

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**  
**им. М. В. Ломоносова**

Факультет фундаментальной механики и математики

Численные методы восстановления поверхностной функции

Студент 431 группы \_\_\_\_\_ А. К. Новиков

Преподаватель \_\_\_\_\_ А. А. Корнев

Москва 2020

# Содержание

<b>1. Введение</b>	<b>3</b>
<b>2. Постановка задачи</b>	<b>3</b>
<b>3. Дискретное преобразование Фурье</b>	<b>6</b>
3.1. Описание метода . . . . .	6
3.2. Программная реализация метода . . . . .	7
<b>4. Кубические интерполяционные сплайны</b>	<b>7</b>
4.1. Описание метода . . . . .	7
4.2. Программная реализация метода . . . . .	9

# 1. Введение

Имеется двумерная область. На данной области задана сетка узлов, в каждом из которых находится значение некоторой неизвестной функции. Часть значений потеряна. Необходимо разработать численный метод, при помощи которого можно будет достаточно точно восстановить значения этой функции в потерянных узлах.

## 2. Постановка задачи

Пусть область представляет из себя прямоугольник на плоскости, а сетка - набор точек разбиения данного прямоугольника на  $n$  равных частей по горизонтали и  $m$  равных частей по вертикали, как представлено на следующем рисунке:

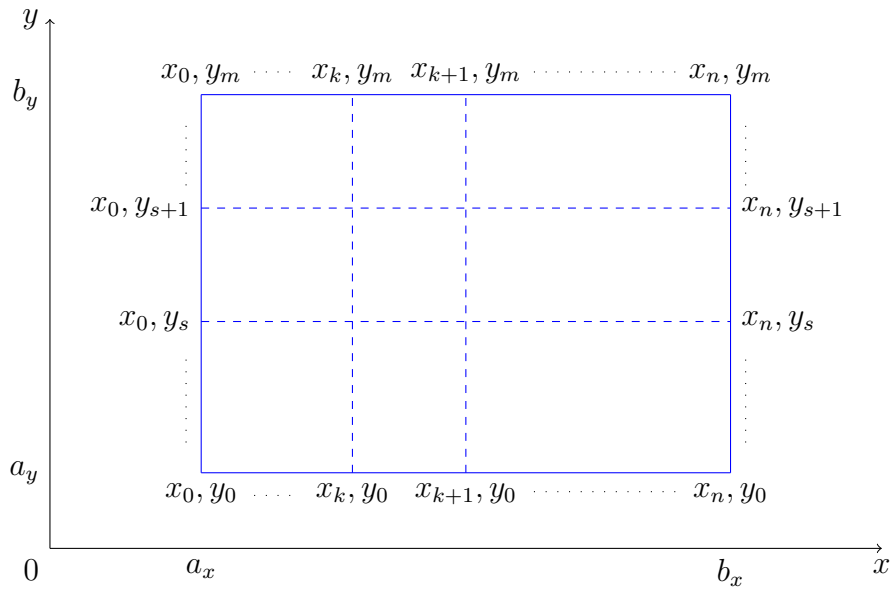


Рис. 1.

$$\Omega = \{(x, y) \in \mathbb{R}^2 | a_x \leq x \leq b_x, a_y \leq y \leq b_y\} \quad (1)$$

Итак, пусть имеются значения некоторой неизвестной функции  $f(x, y)$  в узлах  $(x_k, y_s)$ , где  $0 \leq k \in \mathbb{Z} \leq n$ ,  $0 \leq s \in \mathbb{Z} \leq m$ ,  $n, m \in \mathbb{N}$  двумерной области  $\Omega \subset \mathbb{R}^2$ , представляющей из себя прямоугольник (1), изображенный на Рис. 1, причём часть значения потеряна.

Для того, чтобы восстановить потерянные значения, нам необходимо выбрать направление движения и применить один из методов восстановления значений функции на отрезке. Удобней всего выбрать два направления: **по оси абсцисс** при фиксированном значении координаты  $y$  слева направо в сторону увеличения координаты  $x$  и **по оси ординат** при фиксированном значении координаты  $x$  снизу вверх в сторону увеличения координаты  $y$ .

Возьмём в качестве функции, значения которой необходимо будет восстанавливать, следующую функцию:

$$f(x, y) = (x - a_x)(x - b_x)(\cos(x) - \ln(y^2)) \quad (2)$$

Изобразим поверхность, которую задаёт выбранная функция (2) при  $a_x = 2$ ,  $b_x = 6$ ,  $a_y = 1$  и  $b_y = 4$ .

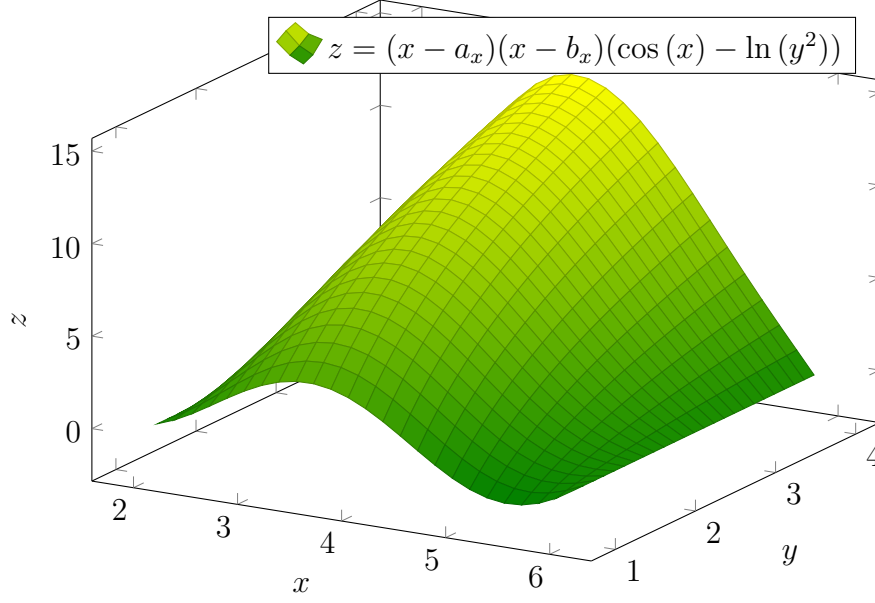


Рис. 2.

Нетрудно заметить, что введённая функция (2) равна нулю в одной паре краевых отрезках области (1), а именно при  $x = 2$ ,  $x = 4 \forall y \in [1, 4]$ . Таким образом, в направлении по оси абсцисс удобно будет применить **дискретное преобразование Фурье**. А в направлении по оси ординат применим метод восстановления кривой при помощи **кубических интерполяционных сплайнов**. Затем, получив значения во всех узлах, получим восстановленную поверхность.

Далее, для оценки точности каждого из методов нужно вычислить, на сколько построенная поверхность отличается от истинной, взяв разность сгенерируемых и истинных значений в потерянных узлах. Также необходимо узнать скорость сходимости восстановленной матрицы к истинной.

В качестве входных данных для алгоритмов, реализующих восстановление значений поверхностной функции по имеющимся узлам, выступают четыре числа:  $a_x, b_x, a_y, b_y \in \mathbb{R}$ , описывающие прямоугольную область  $\Omega = \{(x, y) \in \mathbb{R}^2 | a_x \leq x \leq b_x, a_y \leq y \leq b_y\}$  и две прямоугольные  $(m + 1) \times (n + 1)$  матрицы: матрица  $MatrixTrue \in \mathbb{R}_{n+1}^{m+1}$  значений известной функции  $f(x, y)$  в некоторых точках, в которой на  $(i, j)$ -ом месте стоит значение  $f(a_x + j \frac{b_x - a_x}{n}, a_y + i \frac{b_y - a_y}{m})$ , а также матрица  $MatrixFilter \in \mathbb{R}_{n+1}^{m+1}$  индикаторов, состоящая из нулей и единиц, где единица на  $(i, j)$ -ом месте означает, что имеется значение функции  $f(x, y)$  в точке  $(a_x + j \frac{b_x - a_x}{n}, a_y + i \frac{b_y - a_y}{m})$ , а ноль - что это значение потеряно. Предполагается, что в матрице значений функции на  $(i, j)$ -ом месте стоит любое число, если в матрице индикаторов на  $(i, j)$ -ом месте находится ноль. Данное введение необходимо для упрощения представления входных данных, а также для более удобной реализации алгоритма.

Для того, чтобы реализовать алгоритм в горизонтальном направлении, то есть по оси абсцисс, необходимо разбить имеющуюся область  $\Omega$  на горизонтальные промежутки, и применить метод дискретного преобразования Фурье для восстановления значений функции на отрезке по каждому получившемуся в результате разбиения горизонтальному отрезку  $\{(x_k, y_s) \in \mathbb{R}^2 | y_s \in [a_y, b_y], s = 0, \dots, m - \text{фиксированная точка, а } x_k \in [a_x, b_x],$

пробегают все значения при  $k = 0, \dots, n$ , а затем скомпановать получившиеся восстановленные значения по каждому отрезку в единую матрицу  $MatrixRecovery \in \mathbb{R}_{n+1}^{m+1}$  восстановленных значений. Данная матрица будет выступать в качестве выходных данных программы.

Для того, чтобы реализовать алгоритм в вертикальном направлении, то есть по оси ординат, необходимо разбить имеющуюся область  $\Omega$  на вертикальные промежутки, и применить метод кубических интерполяционных сплайнов для восстановления значений функции на отрезке по каждому получившемуся в результате разбиения вертикальному отрезку  $\{(x_k, y_s) \in \mathbb{R}^2 | x_k \in [a_x, b_x], k = 0, \dots, n - \text{фиксированная точка, а } y_s \in [a_y, b_y], \text{ пробегает все значения при } s = 0, \dots, m\}$ , а затем также скомпановать получившиеся восстановленные значения по каждому отрезку в единую матрицу  $MatrixRecovery \in \mathbb{R}_{n+1}^{m+1}$  восстановленных значений. Данная матрица будет выступать в качестве выходных данных программы.

Для оценки точности работы реализованного алгоритма предлагается использовать заранее известную функцию  $f(x, y)$ . Сначала полностью заполняется матрица истинных значений, затем случайно генерируется матрица индикаторов. После этого обе матрицы подаются на вход одному из описанных выше алгоритмов, каждый из которых выводит матрицу восстановленных значений. Если в некотором узле  $(i, j)$  значение было известно алгоритму, то в выходной матрице восстановленных значений в этом узле будет находиться в точности известное значение, а если в узле  $(i, j)$  значение подавалось на вход алгоритму, как потерянное, то в этом узле матрицы восстановленных значений будет находиться значение, отличающееся от истинного на некоторую погрешность. Так как, мы подали на вход алгоритму матрицу, созданную при помощи заранее известной функции, мы можем сравнить истинные и восстановленные значения и сгенерировать матрицу  $MatrixDelta \in \mathbb{R}_{n+1}^{m+1}$ , состоящую из разниц истинных и восстановленных значений.

Далее вычислим скорость сходимости восстановленной матрицы к истинной. Для этого определим множество неизвестных значений:

$$\Gamma = \{(i, j) \mid i = 0, \dots, m, j = 0, \dots, n, MatrixFilter_{i,j} = 0\} \quad (3)$$

Также понадобится определить количество неизвестных значений:

$$\gamma = |\Gamma| = \#\{(i, j) \mid i = 0, \dots, m, j = 0, \dots, n, MatrixFilter_{i,j} = 0\} \quad (4)$$

Скорость сходимости восстановленной матрицы к истинной будет характеризоваться константой  $p$ , определяемой следующим образом:

$$\begin{aligned} \frac{1}{\gamma} \sum_{(i,j) \in \Gamma} MatrixDelta_{i,j} &\approx \left( \frac{b_y - a_y}{m} \right)^p = (h_y)^p \\ p &= \log_{h_y} \left( \frac{1}{\gamma} \sum_{(i,j) \in \Gamma} MatrixDelta_{i,j} \right) \end{aligned} \quad (5)$$

При помощи значений (5) скорости сходимости восстановленной матрицы к истинной можно будет судить о точности применённого метода, а также сравнивать этот метод с другими.

### 3. Дискретное преобразование Фурье

#### 3.1. Описание метода

Дискретное преобразование Фурье можно применять, если на отрезке  $[0, 1]$  задана 1-периодическая функция  $g(x)$ , то есть  $g(0) = g(1) = 0$ . В качестве базиса можно взять систему

$$\{1, \cos 2\pi x, \sin 2\pi x, \dots, \cos 2\pi mx, \sin 2\pi mx\} \quad (6)$$

и построить ряд Фурье для  $g(x)$ .

Если функция  $g(x)$  является  $(b-a)$ -периодической, то есть  $g(a) = g(b) = 0$ , то рассмотрим  $g_j = g(a + jh)$ ,  $h = \frac{b-a}{n}$ ,  $0 \leq j \in \mathbb{Z} \leq n$ ,  $n \in \mathbb{N}$ . На пространстве дискретных функций  $g_j$ ,  $g_0 = g_n = 0$  определим скалярное произведение

$$(g, p) = \frac{1}{n} \sum_{j=1}^{n-1} g_j p_j \quad (7)$$

Тогда система

$$\{\varphi_j^{(m)} = C \sin \frac{\pi m j h}{b-a}, \quad m = 1, \dots, n-1, \quad j = 0, \dots, n\} \quad (8)$$

ортогональна. Более того, система (8) является ещё и ортонормальной при некотором значении константы  $C$ .

Покажем сначала ортогональность:

$$\begin{aligned} (\varphi_j^{(m)}, \varphi_j^{(k)}) &= \frac{1}{n} \sum_{j=1}^{n-1} (\varphi_j^{(m)} \cdot \varphi_j^{(k)}) = \frac{1}{n} \sum_{j=1}^{n-1} \left( C \sin \frac{\pi m j h}{b-a} \cdot C \sin \frac{\pi k j h}{b-a} \right) = \\ &= \frac{C^2}{n} \sum_{j=1}^{n-1} \left( \sin \frac{\pi m j h}{b-a} \sin \frac{\pi k j h}{b-a} \right) = \frac{C^2}{2n} \sum_{j=1}^{n-1} \left( \cos \frac{\pi(m-k)jh}{b-a} - \cos \frac{\pi(m+k)jh}{b-a} \right) = \\ &= \frac{C^2}{2n} \sum_{j=1}^{n-1} \operatorname{Re} \left[ \exp\left(\frac{i\pi(m-k)jh}{b-a}\right) - \exp\left(\frac{i\pi(m+k)jh}{b-a}\right) \right] = \\ &= \frac{C^2}{2n} \sum_{j=1}^{n-1} \operatorname{Re} \left[ \exp\left(\frac{i\pi m j h}{b-a}\right) \left( \exp\left(\frac{-i\pi k j h}{b-a}\right) - \exp\left(\frac{i\pi k j h}{b-a}\right) \right) \right] = \\ &= \frac{C^2}{2n} \sum_{j=1}^{n-1} \left( \cos \frac{\pi m j h}{b-a} \cdot \operatorname{Re} \left[ \exp\left(\frac{-i\pi k j h}{b-a}\right) - \exp\left(\frac{i\pi k j h}{b-a}\right) \right] \right) = \\ &= \frac{C^2}{2n} \sum_{j=1}^{n-1} \left( \cos \frac{\pi m j h}{b-a} \cdot \operatorname{Re} \left[ \cos \frac{\pi k j h}{b-a} - i \sin \frac{\pi k j h}{b-a} - \cos \frac{\pi k j h}{b-a} - i \sin \frac{\pi k j h}{b-a} \right] \right) = \\ &= \frac{C^2}{2n} \sum_{j=1}^{n-1} \left( \cos \frac{\pi m j h}{b-a} \cdot \operatorname{Re} \left[ -2i \cdot \sin \frac{\pi k j h}{b-a} \right] \right) = \frac{C^2}{2n} \sum_{j=1}^{n-1} \left( \cos \frac{\pi m j h}{b-a} \cdot 0 \right) = \\ &= 0, \quad \forall 1 \leq k \neq m \leq n-1 \end{aligned} \quad (9)$$

Таким образом, из (9) следует ортогональность системы (8). Для доказательства ортонормальности системы (8) необходимо найти константу  $C$  так, чтобы:

$$(\varphi^{(m)}, \varphi^{(m)}) = 1, \quad \forall 1 \leq m \leq n-1 \quad (10)$$

$$\begin{aligned} (\varphi^{(m)}, \varphi^{(m)}) &= \frac{1}{n} \sum_{j=1}^{n-1} (\varphi_j^{(m)} \cdot \varphi_j^{(m)}) = \frac{1}{n} \sum_{j=1}^{n-1} C^2 \sin^2 \frac{\pi m j h}{b-a} = \frac{C^2}{2n} \sum_{j=1}^{n-1} \left(1 - \cos \frac{2\pi m j h}{b-a}\right) = \\ &= \frac{C^2}{2n} \sum_{j=1}^{n-1} \left(1 - \cos \frac{2\pi m j}{n}\right) = \frac{C^2(n-1)}{2n} - \frac{C^2}{2n} \sum_{j=1}^{n-1} \cos \frac{2\pi m j}{n} = \frac{C^2(n-1)}{2n} - \\ &- \frac{C^2}{2n} \operatorname{Re} \left[ \sum_{j=1}^{n-1} \exp\left(\frac{2\pi m i j}{n}\right) \right] = \frac{C^2(n-1)}{2n} - \frac{C^2}{2n} \operatorname{Re} \left[ \frac{\exp\left(\frac{2\pi m i}{n}\right) \left(\exp\left(\frac{2\pi m i (n-1)}{n}\right) - 1\right)}{\exp\left(\frac{2\pi m i}{n}\right) - 1} \right] = \\ &= \frac{C^2(n-1)}{2n} - \frac{C^2}{2n} \operatorname{Re} \left[ \frac{\exp\left(\frac{2\pi m i n}{n}\right) - \exp\left(\frac{2\pi m i}{n}\right)}{\exp\left(\frac{2\pi m i}{n}\right) - 1} \right] = \frac{C^2(n-1)}{2n} - \frac{C^2}{2n} \operatorname{Re} \left[ \frac{1 - \exp\left(\frac{2\pi m i}{n}\right)}{\exp\left(\frac{2\pi m i}{n}\right) - 1} \right] = \\ &= \frac{C^2(n-1)}{2n} + \frac{C^2}{2n} = \frac{C^2}{2} = 1 \Leftrightarrow C = \sqrt{2} \end{aligned} \quad (11)$$

Таким образом, из (11) следует, что для ортонормальности системы (8) необходимо и достаточно положить  $C = \sqrt{2}$ . Тогда ортонормальный базис Фурье имеет вид:

$$\{\varphi_j^{(m)} = \sqrt{2} \sin \frac{\pi m j h}{b-a} = \sqrt{2} \sin \frac{\pi m j}{n}, \quad m = 1, \dots, n-1, \quad j = 0, \dots, n\} \quad (12)$$

Разложение функции  $g(x)$  по ортонормальному базису (12) имеет вид:

$$g_j = \sum_{m=1}^{n-1} c_m \varphi_j^{(m)}, \quad c_m = (g_j, \varphi_j^{(m)}) \quad \text{и} \quad g(x) \approx \sum_{m=1}^{n-1} c_m \varphi^{(m)}(x) \quad (13)$$

Где  $\varphi^{(m)}(x) = \sqrt{2} \sin \frac{\pi m(x-a)}{b-a}$  - значение базисной гармоник из системы (12) в некоторой промежуточной точке  $x$ .

### 3.2. Программная реализация метода

Дискретное преобразование Фурье подразумевает равномерное разбиения отрезка на узлы, поэтому для реализации данного метода необходимо.

А что необходимо? Не ясно.

## 4. Кубические интерполяционные сплайны

### 4.1. Описание метода

Пусть имеется функция  $p(x)$ , определённая на отрезке  $[a, b]$ , который разбит на подотрезки  $[x_{i-1}, x_i]$ ,  $i = 1, \dots, n$ , причём,  $x_0 = a$ ,  $x_n = b$ . Кубический интерполяционный сплайн на каждом из подотрезков  $[x_{i-1}, x_i]$  является полиномом третьей степени

$S_3(x) = a_i x_i^3 + b_i x_i^2 + c_i x + d_i$ , который проходит через точки  $(x_i, p(x_i))$ ,  $i = 0, \dots, n$  (два условия на каждом отрезке:  $2n$  условий) и имеет непрерывную первую и вторую производные в точках  $x_i$ ,  $i = 1, \dots, n-1$  (два условия для каждой внутренней точки:  $2(n-1) = 2n-2$  условий). Итого имеем  $4n-2$  уравнений, а неизвестных коэффициентов  $4n$ . Поэтому для однозначного решения, необходимо дополнительно задать ещё два уравнения.

Пусть  $M_i = S_3''(x_i)$ ,  $i = 0, \dots, n$ . Тогда  $M_i$ , в силу условия непрерывности второй производной во внутренних точках разбиения, удовлетворяют системе линейных уравнений  $C\vec{M} = \vec{d}$ , где прямоугольная матрица  $C \in \mathbb{R}_{n+1}^{n-1}$  и вектор-столбцы  $\vec{d} \in \mathbb{R}^{n-1}$ ,  $\vec{M} \in \mathbb{R}^{n+1}$  имеют вид:

$$C = \begin{pmatrix} \frac{h_1}{6} & \frac{h_1+h_2}{3} & \frac{h_2}{6} & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & \frac{h_2}{6} & \frac{h_2+h_3}{3} & \frac{h_3}{6} & \dots & 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & \frac{h_{n-2}}{6} & \frac{h_{n-2}+h_{n-1}}{3} & \frac{h_{n-1}}{6} & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & \frac{h_{n-1}}{6} & \frac{h_{n-1}+h_n}{3} & \frac{h_n}{6} \end{pmatrix} \quad (14)$$

$$\vec{d} = \left( \frac{p_2 - p_1}{h_2} - \frac{p_1 - p_0}{h_1}, \dots, \frac{p_n - p_{n-1}}{h_n} - \frac{p_{n-1} - p_{n-2}}{h_{n-1}} \right)^T \quad (15)$$

$$\vec{M} = (M_0, M_1, \dots, M_{n-1}, M_n)^T \quad (16)$$

Где  $h_i = x_i - x_{i-1}$ ,  $i = 1, \dots, n$ .

В качестве доопределяющих уравнений возьмём:

$$S_3''(x_0) = M_0 = S_3''(x_n) = M_n = 0 - \text{естественный сплайн} \quad (17)$$

Таким образом система линейных уравнений  $C\vec{M} = \vec{d}$  с учётом доопределённых уравнений (17) имеет вид:

$$\begin{pmatrix} \frac{h_1+h_2}{3} & \frac{h_2}{6} & 0 & \dots & 0 & 0 & 0 \\ \frac{h_2}{6} & \frac{h_2+h_3}{3} & \frac{h_3}{6} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \frac{h_{n-2}}{6} & \frac{h_{n-2}+h_{n-1}}{3} & \frac{h_{n-1}}{6} \\ 0 & 0 & 0 & \dots & 0 & \frac{h_{n-1}}{6} & \frac{h_{n-1}+h_n}{3} \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ \dots \\ M_{n-2} \\ M_{n-1} \end{pmatrix} = \begin{pmatrix} \frac{p_2-p_1}{h_2} - \frac{p_1-p_0}{h_1} \\ \frac{p_3-p_2}{h_3} - \frac{p_2-p_1}{h_2} \\ \dots \\ \frac{p_{n-1}-p_{n-2}}{h_{n-1}} - \frac{p_{n-2}-p_{n-3}}{h_{n-2}} \\ \frac{p_n-p_{n-1}}{h_n} - \frac{p_{n-1}-p_{n-2}}{h_{n-1}} \end{pmatrix} \quad (18)$$

Теперь мы можем однозначно решить систему линейных уравнений (18).

Получив все значения  $M_i$ ,  $i = 0, \dots, n$  в силу непрерывности и линейности вторых производных во внутренних точках, представим кубический сплайн на каждом подотрезке  $[x_{i-1}, x_i]$  в виде:

$$S_3''(x) = M_{i-1} \frac{x_i - x}{h_i} + M_i \frac{x - x_{i-1}}{h_i} \quad (19)$$

Для получения аналитического представления кубического интерполяционного сплайна на подотрезке  $[x_{i-1}, x_i]$  необходимо дважды проинтегрировать полученное выражение второй производной (19) с учётом  $S_3(x_i) = p_i$ ,  $S_3(x_{i-1}) = p_{i-1}$ :

$$\begin{aligned} S_3(x) &= M_{i-1} \frac{(x_i - x)^3}{6h_i} + M_i \frac{(x - x_{i-1})^3}{6h_i} + \\ &+ \left( f_{i-1} - \frac{M_{i-1}h_i^2}{6} \right) \frac{x_i - x}{h_i} + \left( f_i - \frac{M_i h_i^2}{6} \right) \frac{x - x_{i-1}}{h_i} \end{aligned} \quad (20)$$



## 4.2. Программная реализация метода

Реализовав данный алгоритм в виде программного кода и выполнив его, получаем следующую таблицу значений скорости сходимости, где  $n$  - количество отрезков разбиения области  $\Omega$  по оси абсцисс, а  $m$  - количество отрезков разбиения области  $\Omega$  по оси ординат:

$n \backslash m$	100	200	300	400	500	600	700	800	900	1000
10	3.03	2.90	<b>3.31</b>	2.92	3.02	3.04	2.97	2.93	2.83	3.06
20	2.85	2.90	<b>3.08</b>	2.84	2.81	3.00	2.89	2.94	2.91	2.96
30	2.95	3.00	2.96	2.97	2.97	3.06	3.02	<b>3.08</b>	3.07	2.07
40	<b>3.25</b>	2.89	2.85	2.99	3.08	3.03	3.07	2.82	3.01	2.93
50	2.77	2.97	<b>3.00</b>	2.95	2.95	2.99	2.97	2.95	2.96	2.89
Среднее	2.97	2.93	<b>3.04</b>	2.93	2.97	3.02	2.98	2.94	2.96	2.79

Таблица 1. Скорость (5) сходимости восстановленной матрицы к истинной

По результатам Таблицы 1. можно сделать вывод, что при очень маленьких значениях количества отрезков разбиения (100 и 200) точность восстановления не высокая в силу слишком больших длин отрезков разбиения, в то же время, при больших значениях количества отрезков разбиения (700 и больше) мы теряем точность в силу машинной погрешности, так как процессору приходится работать с очень маленькими числами. Чтобы уменьшить зависимость точности от количества точек разбиения по оси абсцисс, возьмём при каждом значении  $m$  среднее арифметическое по всем значениям  $n$ . Средние значения приведены в последней строке Таблицы 1.

Как итог, можно выбрать **оптимальное значение** количества отрезков разбиения по оси ординат, это **300**, а скорость сходимости восстановленной матрицы к истинной равна **3.04**.

Таким образом, мы построили алгоритм восстановления потерянных значений функции в вертикальном направлении по оси ординат.