z_2'(x) = -Z_2(x)[Z_1(x) + p(x)] + f(x), \quad Z_2(a) = -\frac{\gamma_0}{\beta_0} \quad (11.5)$$

2. Используя значения  $Z_1(x)$ ,  $Z_2(x)$ , находим  $y(x)$  как решение задачи Коши

$$y'(x) = Z_1(x)y(x) + Z_2(x) \quad (11.6)$$

$$y(b) = \frac{\gamma_1 - \beta_1 Z_2(b)}{\alpha_1 - \beta_1 Z_1(b)}.$$

Решение задач (4) и (5) называют прямой прогонкой, а вычисление  $y(x)$  – решение задачи (6) – обратной прогонкой.

**При условии**  $\alpha_0 \neq 0$  схема определения решения имеет вид:

1. Решение задачи Коши

$$u_1'(x) = -u_1^2(x)q(x) + u_1(x)p(x) + 1, \quad a \leq x \leq b \quad (11.7)$$

$$u_1(a) = -\frac{\beta_0}{\alpha_0}$$

$$u_2'(x) = -u_1(x)[u_2(x)q(x) + f(x)], \quad a \leq x \leq b \quad (11.8)$$

$$u_2(a) = -\frac{\gamma_0}{\alpha_0}$$

2. После вычисления  $u_1(x)$  и  $u_2(x)$  находят  $y(x)$  как решение задачи Коши

$$y(x) = u_1(x)y'(x) + u_2(x) \quad \text{или} \quad y'(x) = \frac{y(x) - u_2(x)}{u_1(x)}$$

$$y(b) = \frac{\gamma_1 u_1(b) + \beta_1 u_2(b)}{\beta_1 + \alpha_1 u_1(b)}$$

**Примечание.** Схема численного решения задачи Коши вида

$$y'(x) = f(x, y), \quad y(x_0) = y_0$$

по методу Рунге-Кутты четвертого порядка с шагом  $h$  имеет вид

$$y_{i+1} = y_i + \Delta y_i \quad (i = 0, 1, 2, \dots), \quad \Delta y_i = \frac{1}{6} (k_1^i + 2k_2^i + 2k_3^i + k_4^i)$$

$$\begin{cases} k_1^i = hf(x_i, y_i) \\ k_2^i = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1^i}{2}\right) \\ k_3^i = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_2^i}{2}\right) \\ k_4^i = hf(x_i + h, y_i + k_3^i) \end{cases}$$

### *Задания к лабораторной работе*

**Задание 1.** Методом дифференциальной прогонки найти решение граничных задач в точках  $x_k = a + 0,1k$ ;  $k = 0, 1, \dots, 10$ . Для решения задач Коши использовать метод Рунге-Кутты четвертого порядка с шагом  $0,1$ :

1. 
$$y''(x) + 2y'(x) - \frac{4}{x}y(x) = 2; \quad 0,5 \leq x \leq 1; \quad y'(0,5) = 1,5;$$

$$y(1) + y'(1) = 4$$

2. 
$$y''(x) - \frac{6x}{3x^2 - 0,5} y'(x) - \frac{1}{x} y(x) = 0,5 - x^2; 0,5 \leq x \leq 1;$$

$$y'(0,5) = 0,25; 2y(1) + y'(1) = 3,5$$
3. 
$$y''(x) + \frac{1}{x} y'(x) - \frac{1}{x^2} y(x) = -\frac{2}{x^3}; 0,5 \leq x \leq 1; y(0,5) = -2 \ln 2;$$

$$y(1) = 0$$
4. 
$$y''(x) + 2xy'(x) - y(x) = 2(x^2 + 1) \cos x; 0 \leq x \leq 0,5;$$

$$y(0) = 0; y(0,5) = 0,5 \sin 0,5$$
5. 
$$y''(x) + y'(x) - 2y(x) = 3e^x; \quad 0 \leq x \leq 1; \quad y(0) = 0;$$

$$y(1) + y'(1) = 3e$$
6. 
$$y''(x) - \frac{1}{x+1} y'(x) - 2y(x) = -(x+1)^2; \quad 0 \leq x \leq 0,5;$$

$$y'(0) = 1; 2y(0,5) + 1,5y'(0,5) = 4,5$$
7. 
$$y''(x) + \frac{1}{x} y'(x) - 2y(x) = -2x^2; \quad 0,5 \leq x \leq 1; \quad y'(0,5) = 1;$$

$$y(1) + y'(1) = 5$$
8. 
$$y''(x) + x^2 y'(x) - \frac{2}{x^2} y(x) = 1; \quad 0,5 \leq x \leq 1;$$

$$y(0,5) - y'(0,5) = 6; y(1) = 1$$
9. 
$$y''(x) - y'(x) - 2y(x) = -2e^x; \quad 0 \leq x \leq 0,5; \quad y'(0) = 0;$$

$$y(0,5) + y'(0,5) = 3,297$$
10. 
$$y''(x) + (x+3)^2 y'(x) - \frac{2}{(x+3)^2} y(x) = -\frac{6}{(x+3)^2} + 3(x+3)^2 + 1;$$

$$0 \leq x \leq 1; y(0) = -\frac{1}{3}; y(1) + 4y'(1) = 15$$

Представить вычислительную схему метода прогонки в зависимости от коэффициентов  $\alpha_0$  и  $\beta_0$ . Представить таблицы вычисления функций  $Z_1(x)$ ,  $Z_2(x)$ ,  $y(x)$ . График искомой функции по точкам.



## МЕТОД СЕТОК РЕШЕНИЯ СМЕШАННОЙ ЗАДАЧИ ДЛЯ УРАВНЕНИЙ ГИПЕРБОЛИЧЕСКОГО ТИПА

**Цель:** Изучить метод сеток для решения дифференциальных уравнений.

### *Теоретические сведения*

Пусть требуется найти  $u(x, t)$  – решение уравнения

$$\frac{\partial^2 u(x, t)}{\partial t^2} = a^2(x, t) \frac{\partial^2 u(x, t)}{\partial x^2} + \varphi(x, t), \quad (12.1)$$
$$0 \leq x \leq 1, \quad 0 \leq t \leq T,$$

удовлетворяющее граничным условиям

$$u(0, t) = \gamma_0(t), \quad u(1, t) = \gamma_1(t), \quad 0 \leq t \leq T, \quad (12.2)$$

и начальным условиям

$$u(x, 0) = \alpha(x), \quad \frac{\partial u(x, 0)}{\partial t} = \beta(x), \quad 0 \leq x \leq 1. \quad (12.3)$$

В качестве сетки возьмем совокупность точек  $(x_m, t_n)$  с координатами  $x_m = m \cdot h$ ,  $t_n = n \cdot \tau$ ,  $m = 0, 1, \dots, M$ ,  $n = 0, 1, \dots, N$ ,  $h = 1/M$ ,  $\tau = 1/N$ . Пользуясь заменой вторых производных разностными отношениями, из (12.1) получим разностное уравнение

$$u_m^{n+1} = s_m^n u_{m+1}^n + 2(1 - s_m^n) u_m^n + s_m^n u_{m-1}^n - u_m^{n-1} + \tau \varphi(x_m, t_n) \quad (12.4)$$

$$m = 1, \dots, M - 1, \quad n = 1, \dots, N - 1, \quad s_m^n = \frac{\tau^2}{h^2} a^2(x_m, t_n),$$

аппроксимирующее дифференциальное уравнение (12.1) с погрешностью  $M_4(\tau^2 + Ah^2)/12$ , где  $M_4 = \max_{x,t} \left\{ \left| \frac{\partial^4 u}{\partial t^4} \right|, \left| \frac{\partial^4 u}{\partial x^4} \right| \right\}$ ,

$$A = \max_{x,t} a^2(x, t).$$

Из (12.2) имеем

$$u_0^n = \gamma_0(t_n), \quad u_M^n = \gamma_1(t_n), \quad n = 0, 1, \dots, N. \quad (5)$$

Следовательно, мы сможем найти  $u_m^{n+1}$  при  $n = 1, \dots, N - 1$  по (12.4) и (12.5), если будут известны  $u_m^0$  и  $u_m^1$  при  $m = 1, \dots, M - 1$ .

Для вычисления  $u_m^0$  и  $u_m^1$  следует использовать начальные условия (12.3). Это можно сделать несколькими способами.

**I способ.** Воспользовавшись представлением  $\frac{\partial u(x,0)}{\partial t} = \frac{u(x, t_1) - u(x, t_0)}{\tau} - \frac{\tau}{2} \frac{\partial^2 u(x, \tilde{t}_0)}{\partial t^2}$ ,  $t_0 \leq \tilde{t}_0 \leq t_1$ , из (12.3) получим

$$u_m^0 = \alpha(x_m), \quad u_m^1 = \tau\beta(x_m) + \alpha(x_m), \quad m = 1, \dots, M - 1. \quad (12.6)$$

Погрешность аппроксимации начальных условий (12.3) уравнениями (12.6) оценивается величиной  $\tau M_t^{(2)} = \max_{x,t} \left| \frac{\partial^2 u}{\partial t^2} \right|$ .

**II способ.** По формуле Тейлора имеем

$$u(x, t_1) = u(x, t_0) + \frac{\tau}{1!} \frac{\partial u(x, t_0)}{\partial t} + \frac{\tau^2}{2!} \frac{\partial^2 u(x, t_0)}{\partial t^2} + \frac{\tau^3}{3!} \frac{\partial^3 u(x, \tilde{t}_1)}{\partial t^3},$$

$$t_0 \leq \tilde{t}_1 \leq t_1.$$

Отсюда

$$\frac{\partial u(x, t_0)}{\partial t} = \frac{u(x, t_1) - u(x, t_0)}{\tau} - \frac{\tau}{2} \frac{\partial^2 u(x, t_0)}{\partial t^2} - \frac{\tau^2}{6} \frac{\partial^3 u(x, \tilde{t}_1)}{\partial t^3} \quad (12.7)$$

Если  $\alpha(x)$  имеет конечную вторую производную, то из (12.1) и (12.3) получим

$$\frac{\partial^2 u(x, t_0)}{\partial t^2} = a^2(x, t_0) \frac{\partial^2 u(x, t_0)}{\partial x^2} + \varphi(x, t_0) = a^2(x, t_0) \alpha''(x) + \varphi(x, t_0)$$

Подставив это значение в (12.7), найдем

$$\begin{aligned} u_m^0 &= \alpha(x_m), \\ u_m^1 &= \alpha(x_m) + \tau \beta(x_m) + \frac{\tau^2}{2} [a^2(x_m, 0) \alpha''(x_m) + \varphi(x_m, 0)], \quad (12.8) \\ m &= 1, \dots, M - 1. \end{aligned}$$

Погрешность аппроксимации начальных условий уравнениями (12.8) оценивается величиной  $\tau^2 M_t^{(3)} / 6$ , где  $M_t^{(3)} = \max_{x,t} \left| \frac{\partial^3 u}{\partial t^3} \right|$ .

Разностные схемы (12.4) – (12.6) и (12.4), (12.5), (12.8) будут устойчивы при выполнении условия  $\frac{\tau^2}{h^2} \leq \frac{1}{A}$ .

#### Алгоритм решения задачи:

1. Записать разностное уравнение вида (12.4).
2. Записать аппроксимацию граничных условий по формулам (12.5).
3. Аппроксимировать начальные условия, используя формулы (12.6) или (12.8).

4. Вычислить значения функции  $u(x, t)$  в остальных узлах сетки  $(x_m, t_n)$  и представить в виде таблицы

$n$	$u_0^n$	$u_1^n$	...	$u_{M-1}^n$	$u_M^n$
0	$\gamma_0(t_0)$	$\alpha(x_1)$	...	$\alpha(x_{M-1})$	$\gamma_1(t_0)$
1	$\gamma_0(t_1)$	$u_1^1$	...	$u_{M-1}^1$	$\gamma_1(t_1)$
...	...	...	...	...	...
$N$	$\gamma_0(t_N)$	$u_1^N$	...	$u_{M-1}^N$	$\gamma_1(t_N)$

5. Построить по точкам поверхность для  $u(x, t)$ .

### *Задания к лабораторной работе*

**Задание 1.** Найти решения смешанных задач для уравнений гиперболического типа на сетке  $(x_m, t_n)$ , где  $x_m = 0,1m$ ,  $t_n = 0,1n$ ,  $m, n = 0,1, \dots, 10$ :

1	$\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2} + \frac{2(\alpha^2 - 1)}{(x + \alpha t + 1)^3}, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1,$ $u(0, t) = \frac{1}{\alpha t + 1}, \quad u(1, t) = \frac{1}{\alpha t + 2}, \quad 0 \leq t \leq 1$ $u(x, 0) = \frac{1}{1 + x}, \quad \frac{\partial u(x, 0)}{\partial t} = -\frac{\alpha}{(1 + x)^2}, \quad 0 \leq x \leq 1,$ $\alpha = 0,5 + 0,1k, \quad k = 0,1, \dots, 10$
2	$\frac{\partial^2 u(x, t)}{\partial t^2} = \frac{\partial^2 u(x, t)}{\partial x^2}, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1,$ $u(0, t) = \frac{1}{t + \alpha}, \quad u(1, t) = \frac{1}{t + \alpha + 1}, \quad 0 \leq t \leq 1$ $u(x, 0) = \frac{1}{\alpha + x}, \quad \frac{\partial u(x, 0)}{\partial t} = -\frac{1}{(\alpha + x)^2}, \quad 0 \leq x \leq 1,$ $\alpha = 0,5 + 0,1k, \quad k = 1, \dots, 10$

3	$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2} + \frac{2\alpha}{(\alpha(x+t)+2)^2}, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1,$ $u(0,t) = 0, \quad u(1,t) = \frac{1}{\alpha(t+1)+2}, \quad 0 \leq t \leq 1$ $u(x,0) = \frac{x}{\alpha x + 2}, \quad \frac{\partial u(x,0)}{\partial t} = -\frac{\alpha x}{(\alpha x + 2)^2}, \quad 0 \leq x \leq 1,$ $\alpha = 0,2k, \quad k = 1, \dots, 10$
4	$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2}, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1,$ $u(0,t) = e^{\alpha t}, \quad u(1,t) = e^{\alpha(t+1)}, \quad 0 \leq t \leq 1$ $u(x,0) = e^{\alpha x}, \quad \frac{\partial u(x,0)}{\partial t} = \alpha e^{\alpha x}, \quad 0 \leq x \leq 1,$ $\alpha = 0,5k, \quad k = -5, -4, -3, \dots, 5$
5	$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2}, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq 1,$ $u(0,t) = 0, \quad u(1,t) = 0, \quad 0 \leq t \leq 1$ $u(x,0) = 0, \quad \frac{\partial u(x,0)}{\partial t} = \pi \sin \pi x, \quad 0 \leq x \leq 1,$ $\alpha = 0,5k, \quad k = -5, -4, -3, \dots, 5$

## МЕТОД КОНЕЧНЫХ ЭЛЕМЕНТОВ.

**Цель:** изучить метод конечных элементов.

### *Теоретические сведения*

*Схематизация свойств упругой конструкции.*

*Матрица жесткости. Матрица масс.*

В МКЭ схематизация свойств упругих конструкций осуществляется на основе аппроксимирующих функций. В частности, если положение упругого балочного элемента (рис.13.1) охарактеризовать четырьмя узловыми перемещениями  $z_1, z_2, z_3, z_4$ , то можно сделать следующее предположение о законе изменения прогиба элемента балки:

$$y(x, t) = \alpha_1(t) + \alpha_2(t) \cdot x + \alpha_3(t) \cdot x^2 + \alpha_4(t) \cdot x^3, \quad (13.1)$$

где  $\alpha_i$  – произвольные переменные, подлежащие определению.



Рис. 13.1. Нумерация узловых перемещений

Параметры  $\alpha_i$  можно выразить через узловые перемещения

$$z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix}$$

подставляя в (13.1) граничные условия на концах конечного элемента можно получить выражения для нахождения коэффициентов  $\alpha_i(t)$ :

$$z_1 = y(0, t) = \alpha_1(t), \quad z_2 = \left. \frac{\partial y}{\partial x} \right|_{x=0} = \alpha_2(t),$$

$$z_3 = y(l, t) = \alpha_1(t) + \alpha_2(t) \cdot l + \alpha_3(t) \cdot l^2 + \alpha_4(t) \cdot l^3,$$

$$z_4 = \left. \frac{\partial y}{\partial x} \right|_{x=l} = \alpha_2(t) + 2 \cdot \alpha_3(t) \cdot l + 3 \cdot \alpha_4(t) \cdot l^2.$$

Коэффициенты  $\alpha_i(t)$  позволяют записать закон изменения прогиба оси элемента балки в виде

$$y(x, t) = \sum_{j=1}^4 z_j(t) \mathcal{E}_j(x), \quad (13.2)$$

где аппроксимирующие функции  $\mathcal{E}_j(x)$  зависят от координат точки и имеют вид:

$$\begin{aligned} \mathcal{E}_1(x) &= 1 - 3 \frac{x^2}{l^2} + 2 \frac{x^3}{l^3}, & \mathcal{E}_2(x) &= x - 2 \frac{x^2}{l} + \frac{x^3}{l^2}, \\ \mathcal{E}_3(x) &= 3 \frac{x^2}{l^2} - 2 \frac{x^3}{l^3}, & \mathcal{E}_4(x) &= -\frac{x^2}{l} + \frac{x^3}{l^2}. \end{aligned}$$

Уравнение малых колебаний однородного упругого элемента балки можно получить в форме уравнений Лагранжа II рода. Для этого предварительно вычислим потенциальную  $\Pi$  и кинетическую  $T$  энергии элемента длиной  $l$ , погонной массой  $m$  и жесткостью на изгиб  $EJ$ :

$$\begin{aligned} 2\Pi &= EJ \int_0^l [y''(x, t)]^2 dx, \\ 2T &= m \int_0^l \dot{y}^2(x, t) dx. \end{aligned}$$

Подставляя сюда (1.2), получим:

$$2\Pi = \sum_{i=1}^4 \sum_{k=1}^4 c_{ik} z_i z_k, \quad 2T = \sum_{i=1}^4 \sum_{k=1}^4 m_{ik} \dot{z}_i \dot{z}_k, \quad (13.3)$$

$$\text{где } c_{ik} = m \int_0^l \mathcal{E}''_i(x) \cdot \mathcal{E}''_k(x) dx,$$

$$m_{ik} = m \int_0^l \mathcal{E}_i(x) \cdot \mathcal{E}_k(x) dx.$$

Уравнение Лагранжа II рода, соответствующее варьированию координаты  $z_i$ , имеет вид

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{z}_i} \right) - \left( \frac{\partial T}{\partial z_i} \right) + \frac{\partial \Pi}{\partial z_i} = Q_i,$$

где  $Q_i$  – обобщенные внешние силы.

Подставляя в уравнение движения выражения (13.3) и принимая  $Q_i = 0$ , получаем уравнения свободных колебаний однородного упругого элемента балки

$$\sum_{k=1}^4 m_{ik} \ddot{z}_k + \sum_{k=1}^4 c_{ik} z_k = 0 \quad (i=1,2,3,4). \quad (4)$$

В матричной форме уравнение (13.4) принимает вид

$$M \ddot{z} + C z = 0,$$

где  $M$  – матрица масс элемента;

$$M_u = m \begin{bmatrix} \frac{13}{35}l & \frac{11}{210}l^2 & \frac{9}{70}l & -\frac{13}{420}l^2 \\ \frac{11}{210}l^2 & \frac{1}{105}l^3 & \frac{13}{420}l^2 & -\frac{1}{140}l^3 \\ \frac{9}{70}l & \frac{13}{420}l^2 & \frac{13}{35}l & -\frac{11}{210}l^2 \\ -\frac{13}{420}l^2 & -\frac{1}{140}l^3 & -\frac{11}{210}l^2 & \frac{1}{105}l^3 \end{bmatrix},$$



$C$  – матрица жесткости элемента;

$$C_u = EJ \begin{bmatrix} \frac{12}{l^3} & \frac{6}{l^2} & -\frac{12}{l^3} & \frac{6}{l^2} \\ \frac{6}{l^2} & \frac{4}{l} & -\frac{6}{l^2} & \frac{2}{l} \\ -\frac{12}{l^3} & -\frac{6}{l^2} & \frac{12}{l^3} & -\frac{6}{l^2} \\ \frac{6}{l^2} & \frac{2}{l} & -\frac{6}{l^2} & \frac{4}{l} \end{bmatrix}.$$

Индекс  $u$  указывает на связь данной величины с изгибными деформациями.

Аналогичным образом можно получить свободные колебания упругого однородного элемента балки, работающего на растяжение-сжатие.

Деформированное состояние элемента от растяжения-сжатия в рассматриваемом случае можно охарактеризовать узловыми перемещениями  $z_5$ ,  $z_6$  в направлении оси  $x$  (рис.13.2.). Перемещение произвольного сечения элемента в этом направлении аппроксимируется выражением

$$u(x,t) = z_5(t) \vartheta_5(x) + z_6(t) \vartheta_6(x), \text{ где } \vartheta_5(x) = 1 - \frac{x}{l}, \vartheta_6(x) = \frac{x}{l}.$$



Рис.13.2. Нумерация узловых перемещений

Соответствующая матрица масс и матрица жесткости получаются в следующем виде:

$$M_p = m \begin{bmatrix} \frac{l}{3} & \frac{l}{6} \\ \frac{l}{6} & \frac{l}{3} \end{bmatrix}, C_p = \frac{EF}{l} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Здесь  $EF$  – жесткость элемента на растяжение-сжатие. Индекс  $p$  указывает на связь данной величины с деформациями растяжения-сжатия элемента.

Если предположить, что осевые усилия не влияют на изгиб элемента балки, то матрица жесткости и матрица масс для конечного элемента балки при совместном учете деформаций изгиба и растяжения-сжатия определяется путем суммирования его деформации изгиба  $\{z_1, z_2, z_3, z_4\}$  и деформации растяжения-сжатия  $\{z_5, z_6\}$ .



Рис.13.3. Нумерация узловых перемещений

В связи с этим можно записать

$$C = \begin{bmatrix} C_u & 0 \\ 0 & C_p \end{bmatrix}; M = \begin{bmatrix} M_u & 0 \\ 0 & M_p \end{bmatrix}. \quad (13.5)$$

Принятая при выводе зависимостей (13.5) нумерация узловых перемещений и узловых усилий, приводит к некоторым неудобствам при построении матрицы жесткости и матрицы масс для всей дискретной модели. Более удобной оказывается нумерация узловых перемещений и узловых усилий, приведенная на рис.13.4.



Рис.13.3. Нумерация узловых перемещений

В этом случае закон изменения компонент перемещений для произвольной точки КЭ имеет вид

$$y(x, t) = z_2(t)\mathcal{E}_2(x) + z_3(t)\mathcal{E}_3(x) +$$

$$+ z_5(t)\vartheta_5(x) + z_6(t)\vartheta_6(x), \quad (13.6)$$

$$u(x, t) = z_1(t)\vartheta_1(x) + z_4(t)\vartheta_4(x), \quad (13.7)$$

где  $\vartheta_1(x) = 1 - \frac{x}{l}$ ,  $\vartheta_2(x) = 1 - 3\frac{x^2}{l^2} + 2\frac{x^3}{l^3}$ ,

$$\vartheta_3(x) = x - 2\frac{x^2}{l} + \frac{x^3}{l^2}, \quad \vartheta_4(x) = \frac{x}{l},$$

$$\vartheta_5(x) = 3\frac{x^2}{l^2} - 2\frac{x^3}{l^3}, \quad \vartheta_6(x) = -\frac{x^2}{l} + \frac{x^3}{l^2}.$$

Если в качестве КЭ использовать однородную балку с погонной массой  $m$ , то матрица масс такого элемента будет иметь вид

$$M = m \cdot \begin{bmatrix} \frac{l}{3} & 0 & 0 & \frac{l}{6} & 0 & 0 \\ \frac{13}{35}l & \frac{11}{210}l^2 & 0 & \frac{9}{70}l & -\frac{13}{420}l^2 & 0 \\ \frac{l^3}{105} & 0 & \frac{13}{420}l^2 & -\frac{l^3}{140} & 0 & 0 \\ \frac{l}{3} & 0 & 0 & 0 & 0 & 0 \\ \text{симметрично} & \frac{13}{5}l & -\frac{11}{210}l^2 & \frac{l^3}{105} & 0 & 0 \end{bmatrix} \quad (8)$$

Матрица жесткости переписется в следующем виде:

$$C = \begin{bmatrix} \frac{EF}{l} & 0 & 0 & -\frac{EF}{l} & 0 & 0 \\ & \frac{12EJ}{l^3} & \frac{6EJ}{l^2} & 0 & -\frac{12EJ}{l^3} & \frac{6EJ}{l^2} \\ & & \frac{4EJ}{l} & 0 & -\frac{6EJ}{l^2} & \frac{2EJ}{l} \\ & & & \frac{EF}{l} & 0 & 0 \\ & & & & \frac{12EJ}{l^3} & -\frac{6EJ}{l^2} \\ & & & & & \frac{4EJ}{l} \end{bmatrix} \quad (13.9)$$

симметрично

## 1.2. Матрицы жесткости и масс в общей системе координат

Матрицы, полученные выше, справедливы для элементов, ориентированных в локальных (собственных) системах координат  $x'y'$ . Для записи уравнений движения системы в целом целесообразно использование общей системы координат  $xu$  (рис.13.5).

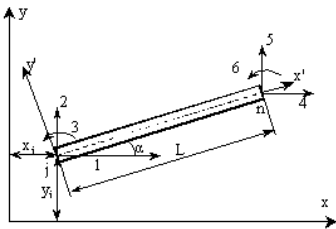


Рис. 13.5. Общая  $xu$  и локальная  $x'y'$  системы координат

Узловые силы и перемещения преобразуются из общей в локальную систему координат с помощью матрицы  $N$

$$z' = N z, \quad )$$

где  $N = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}$ , а  $\lambda$  – представляет собой квадратную.

матрицу направляющих косинусов, т.е  $\underline{\lambda} = \begin{vmatrix} \lambda_{x'x} & \lambda_{x'y} & \lambda_{x'z} \\ \lambda_{y'x} & \lambda_{y'y} & \lambda_{y'z} \\ \lambda_{z'x} & \lambda_{z'y} & \lambda_{z'z} \end{vmatrix}$

Если известны координаты  $x_j, y_j$  и  $x_n, y_n$  концов балочного элемента в общей системе координат, то выражения для направляющих косинусов в рассматриваемом случае имеют вид

$$\begin{aligned} \lambda_{x'x} &= \frac{x_n - x_j}{l}, \quad \lambda_{x'y} = \frac{y_n - y_j}{l}, \quad \lambda_{x'z} = 0; \\ \lambda_{y'x} &= -\frac{y_n - y_j}{l}, \quad \lambda_{y'y} = \frac{x_n - x_j}{l}, \quad \lambda_{y'z} = 0; \\ \lambda_{z'x} &= 0, \quad \lambda_{z'y} = 0, \quad \lambda_{z'z} = 1; \end{aligned}$$

где  $l = \sqrt{(x_n - x_j)^2 + (y_n - y_j)^2}$ .

Вводя обозначения  $\lambda_1 = \frac{x_n - x_j}{l}$ ,  $\lambda_2 = \frac{y_n - y_j}{l}$ , получаем для матрицы  $\underline{\lambda}$  следующее выражение:

$$\underline{\lambda} = \begin{vmatrix} \lambda_1 & \lambda_2 & 0 \\ -\lambda_2 & \lambda_1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Обозначим полученные выше в локальной системе координат матрицы жесткости и масс, как  $C'$  и  $M'$ . В этом случае выражения для потенциальной и кинетической энергии элемента имеют вид

$$2\Pi' = \dot{z}'^T C' z', \quad 2T' = \dot{z}'^T M' \dot{z}'.$$

Учитывая (13.4), получим

$$2\Pi' = z^T N^T C' N z, \quad 2T' = \dot{z}^T N^T M' N \dot{z}.$$

Так как потенциальная и кинетическая энергии не зависят от выбора системы координат, то

$$2\Pi = z^T C z, \quad 2T = \dot{z}^T M \dot{z},$$

где  $C$ ,  $M$  – соответственно матрица жесткости и матрица масс конечного элемента в общей системе координат.

Приравнивая соответствующие выражения для энергий, получим

$$C = NTC'N, \quad M = NTM'N.$$

#### *Формирование уравнений движения системы*

Располагая матрицей жесткости и матрицей масс для отдельных элементов конструкции можно получить соответствующие матрицы конструкции в целом. Воспользуемся для этого матрицей индексов.

Для построения матрицы индексов для каждого конечного элемента вводится вектор  $\{n\}$ , содержащий номера узловых перемещений в общей системе нумерации неизвестных. В матрице индексов эти вектора являются столбцами:

$$[n] = [\{n\}_1 \{n\}_2 \dots \{n\}_K].$$

Матрица индексов содержит число столбцов, равное числу конечных элементов, и число строк, равное числу степеней свободы элемента.

Для построения глобальных матриц масс и жесткости следуют алгоритму:

обнуление всех элементов матрицы  $M$  и  $C$ .

$$C_{ij} = 0; M_{ij} = 0; i = 1, \dots, K; j = 1, \dots, K.$$

Суммирование частных матриц конечных элементов по формуле (матрица масс аналогично)

$$C_{n_{is}, n_{js}} + = C_{ij}^s,$$

где  $n_{is}, n_{js}$  – индексы (составляющие) матрицы индексов  $[n]$ ;

$i, j = 1 \dots L_s$  – количество степеней свободы  $s$ -того конечно-го элемента);

$s = 1 \dots K$  – номер конечного элемента.

С учетом граничных условий для нулевых значений перемещений соответствующие строки и столбцы матрицы жесткости и матрицы масс должны быть вычеркнуты.

Полученные матрицы позволяют записать уравнения движения модели в целом:

$$M \ddot{z} + C z = 0, \quad (13.11)$$

где  $M, C$  – глобальные матрицы масс и жёсткости соответственно.

### *Частоты и формы свободных колебаний*

Для линейных систем при вычислении собственных частот и форм колебаний без учета трения уравнение движения принимает следующий вид:

$$[M] \cdot \{\ddot{x}\} + [C] \cdot \{x\} = 0. \quad (13.12)$$

Будем искать решение в виде  $\{x\} = \{A\} \sin(\omega t)$ , подставив его в уравнение (13.12), получим

$$-\omega^2 [M] \cdot \{A\} + [C] \cdot \{A\} = 0$$

$$[-\omega^2 M + C]\{A\} = 0$$

Не тривиальное решение такой системы существует, если определитель матрицы  $[-\omega^2 M + C]$  равен нулю. Равенство нулю определителя дает нам частотное уравнение, решение которого эквивалентно решению задачи на нахождение собственных значений матрицы  $[[M]^{-1} \cdot [C]]$ .

Собственные значения матрицы  $[[M]^{-1} \cdot [C]]$  являются квадратами собственных частот системы. Соответствующий собственный вектор матрицы является собственной формой колебаний системы.

### *Порядок решения задачи методом конечных элементов*

Разбивка расчетной схемы на конечные элементы.

Нумерация перемещений и узлов.

Формирование частных матриц конечных элементов.

Определение глобальной системы координат и построение частных матриц направляющих косинусов.

Построение глобальных матриц масс и жесткости.

Учёт граничных условий

Определение собственных чисел и векторов для  $M^{-1}C$ .

### ***Задания к лабораторной работе***

**Задание 1.** Рассчитать частоты и формы свободных колебаний методом конечного элемента для расчетной схемы, показанной на рис.13.6. (рис.13, *a* – для четных вариантов; рис. 13,*б* – для нечетных вариантов).

При выполнении расчетов полагать, что материал конструкции – сталь, а стержни имеют круглое поперечное сечение диаметра  $d$ . Размеры элементов (в метрах) приведены в таблице :

Для схемы на рис.13.6, <i>a</i>			Для схемы на рис .13.6, <i>б</i>				
$l_1$	$l_2$	$d$	$l_0$	$l_1$	$l_2$	$l_3$	$d$



1.2	0.6	0.04	2.0	1.5	0.8	0.4	0.05
-----	-----	------	-----	-----	-----	-----	------

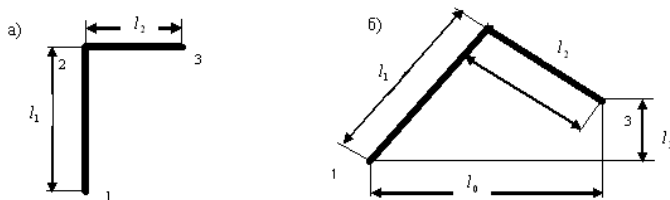


Рис. 13.6

Варианты граничных условий:

№ Варианта	Перемещения в направлении координат ( $XU$ – угол поворота)								
	Узел №1			Узел №2			Узел №3		
	$X$	$U$	$XU$	$X$	$U$	$XU$	$X$	$U$	$XU$
1	+	+	+	-	-	-	-	-	-
2	+	+	+	-	-	-	-	+	-
3	+	+	-	-	-	-	+	+	-
4	-	-	-	-	-	-	+	+	+
5	+	-	-	-	-	-	+	+	+
6	-	+	-	-	-	-	+	+	+
7	+	+	+	-	-	-	+	-	-
8	+	+	+	-	+	-	-	-	-
9	+	+	+	-	+	-	-	-	-
10	+	+	-	-	+	-	-	+	-
11	+	-	-	-	+	-	+	+	-
12	+	-	-	+	+	-	-	+	-
13	-	+	-	+	+	-	-	-	-
14	-	-	-	+	+	-	-	+	-
15	-	+	-	-	+	-	+	+	+

Знаку «+» соответствует наличие указанной связи, а знаку «-» – ее отсутствие.

Порядок выполнения работы.

1. Для расчетной схемы своего варианта задания выполнить разбивку на конечные элементы.
2. Задать нумерацию неизвестных перемещений.
3. Выбрать глобальную систему координат.
4. Для каждого конечного элемента вычислить матрицы жесткости, масс и матрицу направляющих косинусов.
5. Преобразовать матрицы жесткости и масс к глобальной системе координат.
6. Составить матрицу индексов.
7. Получить общую матрицу жесткости и матрицу масс в глобальной системе координат.
8. Учитывая граничные условия своего варианта задания исключить из общих матриц строки и столбцы, соответствующие нулевым перемещениям. (Для преобразования матриц жесткости и масс возможно последовательное применение функции `submatrix(A,ir,jr,ic,jc)` Matcad, которая возвращает подматрицу, состоящую из элементов, содержащихся в строках от *ir* до *jr* и в столбцах от *ic* до *jc*.)
9. Вычислить собственные частоты и построить соответствующие формы колебаний. (Для вычислений вектора собственных частот и векторов матрицы *A* возможно применение функций `eigenvals(A)` и `eigenvecs(A)` Mathcad.)

Отчет по работе должен содержать:

1. Расчетную схему с указанием неизвестных переменных и граничных условий.
2. Выражения для инерционной матрицы и матрицы жесткости в символьной форме.
3. Выражения для матриц направляющих косинусов для всех конечных элементов.
4. Матрицу индексов.
5. Найденные значения частот свободных колебаний.
6. Графическое изображение первых двух форм свободных колебаний.

## ЛИТЕРАТУРА

1. Амосов, А.А. Вычислительные методы для инженеров. / А.А. Амосов, Ю.А. Дубинский, Н.В. Копченова. – М.: Высшая школа, 1994.
2. Самарский, А.А., Численные методы. / А.А. Самарский, А.В. Гулин – М.: Наука, 1989.
3. Сборник по методам вычислений / под ред. Монастырного.
4. Зенкевич, О. Метод конечных элементов в технике. / О. Зенкевич – М.: Мир, 1975. – 541 с.
5. Бате, К. Численные методы анализа и метод конечных элементов: Пер. с англ. / К. Бате, Е. Вилсон – М., Мир, 1982.
6. Строительная механика. Динамика и устойчивость сооружений / А.Ф. Смирнов [и др.], – М.:Стройиздат, 1984. – 416 с.
7. Кандидов, В.П. Метод конечных элементов в задачах динамики / В.П. Кандидов, С.С. Чесноков, В.А. Выслоух. – М.: изд-во Моск. ун-та, 1980. – 165 с.
8. Образцов, И.Ф. Метод конечных элементов в задачах строительной механики летательных аппаратов: учебное пособие для студентов авиационных специальностей вузов. / И.Ф. Образцов, Л.М. Савельев, Х.С. Хазанов. – М.: Высшая школа, 1985. – 392 с.

## Содержание

Лабораторная работа №7. Интерполяция и предсказание. . .	3
Лабораторная работа № 8.Аппроксимация функций. Метод наименьших квадратов. . . . .	10
Лабораторная работа № 9. Численное интегрирование и дифференцирование. . . . .	19
Лабораторная работа № 10. Решение обыкновенных дифференциальных уравнений и их систем. . . . .	29
Лабораторная работа №11. Метод дифференциальной прогонки для решения задачи Коши для ОДУ 2-го порядка. . . . .	36
Лабораторная работа № 12. Метод сеток решения смешанной задачи для уравнений гиперболического типа. . . .	40
Лабораторная работа № 13. Метод конечных элементов. . . .	45
Литература. . . . .	58

Учебное издание

## ЧИСЛЕННЫЕ МЕТОДЫ

Лабораторный практикум  
для студентов специальности  
«Компьютерная мехатроника» 1-55 01 03

В 2 частях

Часть 2

### МАТЕМАТИЧЕСКАЯ ОБРАБОТКА РЕЗУЛЬТАТОВ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ

С о с т а в и т е л и :  
КУЛИК Иван Семенович  
Довга Юлия Алексеевна

Редактор Л.Н. Шалаева  
Компьютерная верстка С.В. Бондаренко

---

Подписано в печать 29.12.2010.

Формат 60×84<sup>1</sup>/<sub>16</sub>. Бумага офсетная.

Отпечатано на ризографе. Гарнитура Таймс.

Усл. печ. л. 2,2. Уч.-изд. л. 1,7. Тираж 50. Заказ 717.

---

Издатель и полиграфическое исполнение:  
Белорусский национальный технический университет.

ЛИ № 02330/0494349 от 16.03.2009.

Проспект Независимости, 65. 220013, Минск.