

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ФГБОУ ВО «КубГУ»)
Кафедра математических и компьютерных методов
ДИПЛОМНАЯ РАБОТА

Устойчивый алгоритм аппроксимации в гильбертовом пространстве по любой независимой системе функций

Работу выполнил Д. А. Пасько

(подпись, дата)

Факультет математики и компьютерных наук курс 3

Направление 02.03.01 математика и компьютерные науки

Научный руководитель доцент кафедры МКМ,

канд. физ.-мат. наук А. А. Свидлов

(подпись, дата)

Нормоконтролер

ст. лаборант Ю. А. Кравченко

(подпись, дата)

Краснодар 2018

1 ВВЕДЕНИЕ

Аппроксимация функций – одно из самых востребованных направлений прикладной математики. Задача аппроксимации считается полностью решённой в сепарабельных гильбертовых пространствах, однако ввиду ограничений, налагаемых компьютерной арифметикой, современная математическая теория не может быть применима на практике во всей полноте. Задачи из многих математических областей при численном решении сталкиваются с существенными ограничениями; в теории приближений таким ограничением является неустойчивость аппроксимации. Для обхода этого ограничения требуется использовать специальные алгоритмы, учитывающие как особенности компьютерной реализации, так и естественные требования к точности и производительности.

В этой работе описан алгоритм сложности $O(n^4)$, реализующий устойчивую аппроксимацию. Также описываются многие особенности численной аппроксимации, которые пришлось учитывать при разработке этого алгоритма.

2 Обозначения

(x, y) – точка на плоскости с координатами x, y ; $x = (x_1, \dots, x_n)$ – вектор из пространства R^n ; $x^m = (x_1, \dots, x_m)$ – m -мерный вектор; $\Delta = \sum_{i=1}^n \frac{\partial^2}{\partial^2 x_i}$ – оператор Лапласа; $L(f_1, \dots, f_m) = \sum_{i=1}^m \alpha_i f_i$, $\alpha_i \in R$ – множество всевозможных линейных комбинаций элементов f_1, \dots, f_m (линейная оболочка); $\varphi = \varphi(x)$ – граничная функция; $D_{c_i} = \frac{\partial}{\partial c_i}$ – частная производная по аргументу c_i ; Q – область на плоскости либо некоторое произвольное множество; $L_2(\partial Q)$ – пространство функций, определённых на кривой ∂Q и интегрируемых с квадратом в смысле Лебега; (α_n, α_m) – скалярное произведение функций в определённом пространстве $(\alpha_n, \alpha_m)_{L_2(\partial Q)} = \oint_{\partial Q} \alpha_n(x) \alpha_m(x) dl$ – скалярное произведение функций $(\alpha_n, \alpha_m) \in L_2(\partial Q)$; $Q^+ = R^2 \setminus (Q \cup \partial Q)$; ε – некоторое очень малое число, большее нуля; а. п. – аналитическое продолжение функции на всю плоскость, используемое для упрощения программирования этой функции.

3 Общие сведения о задаче аппроксимации

В общем случае задача аппроксимации состоит в следующем: требуется максимально точно приблизить некоторый элемент $x \in X$ элементами пространства $F \subset X$, $x \notin F$. Качество приближения оценивается по тому, насколько малым оказывается расстояние от элемента $x \in X$ до элемента $u \in F$, наиболее близкого к x в заданной метрике; само же это расстояние выражается формулой:

$$\rho(x, F) = \inf_{u \in F} \|x - u\|. \quad (1)$$

Если для какого-то $u_0 \in F$ инфимум достигается, то u_0 называется элементом наилучшего приближения. Этот элемент и представляет из себя цель поиска. Далее будем рассматривать только такие случаи, когда элементы наилучшего приближения существуют.

На практике аппроксимация происходит исключительно элементами конечномерных пространств. Так как в конечномерных пространствах имеются базисы, можно выбрать любой из них и искать элемент наилучшего приближения как линейную комбинацию элементов базиса. Сразу же ставятся вопросы: 1) найдётся ли на подпространстве элемент наилучшего приближения и 2) будет ли он единственным? На оба вопроса в самых распространённых случаях ответы положительны; во-первых, в конечномерном подпространстве элемент наилучшего приближения однозначно существует ([3, 4]); во-вторых, при нахождении решения обычно работают с функциями из строго нормированных пространств (включая $L_2(Q)$, где Q – компакты либо поверхности в многомерном евклидовом пространстве), а из строгой нормированности следует единственность элемента наилучшего приближения; более того, пространство $L_2(Q)$ является гильбертовым, а потому наилучшим приближением элемента $x \in L_2(Q)$ является его проекция на линейную оболочку $L(f_1, \dots, f_m)$ исходного множества. Таким образом, задача аппроксимации сводится к задаче минимизации функционала

$$F(c) = \left\| \varphi - \sum_{m=1}^N c_m \alpha_m \right\|_{L_2(Q)} \rightarrow \min, \quad (2)$$

где $\varphi \in L_2(Q)$ – аппроксимируемая функция, $\alpha_m \in L_2(Q)$, $m = 1, \dots, N$ – базис подпространства, в котором ищется элемент наилучшего приближения.

Решим задачу аппроксимации аналитически. Так как пространство $L_2(Q)$ – гильбертово, то норма в нём выражается через скалярное произведение:

$$\left\| \varphi - \sum_{m=1}^N c_m \alpha_m \right\|_{L_2(Q)} = \sqrt{\left(\varphi - \sum_{m=1}^N c_m \alpha_m, \varphi - \sum_{m=1}^N c_m \alpha_m \right)_{L_2(Q)}} = \sqrt{(\varphi, \varphi) - 2 \sum_{m=1}^N c_m (\varphi, \alpha_m) + \sum_{n=1}^N c_n \sum_{m=1}^N c_m (\alpha_n, \alpha_m)} \quad (3)$$

Так как норма является неотрицательной по определению, то предыдущее выражение можно возвести в квадрат без риска потерять решения или обрести новые:

$$\left\| \varphi - \sum_{m=1}^N c_m \alpha_m \right\|_{L_2(Q)}^2 = (\varphi, \varphi)_{L_2(Q)} - 2 \sum_{m=1}^N c_m (\varphi, \alpha_m)_{L_2(Q)} + \sum_{n=1}^N c_n \sum_{m=1}^N c_m (\alpha_n, \alpha_m)_{L_2(Q)} = (\varphi, \varphi)_{L_2(Q)} - 2 \sum_{m=1}^N c_m (\varphi, \alpha_m)_{L_2(Q)} + (c_1, c_2, \dots, c_N) \begin{pmatrix} (\alpha_1, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_1, \alpha_N)_{L_2(Q)} \\ \vdots & \ddots & \vdots \\ (\alpha_N, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_N, \alpha_N)_{L_2(Q)} \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_N \end{pmatrix}, \quad (4)$$

причём возникшая матрица Грама является невырожденной, если система функций $\alpha_m \in L_2(Q)$ линейно независима; следовательно, аппроксимировать имеет смысл только по линейно независимым система (иначе получится неоднозначное решение).

Получено выражение квадратичного функционала. Вычислим стационарные точки этого функционала из системы:

$$D_{c_i} F(c) = 0, \quad i = 1, 2, \dots, N, \quad (5)$$

где

$$\begin{aligned} D_{c_i} F(c) &= D_{c_i} \left((\varphi, \varphi)_{L_2(Q)} - 2 \sum_{m=1}^N c_m (\varphi, \alpha_m)_{L_2(Q)} + \sum_{n=1}^N c_n \sum_{m=1}^N c_m (\alpha_n, \alpha_m)_{L_2(Q)} \right) = \\ &= -2(\varphi, \alpha_i)_{L_2(Q)} + \sum_{n=1}^N D_{c_i}(c_n) \sum_{m=1}^N c_m (\alpha_n, \alpha_m)_{L_2(Q)} + \sum_{n=1}^N c_n D_{c_i} \sum_{m=1}^N c_m (\alpha_n, \alpha_m)_{L_2(Q)} = \\ &= -2(\varphi, \alpha_i)_{L_2(Q)} + \sum_{m=1}^N c_m (\alpha_i, \alpha_m)_{L_2(Q)} + \sum_{n=1}^N c_n (\alpha_n, \alpha_i)_{L_2(Q)} = \\ &= 2 \left(\sum_{m=1}^N c_m (\alpha_i, \alpha_m)_{L_2(Q)} - (\varphi, \alpha_i)_{L_2(Q)} \right) \end{aligned} \quad (6)$$

Такая система сводится к виду

$$\Gamma c = \begin{pmatrix} (\varphi, \alpha_1)_{L_2(Q)} \\ \vdots \\ (\varphi, \alpha_N)_{L_2(Q)} \end{pmatrix}, \quad (7)$$

где $c = (c_1, c_2, \dots, c_N)$, Γ – матрица Грама нашей системы. Так как эта матрица является невырожденной, система имеет единственное решение; поскольку второй дифференциал функции $F(c)$ является положительно определённой квадратичной формой с той же матрицей Грама, то в точке решения нашей системы функция $F(c)$ действительно имеет минимум. Сама же погрешность аппроксимации равна:

$$\begin{aligned} \rho \left(\sum_{m=1}^N c_m \alpha_m, \varphi \right) &= \left\| \varphi - \sum_{m=1}^N c_m \alpha_m \right\|_{L_2(Q)} = \sqrt{\oint_Q \left(\sum_{m=1}^N c_m \alpha_m - \varphi \right)^2 dl} = \\ &= \left((\varphi, \varphi)_{L_2(Q)} - 2 \sum_{m=1}^N c_m (\varphi, \alpha_m)_{L_2(Q)} + (c_1, c_2, \dots, c_N) \begin{pmatrix} (\alpha_1, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_1, \alpha_N)_{L_2(Q)} \\ \vdots & \ddots & \vdots \\ (\alpha_N, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_N, \alpha_N)_{L_2(Q)} \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_N \end{pmatrix} \right)^{\frac{1}{2}} = \\ &= \left((\varphi, \varphi)_{L_2(Q)} - 2 (c_1, c_2, \dots, c_N) \begin{pmatrix} (\varphi, \alpha_1)_{L_2(Q)} \\ \vdots \\ (\varphi, \alpha_N)_{L_2(Q)} \end{pmatrix} + (c_1, c_2, \dots, c_N) \begin{pmatrix} (\alpha_1, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_1, \alpha_N)_{L_2(Q)} \\ \vdots & \ddots & \vdots \\ (\alpha_N, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_N, \alpha_N)_{L_2(Q)} \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_N \end{pmatrix} \right)^{\frac{1}{2}} = \\ &= \left((\varphi, \varphi)_{L_2(Q)} - (c_1, c_2, \dots, c_N) \begin{pmatrix} (\varphi, \alpha_1)_{L_2(Q)} \\ \vdots \\ (\varphi, \alpha_N)_{L_2(Q)} \end{pmatrix} \right)^{\frac{1}{2}}. \end{aligned} \quad (8)$$

На этом задача аппроксимации в гильбертовом пространстве считается теоретически решённой. На данный момент мы условились брать только линейно независимые системы функций, иначе задача не решится ввиду вырожденной матрицы Грама. Также крайне желательно, чтобы система

$\alpha_m \in L_2(Q)$ обладала свойством полноты; полная система сможет обеспечивать безграничное возрастание точности аппроксимации при своём расширении. Система из множества $S \subset L_2(Q)$ является полной, если для любого элемента f из $L_2(Q)$ и для всякой наперёд заданной точности найдётся комбинация элементов из S , обеспечивающая данную точность; для практики полнота системы означает, что с ростом числа функций точность всегда будет улучшаться:

$$\left\| f - \sum_{s \in S_n} c_s(f) s \right\| \rightarrow 0, \quad n \rightarrow \infty, \quad S_1 \subset S_2 \subset \dots \subset S_n \subset \dots \subset S. \quad (9)$$

Если же система функций не будет полной, мы никакими способами не сможем осуществить с её помощью приближение лучшее, чем некоторое. Но на практике мы сталкиваемся с той же самой проблемой даже при работе с полными системами: ошибки округлений и заведомо неточные вычисления приводят к тому, что аппроксимация становится крайне неустойчивой, поэтому теряются гарантии эффективного использования теории на практике. Цель следующего изложения – указать причины неустойчивости и предложить алгоритм, исключающий её.

4 Причины неустойчивости аппроксимации и примеры неустойчивости для распространённых систем функций

Наибольший вклад в неустойчивость аппроксимации вносят три причины:

1. Скалярные произведения, заполняющие СЛАУ, обычно представляются конечными суммами или интегралами; при этом это могут быть интегралы по сложным областям или поверхностям или интегралы от функций, которые сами задаются интегралами; в таком случае погрешности квадратурных формул накапливаются и приводят к тому, что исходная СЛАУ несколько отличается от истинной, что негативно сказывается на решении задачи. Даже если удаётся подобрать реализуемый способ интегрирования, часто попытки повысить точность существенно увеличивают время вычислений.
2. Матрица Грамма для любой системы функций является плохо обусловленной по определению, причём число обусловленности быстро растёт с увеличением размерности, из-за чего даже малые отклонения в элементах системы (которые будут иметь место хотя бы по причине пункта 1) приводят к неточным решениям.
3. Огромную роль играют погрешности округления при суммировании больших и маленьких чисел, делении и умножении; эти погрешности заметны при вычислении интегралов, решении СЛАУ и даже при замене координат (для получения системы, ортогональной на произвольном отрезке). Позднее будет показано, что при плохо обусловленной матрице системы даже итерационные методы не способны из-за ошибок округления приблизиться к истинному решению ближе некоторого расстояния. Комбинация пунктов 1) и 3) приводит к тому, что даже аппроксимация по ортогональным системам обладает неустойчивостью при достаточно большом числе функций.

Далее рассматривается аппроксимация для наиболее распространённых систем функций:

1. Мономы $(1, x, x^2, x^3, \dots)$. Традиционный пример полной системы. Матрица Грамма для мономов есть так называемая матрица Гильберта – классический пример плохо обусловленной матрицы. Из-за плохой обусловленности неустойчивость возникает выраженная, хотя на первых порах приближение происходит лучше, чем по другим системам. Время и погрешности вычислений можно уменьшить, если реализовать мономы в виде некоторого класса с реализацией схемы Горнера.

2. Тригонометрические полиномы $(1/2, \sin x, \cos x, \sin 2x, \dots)$. Распространённая ортогональная система, часто используемая при аппроксимации колебаний. Основной недостаток – медленная сходимость: иногда для приличной аппроксимации приходится использовать несколько сотен функций.
3. Многочлены Лежандра $(P_0 = 1, P_1 = x, P_{n+1} = \frac{2n+1}{n+1}xP_n - \frac{n}{n+1}P_{n-1})$. Результат ортогонализации системы из пункта 1, за счёт ортогональности эта система более устойчива, однако численная реализация для большой размерности из-за рекурсии приводит к сильно заметным временным затратам.
4. Экспоненты $(1, e^x, e^{2x}, \dots)$. Неортогональная система, обладает медленной сходимостью и сильной неустойчивостью за счёт быстрого роста экспоненты и больших коэффициентов в СЛАУ.
5. Система Хаара. Ортогональная система, используется для аппроксимации шумов, обладает соразмерной сходимостью с системой 2, но неустойчивостью почти как у системы 4.

Неустойчивость каждой системы выражается по-своему. На следующем графике (Рис. 1) показана зависимость погрешности аппроксимации от числа функций по разным системам. Приближается функция $y = -e^{x/2} + |x| \cos(x)$ на отрезке $[-3, 1]$. На оси X показано количество функций, на оси Y – десятичный логарифм от погрешности. Все СЛАУ решались методом Гаусса с выбором максимального элемента, интегралы в СЛАУ считаются по формулам Гаусса-Кронрода при 61 точках.

Как видно из Рис. 1, системы мономов и Хаара начали показываться неустойчивость (скачки) при 16-18 функциях, система по экспонентам неустойчива уже с 11 функций и на 17-й имеет сильный скачок. Относительно аппроксимации получены следующие данные:

1. Для системы мономов наилучшая аппроксимация равна 0,0109251874472286 при числе функций 18
2. Для системы полиномов Лежандра наилучшая аппроксимация равна 0,0110161134996089 при числе функций 19
3. Для ортонормированной системы тригонометрических полиномов наилучшая аппроксимация равна 0,153425238548837 при числе функций 20
4. Для системы Хаара наилучшая аппроксимация равна 0,12656928280189 при числе функций 17
5. Для системы экспонент наилучшая аппроксимация равна 0,0148738092936054 при числе функций 11

Систему по экспонентам использовать нет смысла из-за сильной неустойчивости и очень медленного роста аппроксимации; многочлены Лежандра не могут быть использованы при больших размерностях из-за больших затрат по времени. Остальные системы могут быть полезны, если уметь их корректировать. Увеличим число функций и пересмотрим результат (Рис. 2).

Погрешность по системе Хаара продолжила возрастать, погрешность по системе мономов при больше чем 20 функция целиком состоит из неустойчивости. При этом наилучшие результаты, можно сказать, почти не изменились:

1. Для системы мономов наилучшая аппроксимация равна 0,0106044647958804 при числе функций 36
2. Для ортонормированной системы тригонометрических полиномов наилучшая аппроксимация равна 0,0791846313802487 при числе функций 70

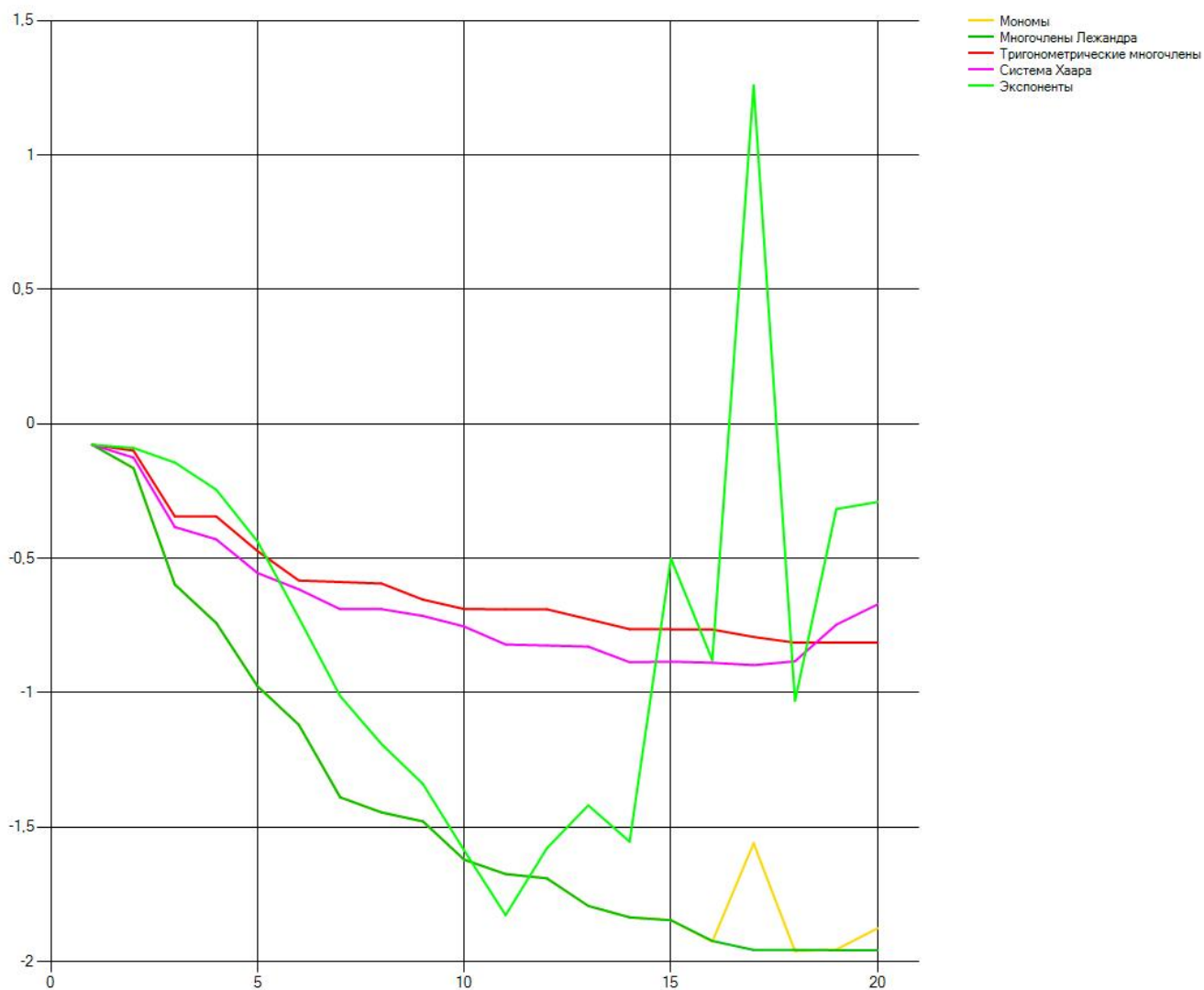


Рис. 1: Погрешность аппроксимации функции $y = -e^{x/2} + |x| \cos(x)$, $x \in [-3; 1]$ разными системами при разном количестве функций в системах

3. Для системы Хаара наилучшая аппроксимация равна 0,12656928280189 при числе функций 17

Дальнейшее увеличение числа функций приведёт к неустойчивости тригонометрической системы (Рис. 3).

На Рис. 4 изображены аналогичные графики для функции $y = e^{x/|x|+1}$ на отрезке $[0, 5]$ (далее часто будет проводиться аппроксимация именно этой функции на этом отрезке, дабы сравнивать результаты в равных условиях).

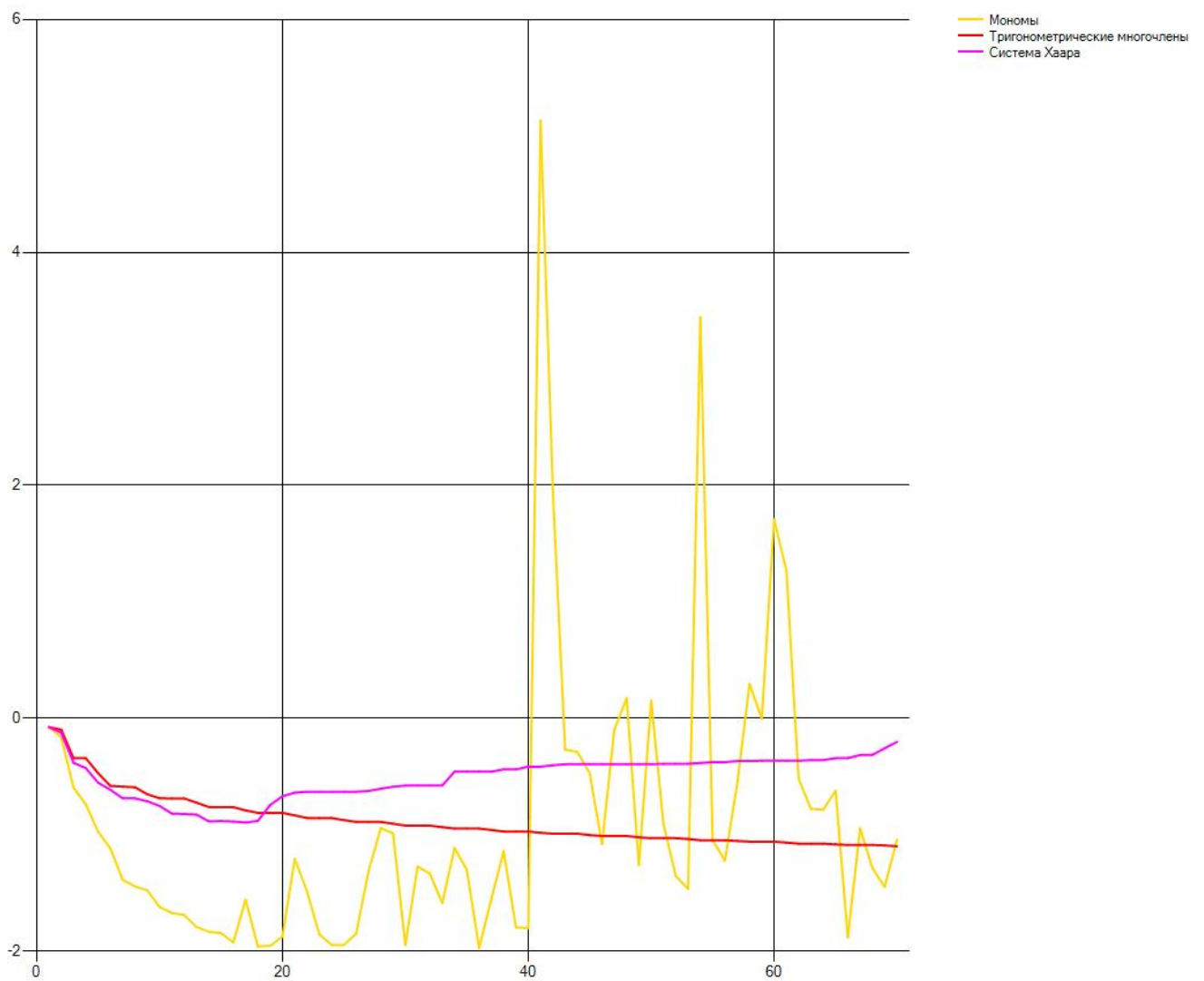


Рис. 2: Аппроксимация при тех же условиях, но с увеличенным числом функций в системах

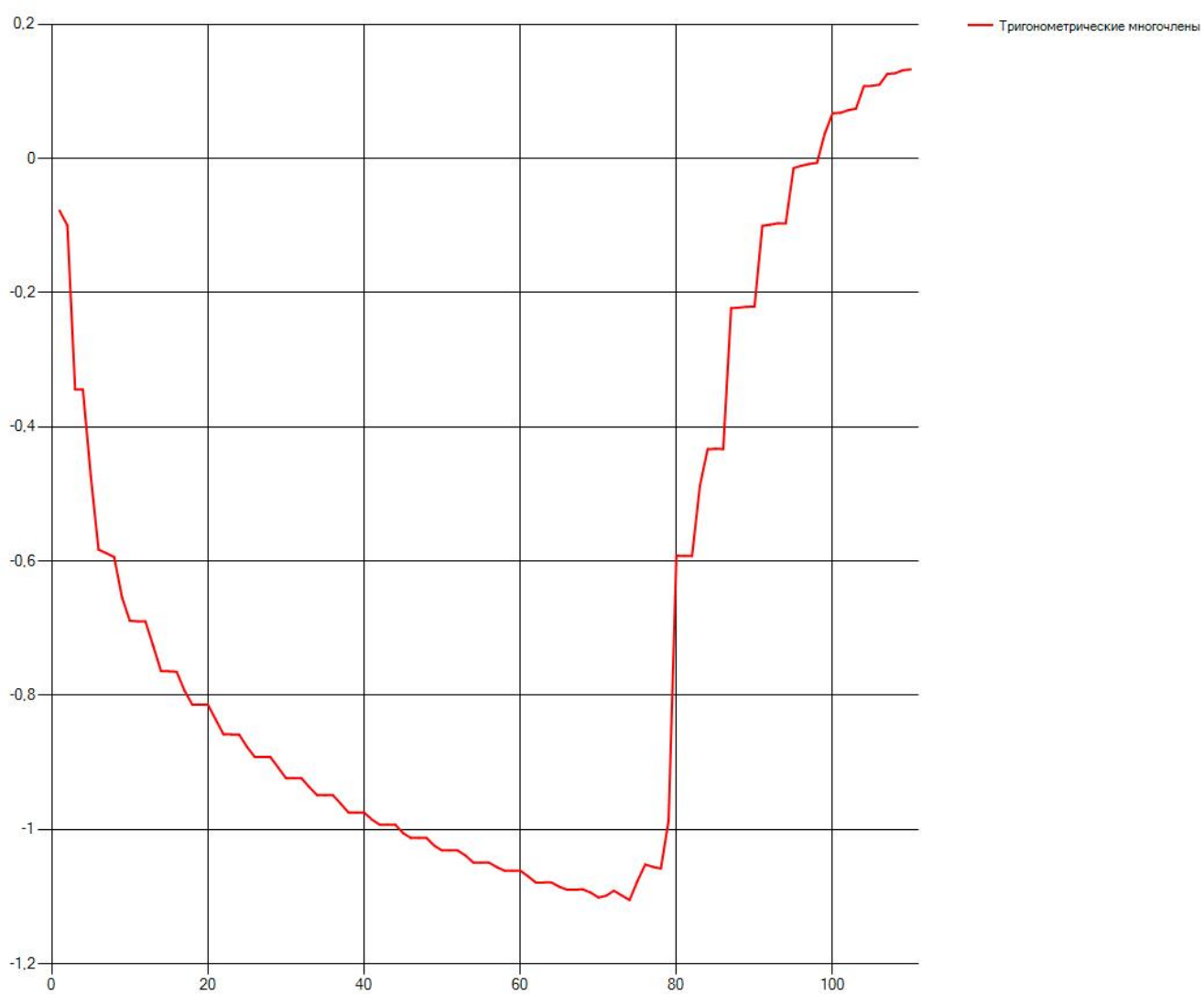


Рис. 3: Те же условия аппроксимации, ещё больше функций в системах

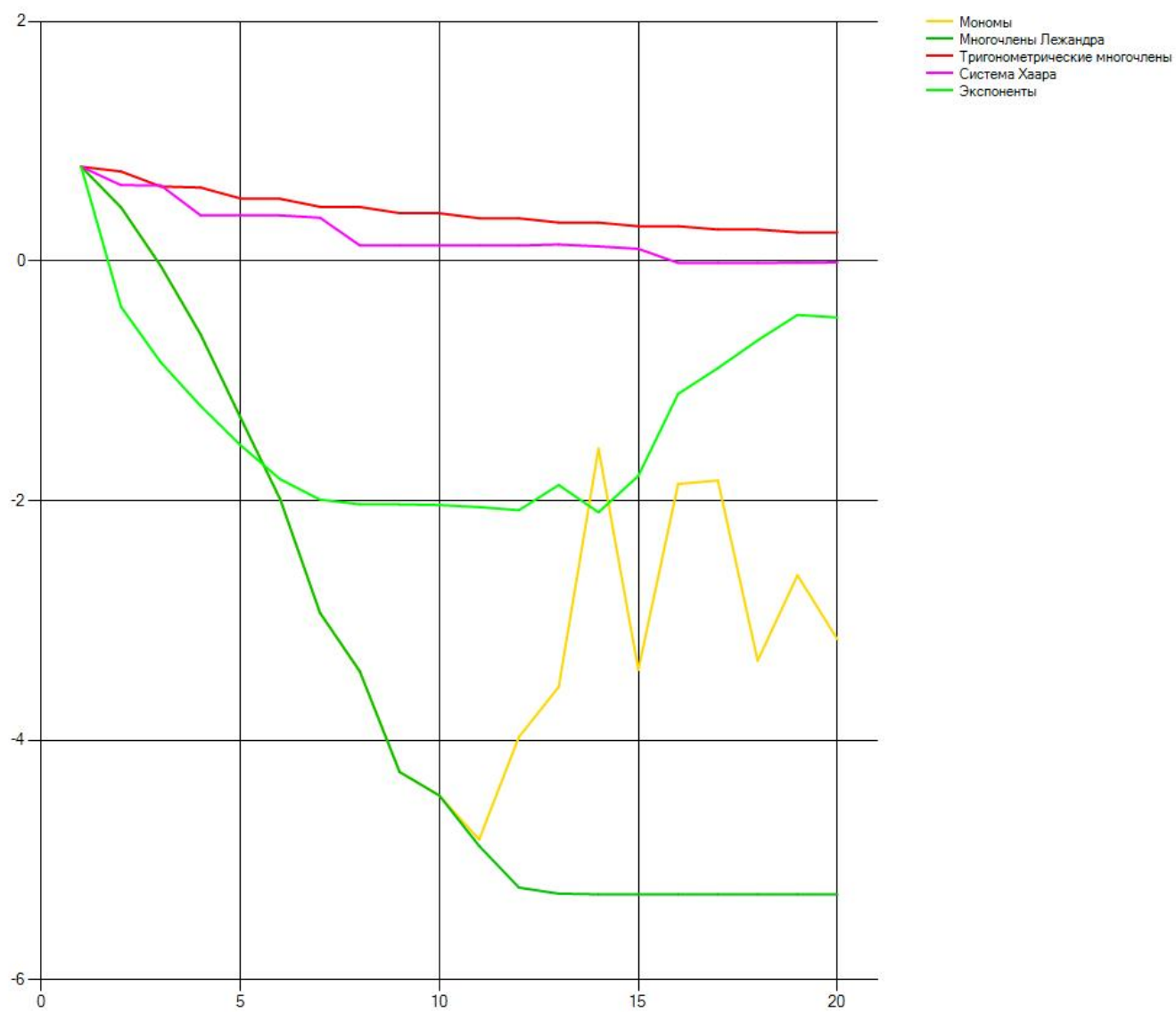


Рис. 4: Погрешность аппроксимации функции $y = e^{x/|x|+1}, x \in [0; 5]$ разными системами при разном количестве функций в системах

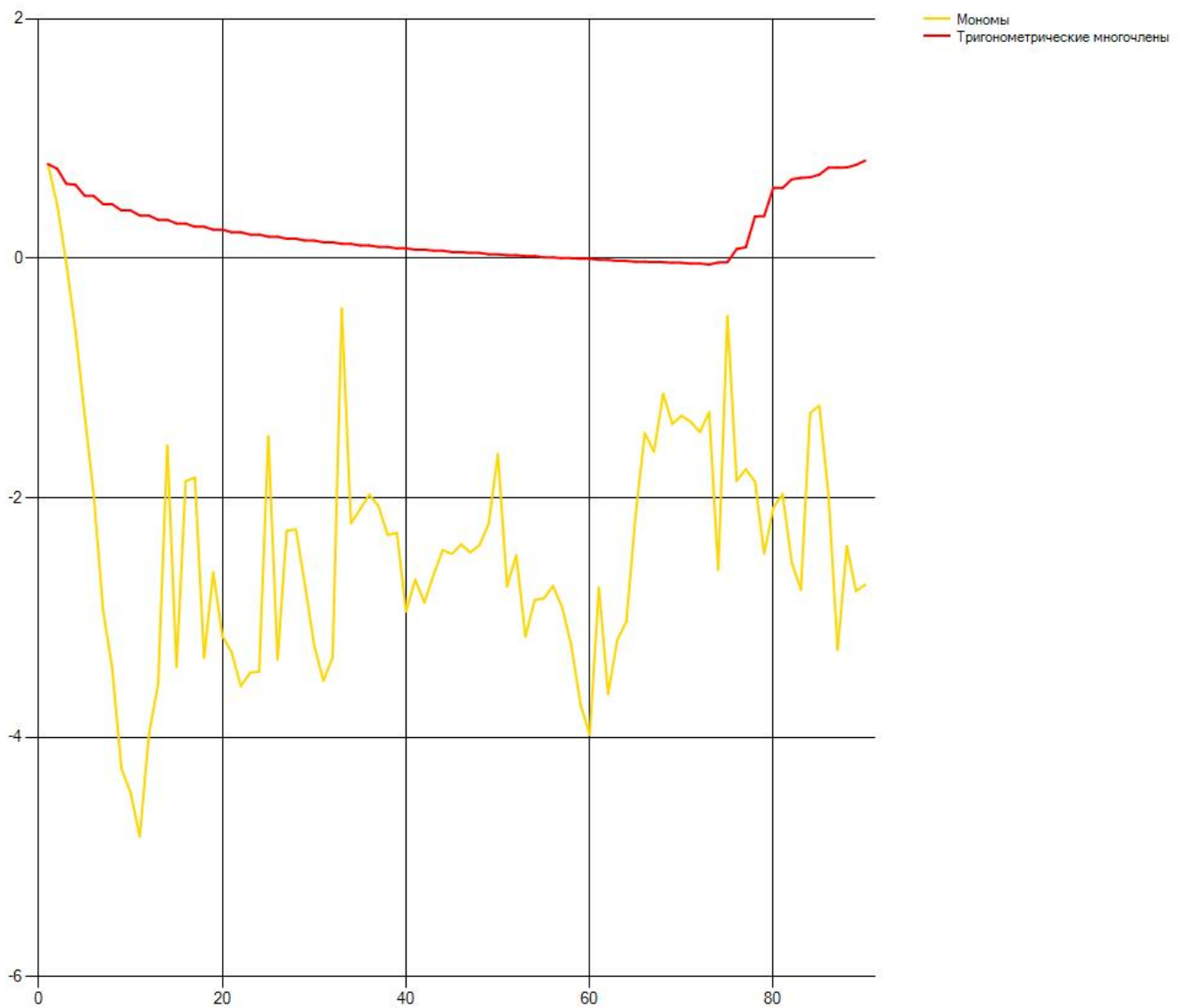


Рис. 5: Погрешность аппроксимации функции $y = e^{x/|x|+1}, x \in [0; 5]$ разными системами при разном количестве функций в системах (при большем количестве функций)

5 Попытки обойти неустойчивость, меняя точность интегрирования и комбинируя разные методы решения СЛАУ

Если при интегрировании использовать квадратуры Гаусса-Кронрода при 15 точках (а не 61), неустойчивость в предыдущем случае становится намного выраженнее и наступает раньше (Рис. 6).

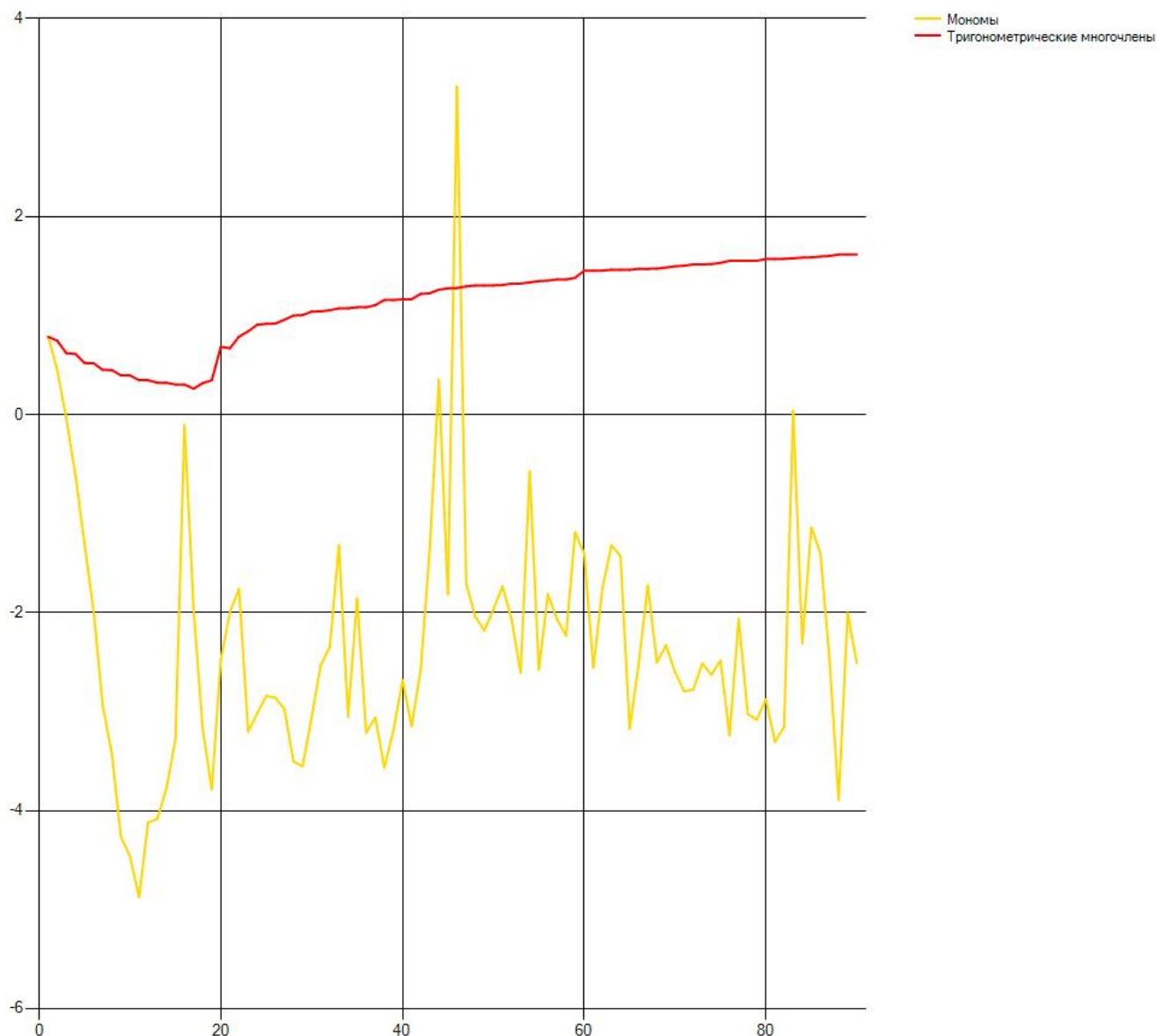


Рис. 6: Погрешность аппроксимации функции $y = e^{x/|x|+1}$, $x \in [0; 5]$ разными системами при разном количестве функций в системах (при менее точном интегрировании)

Вывод: увеличение точности подсчёта интегралов в СЛАУ «сглаживает» неустойчивость, но полностью не устраняет её. Поэтому при заполнении СЛАУ следует добиться некоторого баланса между точностью и временем вычислений; на мой взгляд, квадратуры Гаусса-Кронрода при 31, 51, 61 точках наиболее оптимальны в этом случае. Считая проблему интегрирования вполне решённой, попытаемся решить другие проблемы.

Посмотрим, «сгладится» ли аппроксимация, если попытаться решать СЛАУ точнее. Метод Холleckкого в данном случае, как оказывается, не поможет, поскольку плохая обусловленность матрицы

Грама и ошибки округления делают её не положительно определённой при числе функций около 15 (в этом случае в методе Холецкого появляются корни из отрицательных чисел). Чтобы в этом убедиться, посчитаем определители диагональных миноров матрицы Гильберта для отрезка $[0, 2]$ и числа функций 20; результаты оформлены в таблицу Рис. 7.

Размерность матрицы	Значение определителя
1	1
2	0,333333333
3	0,02962963
4	0,000677249
5	3,93142E-06
6	5,76309E-09
7	2,12681E-12
8	1,97225E-16
9	4,59033E-21
10	2,6807E-26
11	3,99924E-32
12	2,17554E-38
13	1,04472E-44
14	-9,22997E-51
15	-1,547E-56
16	1,06101E-61
17	6,81826E-67
18	2,39344E-71
19	1,79869E-75

Рис. 7: Зависимость определителя матрицы Гильберта от размерности матрицы ($A_{i,j} = \int_h x^{i+j-2} dx, h = [0, 2], i, j = 1, \dots, n$, A – матрица Гильберта, n – её размерность)

Если взять другой отрезок, результаты изменятся по модулю, но всё равно сохраняют тенденцию (Рис. 8). Описанную тенденцию можно изобразить визуальнo (Рис. 9).

Поскольку метод Холецкого использовать нельзя, но метод Гаусса сам по себе является вполне эффективным, попробуем улучшить решение СЛАУ, используя результат метода Гаусса как начальное приближение в методе наискорейшего спуска, выведенного специально для задач минимизации квадратичных функционалов; напомним, что этот метод является итерационным и может быть выражен формулой

$$x^{k+1} = x^k - \frac{(r^k, r^k)}{(Ar^k, r^k)} r^k, x^0 - \text{любой вектор} \quad (10)$$

где $r^k = Ax^k - b$, b – свободный вектор СЛАУ, A – матрица СЛАУ; для ускорения сходимости вектор x^0 желательно брать достаточно близким к истинному решению, что мы и сделаем, взяв за x_0 результат решения СЛАУ методом Гаусса. Как показывают тесты, этот метод для конкретно задачи аппроксимации сходится не всегда, что, возможно, обусловлено нарушением положительной определённости матрицы Грамма при большой размерности; по этой причине количество итераций в методе будет ограничено заранее заданным значением и если после очередной итерации невязка решения СЛАУ возрастёт, метод вернёт прежнее значение и завершится принудительно. Для удобства назовём этот метод *гибридным*. Аппроксимируя, как и предыдущем случае, функцию $y = e^{x/|x|+1}$ на отрезке $[0, 5]$, видим (Рис. 10), что неустойчивость слегка сгладилась (для системы экспонент).

Однако увеличение числа функций по-прежнему приводит к сильным скачкам (Рис. 11).

Размерность матрицы	Значение определителя
1	1
2	2,083333333
3	7,233796296
4	40,36716683
5	357,5606473
6	4998,68755
7	109953,4216
8	3798420,064
9	205833964,7
10	17482089317
11	2,32584E+12
12	4,84689E+14
13	1,59387E+17
14	8,97577E+19
15	8,51894E+22
16	-1,34141E+27
17	-1,76506E+32
18	-9,22986E+36
19	3,23827E+44

Рис. 8: Аналогичная таблица при $h = [-1, 4]$

Значения функции sign от определителей главных миноров матрицы Гильберта для разной размерности миноров

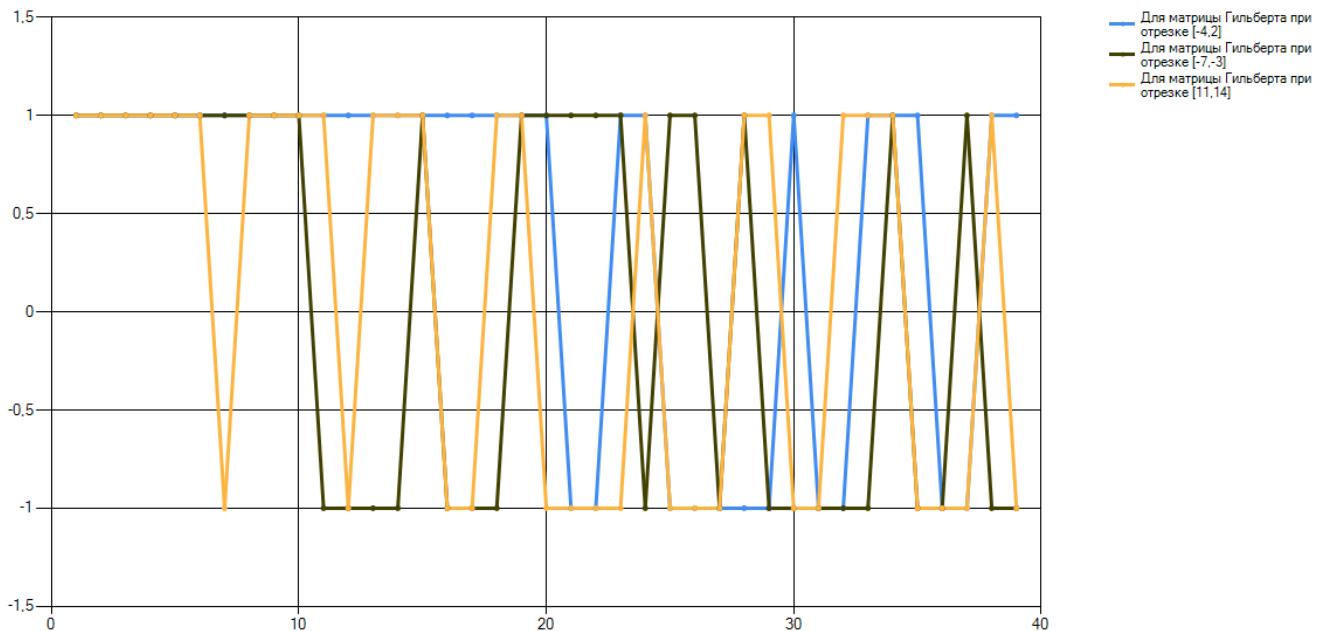


Рис. 9: Матрица Грама при большой размерности численно перестаёт быть положительно определённой

Это связано с тем, что для систем большой размерности невязка при использовании метода наискорейшего спуска начинает возрастать уже с 1-3 шага, либо изначально очень велика и не меняется,

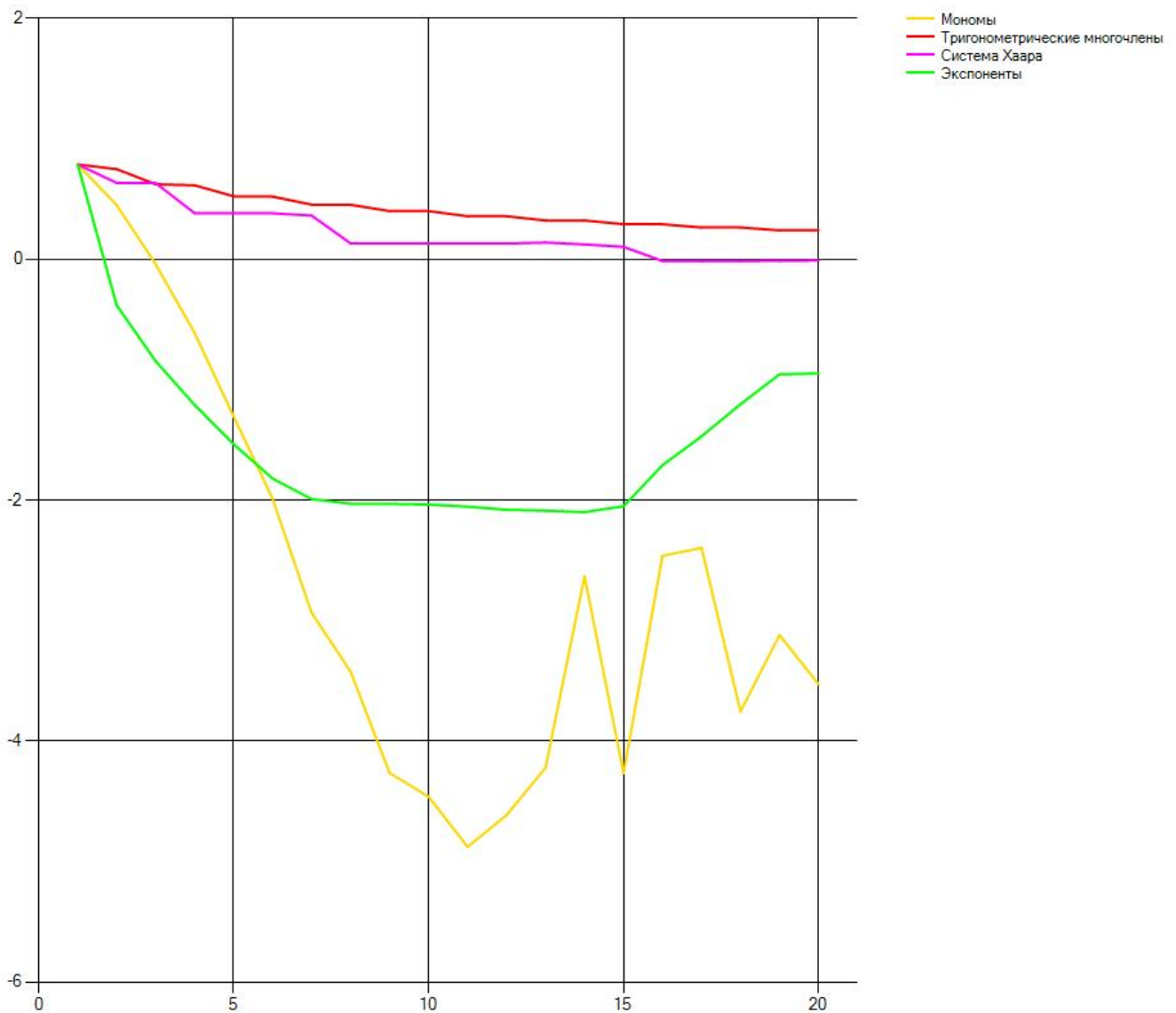


Рис. 10: Погрешность аппроксимации функции $y = e^{x/|x|+1}$, $x \in [0; 5]$ разными системами при разном количестве функций в системах (при гибридном способе решения СЛАУ)

так что польза от добавления наискорейшего спуска сомнительна.

Существует алгоритм уточнения решения ([7]), выражающийся формулой

$$x^{s+1} = x^s + A^{-1}(b - Ax^s), s \geq 0, x^0 - \text{любой вектор}$$

но и он не приносит заметной пользы, поскольку матрица Грама плохо обусловлена и к тому же численно не положительно определённая.

Дальнейшие попытки улучшить решение СЛАУ заведомо бесполезны, поскольку погрешности вычислений и квадратур вместе с плохой обусловленностью матриц не дадут достигнуть более точного решения, а более точное решение само по себе будет лишь приближённым решением задачи аппроксимации. Действительно, следующие графики (Рис. 12, Рис. 13) показывают, что между погрешностью аппроксимации и погрешностью решения СЛАУ корреляция не обязательна, поскольку даже при больших невязках погрешность может оставаться малой и далеко не всегда рост невязок означает рост погрешности. Кроме того, графики показывают, как быстро растут невязки системы,

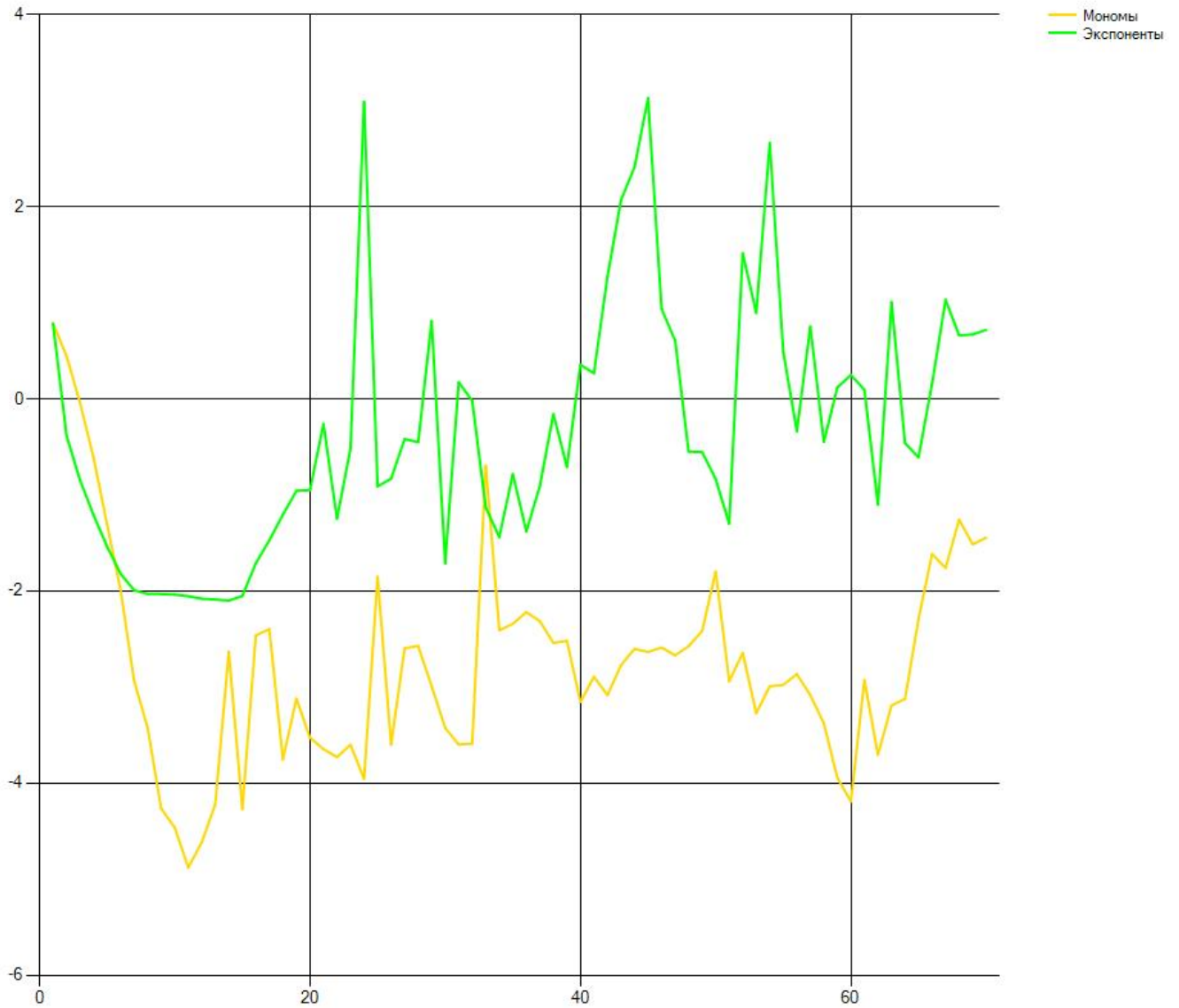


Рис. 11: Погрешность аппроксимации функции $y = e^{x/|x|+1}$, $x \in [0; 5]$ разными системами при разном количестве функций в системах (гибридный метод не решает проблему неустойчивости)

несмотря на все старания решать её точно.

Следовательно, даже весьма неточное решение СЛАУ способно обеспечить относительно хорошую аппроксимацию. Вдобавок можно сделать вывод, что даже при максимально точном решении СЛАУ в пределах машинных возможностей минимизация функции

$$F(c) = \left\| \varphi - \sum_{m=1}^N c_m \alpha_m \right\|_{L_2(Q)} \quad (11)$$

даёт сбой, поскольку результаты тестирования показывают, что при многих значениях $M < N$ имеет место неравенство

$$\left\| \varphi - \sum_{m=1}^M c_m \alpha_m \right\|_{L_2(Q)} < \left\| \varphi - \sum_{m=1}^{M+1} c_m \alpha_m \right\|_{L_2(Q)}, \quad (12)$$

Логарифмическая зависимость аппроксимации функции $\sin(x/2)*\cos(x)$ системой мономов на отрезке $[-4,2]$ и невязок при решении СЛАУ для разного количества функций (точки с нулевыми невязками не выводятся)

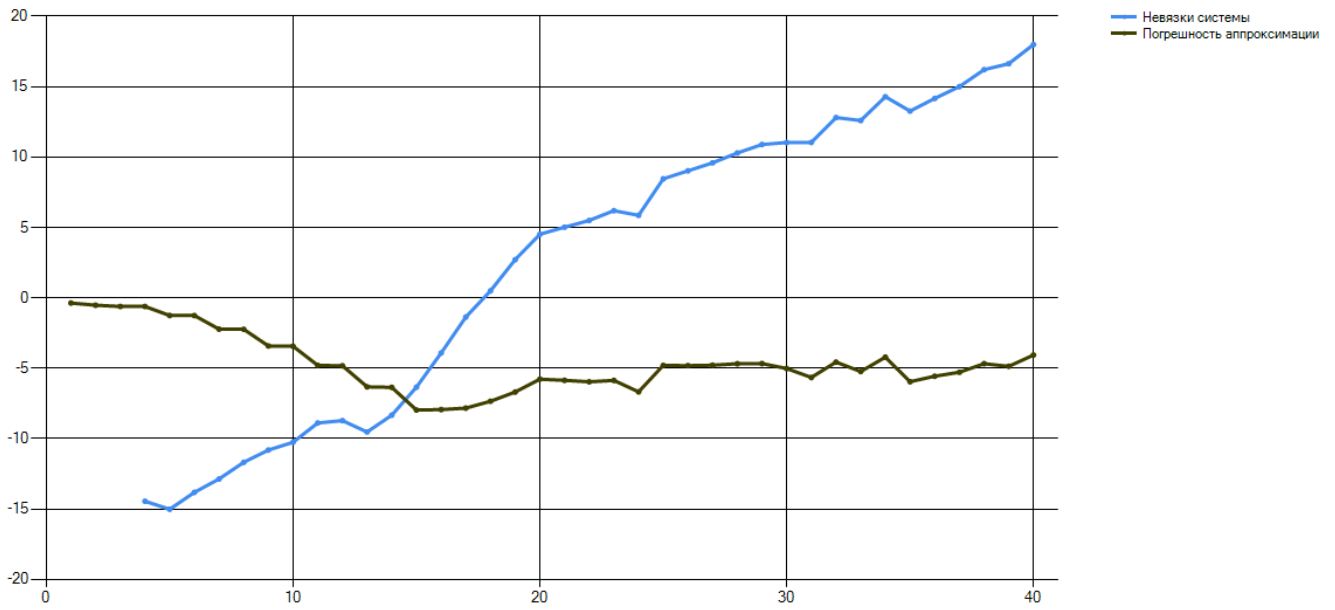


Рис. 12: Между невязкой и погрешностью нет заметной связи

Логарифмическая зависимость аппроксимации функции $\cos(2x)*\exp(x)$ системой экспонент на отрезке $[0,2]$ и невязок при решении СЛАУ для разного количества функций (точки с нулевыми невязками не выводятся)

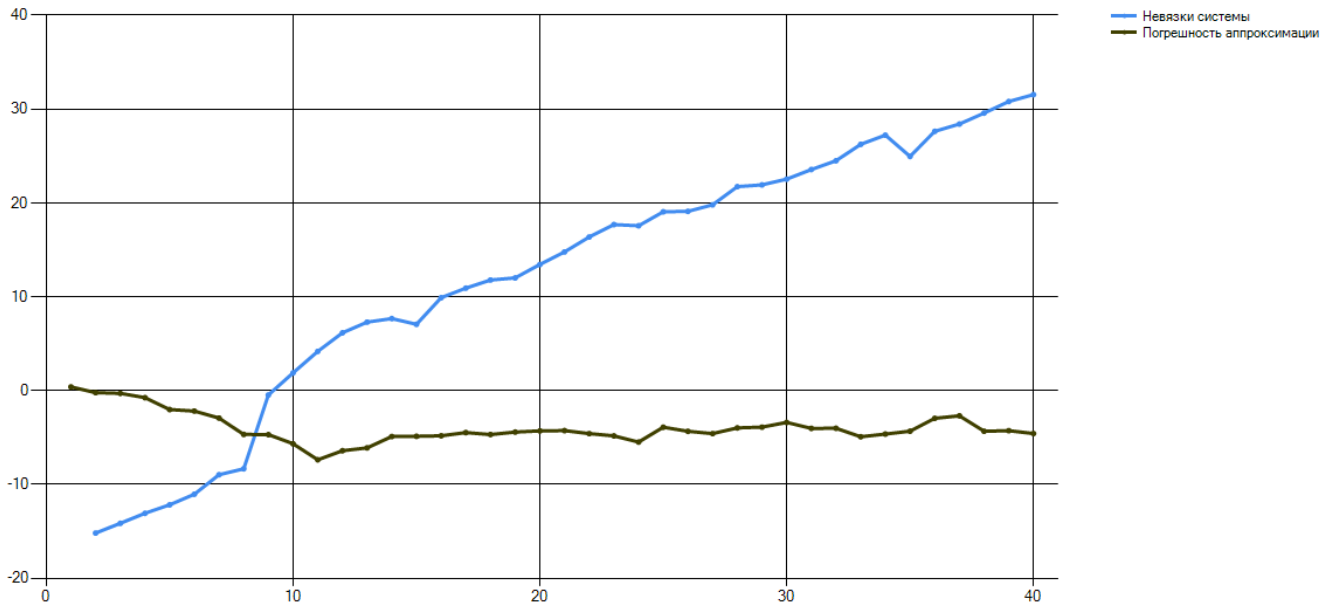


Рис. 13: Рост невязки не ведёт к увеличению погрешности

хотя вполне очевиден другой набор коэффициентов $c_m, m = 1, 2, \dots, M + 1$, обеспечивающий ап-

проксимацию лучше того набора, который находится программой, а именно:

$$\left\| \varphi - \sum_{m=1}^M c_m \alpha_m - 0 \times \alpha_{M+1} \right\|_{L_2(Q)} = \left\| \varphi - \sum_{m=1}^M c_m \alpha_m \right\|_{L_2(Q)} < \left\| \varphi - \sum_{m=1}^{M+1} c_m \alpha_m \right\|_{L_2(\partial Q)}. \quad (13)$$

Значит, если при максимально возможном точном решении СЛАУ неустойчивость остаётся, то единственным оставшимся выходом для её устранения может быть только варьирование получившихся коэффициентов в таком направлении, чтобы аппроксимация стала минимальной. Теория при этом говорит, что получившаяся последовательность приближений должна, по крайней мере, не возрастать.

Попробуем провести манипуляции с приближённым решением СЛАУ с целью приблизить его не к истинному решению СЛАУ, а к истинному решению задачи аппроксимации, используя метод покоординатной минимизации.

6 Метод покоординатной минимизации и его возможная ограниченность вследствие ошибок округления

Допустим, гибридным методом был найден набор чисел $c_m, m = 1, 2, \dots, N$, являющийся максимально точным решением получившейся СЛАУ; однако, в самом деле, этот набор не обязан обеспечивать максимально хорошую аппроксимацию не только из-за неизбежных неточностей при вычислении скалярных произведений (интегралов), но и из-за плохой обусловленности матрицы Грамма и ошибок округления. Повторюсь, что решение системы

$$\begin{pmatrix} (\alpha_1, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_1, \alpha_N)_{L_2(Q)} \\ \vdots & \ddots & \vdots \\ (\alpha_N, \alpha_1)_{L_2(Q)} & \cdots & (\alpha_N, \alpha_N)_{L_2(Q)} \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_N \end{pmatrix} = \begin{pmatrix} (\varphi, \alpha_1)_{L_2(Q)} \\ \vdots \\ (\varphi, \alpha_N)_{L_2(Q)} \end{pmatrix} \quad (14)$$

является максимально точным, но сама система лишь приближённо является той, какой должна быть, вдобавок невязка такой системы существенно растёт с увеличением размерности. Отсюда можно сделать выводы, что некоторая корректировка набора $c_m, m = 1, 2, \dots, N$ оказывается не только полезной, но и обязательной, если при аппроксимации требуется добиться максимальной точности и устойчивости; это понятно и из вывода, сделанного в предыдущем пункте. Можно осуществить такую корректировку при помощи метода покоординатного спуска, который состоит в следующем ([5, 6]).

Найденный (любым методом) вектор (c_1, c_2, \dots, c_N) берётся за начальное приближение; после этого для каждого (либо для некоторых) $c_k, 1 \leq k \leq N$ последовательно выполняется задача минимизации функции одной переменной

$$\begin{aligned} F(c_k) &= \left\| \varphi - \sum_{m=1, m \neq k}^N c_m \alpha_m - c_k \alpha_k \right\|_{L_2(\partial Q)} = \\ &= \left(\varphi - \sum_{m=1, m \neq k}^N c_m \alpha_m - c_k \alpha_k, \varphi - \sum_{m=1, m \neq k}^N c_m \alpha_m - c_k \alpha_k \right)_{L_2(\partial Q)}^{\frac{1}{2}} = \\ &= \left((\varphi, \varphi) - 2 \sum_{m=1, m \neq k}^N c_m (\varphi, \alpha_m) + \sum_{n=1, n \neq k}^N c_n \sum_{m=1, m \neq k}^N c_m (\alpha_n, \alpha_m) + \right. \\ &\quad \left. 2 \sum_{m=1, m \neq k}^N c_m c_k (\alpha_k, \alpha_m) - 2c_k (\alpha_k, \varphi) + c_k^2 (\alpha_k, \alpha_k) \right)^{\frac{1}{2}}, \quad (15) \end{aligned}$$

при этом полученный в конце новый вектор (c_1, c_2, \dots, c_N) теоретически гарантировано обеспечит лучшую аппроксимацию, а потому и устойчивость (устранив «скачки»).

Итак, реализуем задачу минимизации функции

$$F(c_k)^2 = (\varphi, \varphi) - 2 \sum_{m=1, m \neq k}^N c_m (\varphi, \alpha_m) + \sum_{n=1, n \neq k}^N c_n \sum_{m=1, m \neq k}^N c_m (\alpha_n, \alpha_m) + 2 \sum_{m=1, m \neq k}^N c_m c_k (\alpha_k, \alpha_m) - 2c_k (\alpha_k, \varphi) + c_k^2 (\alpha_k, \alpha_k) \quad (16)$$

(возведение в квадрат не повлияет на решения, поскольку функция принимает исключительно неотрицательные значения). Стационарные точки:

$$D_{c_k} F(c_k) = 0 \leftrightarrow 2 \sum_{m=1, m \neq k}^N c_m (\alpha_k, \alpha_m) - 2(\alpha_k, \varphi) + 2c_k (\alpha_k, \alpha_k) = 0 \rightarrow c_k = \frac{(\alpha_k, \varphi) - \sum_{m=1, m \neq k}^N c_m (\alpha_k, \alpha_m)}{(\alpha_k, \alpha_k)}. \quad (17)$$

Очевидно, что найденная стационарная точка является точкой минимума, поскольку при переходе через неё производная функции меняет знак с отрицательного на положительный

$$D_{c_k} F(c_k \pm \varepsilon) = 2 \sum_{m=1, m \neq k}^N c_m (\alpha_k, \alpha_m) - 2(\alpha_k, \varphi) + 2(c_k \pm \varepsilon) \times (\alpha_k, \alpha_k) = \pm 2\varepsilon (\alpha_k, \alpha_k) = \pm \omega, \omega > 0. \quad (18)$$

При тестировании метода покоординатного спуска выяснилось, что он хорошо дополняет предыдущие методы, но его использование вовсе необязательно минимизирует функцию, если в точке начального приближения её значения уже достаточно близки к нулю. Иными словами, если за начальное приближение взять результат работы какого-либо предыдущего метода (то есть вектор (c_1, c_2, \dots, c_N) такой, что $0 < F(c) \leq \varepsilon$), то для получающегося вектора $\tau = (\tau_1, \tau_2, \dots, \tau_N)$ вовсе не обязательно будет выполняться $0 < F(\tau) \leq F(c)$, даже после многократного применения метода покоординатной минимизации; иногда, впрочем, случается, что $F(\tau) \leq F(c)$, но при этом не никакой речи о каком-то монотонном улучшении качества аппроксимации не идёт, откуда и не получится устойчивости. Покажем основные результаты; во всех следующих примерах минимизация проводится по всем элементам вектора c по 10 раз.

Неустойчивость (Рис. 14).

Погрешность и невязка в зависимости от числа функций (Рис. 13).

В этом же примере при любом числе функций многократная минимизация стабильно действует, хоть и очень медленно (Рис. 16).

Как видно из предыдущих графиков, уменьшение погрешности аппроксимации не обязательно уменьшает невязку. В данном случае погрешность стабильно уменьшается, однако часто она может возрастать при использовании покоординатной минимизации при решении достаточно сложных задач аппроксимации. В следующем примере происходит аппроксимация функции двух переменных, заданной на окружности, по системе логарифмов (базисных потенциалов)

$$\alpha_m(y) = \ln \frac{1}{|z^m - y|} = \ln \frac{1}{\sqrt{\sum_{n=1}^2 (z_n^m - y_n)^2}}, \quad z^m \in Q^+, \quad m = 1, 2, \dots, \quad y \in \partial Q, \quad (19)$$

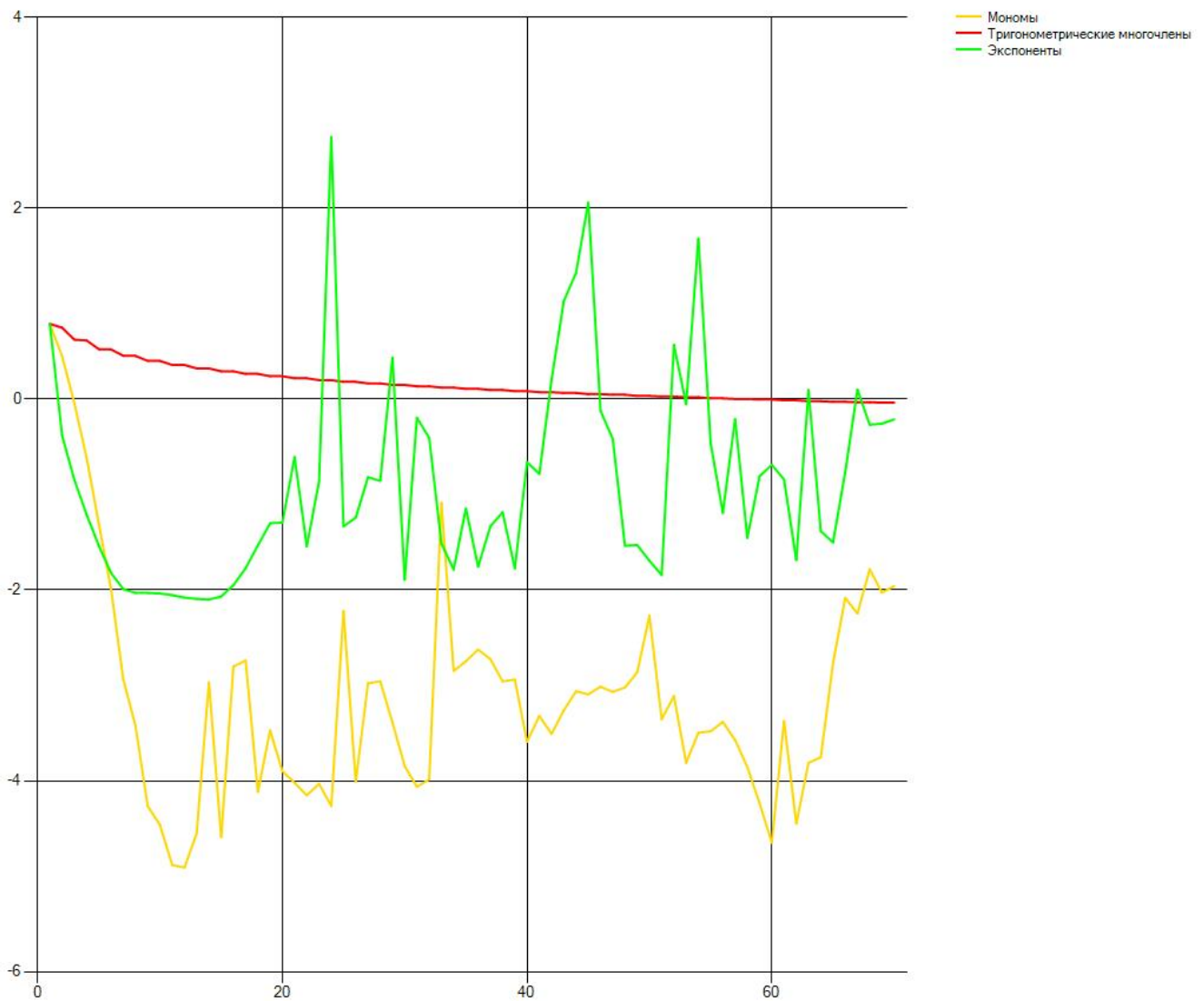


Рис. 14: Неустойчивая аппроксимация методом покоординатной минимизации

где ∂Q – граница круга (окружность, на которой ведётся аппроксимация), Q^+ – внешность круга, z^m – некоторый набор точек, удовлетворяющий определённым условиям. В этом случае при достаточно большом числе функций многократное использование покоординатной минимизации не только не обеспечивает монотонное убывание, но и само убывание не гарантирует (Рис. 17).

Тесты, подобные предыдущему, привели к предположению, что метод покоординатной минимизации не способен гарантированно уменьшать погрешность аппроксимации ниже некоторого фиксированного значения, определённого условиями задачи. При простейшей аппроксимации функции на отрезке погрешность в подавляющем большинстве тестов убывала гарантированно, хоть и медленно; может быть, вероятность неубывания погрешности тем больше, чем меньше невязка системы, что видно из следующего случая (Рис. 18).

Зафиксируем промежуточные выводы:

1. Решение задачи аппроксимации классическими алгоритмами получается неустойчивым вследствие погрешностей квадратур, плохой обусловленности матриц Грама и ошибок округления.
2. Увеличение точности интегрирования безусловно «сглаживает» неустойчивость, однако счи-

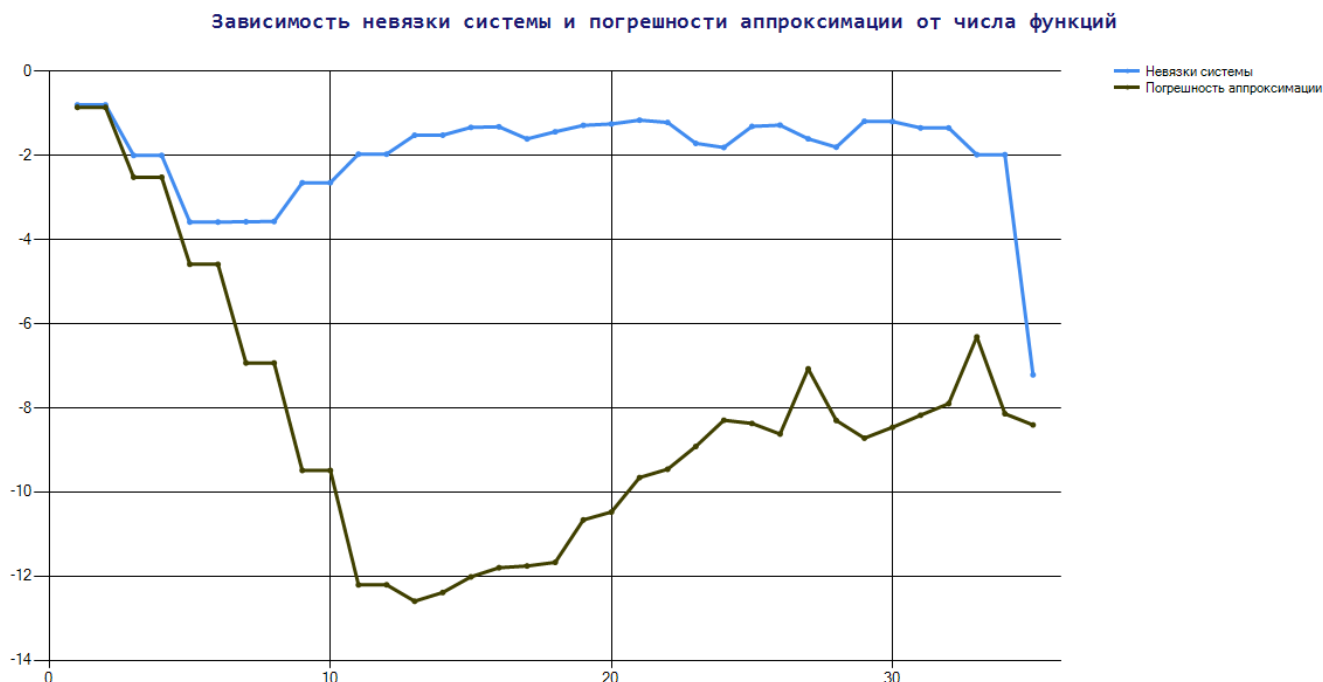


Рис. 15: Зависимость невязки системы и погрешности аппроксимации от числа функций, используется система мономов на отрезке $[-1; 1]$, система решается гибридным методом с последующей покоординатной минимизацией

```

Точность аппроксимации при числе функций 39 =
до использования покоординатной минимизации: 1,35523736382345E-05 (4,2645064636524E+16) - в скобках указана невязка системы
после использования покоординатной минимизации 1 раз: 1,19133029471747E-05 (7,02125342157765E+16)
после использования покоординатной минимизации 2 раз: 6,87551060831899E-06 (2,44301396626311E+16)
после использования покоординатной минимизации 3 раз: 5,96805942869647E-06 (6,43285725702141E+15)
после использования покоординатной минимизации 4 раз: 5,8869908965915E-06 (1,37042921103968E+15)
после использования покоординатной минимизации 5 раз: 5,87197969610934E-06 (4,946020159836E+15)
после использования покоординатной минимизации 6 раз: 5,82125045970412E-06 (6,62837270528886E+15)
после использования покоординатной минимизации 7 раз: 5,7386391503561E-06 (7,40678285855093E+15)
после использования покоординатной минимизации 8 раз: 5,63708064940082E-06 (7,73328275218634E+15)
после использования покоординатной минимизации 9 раз: 5,52578619014006E-06 (7,82367490500914E+15)
после использования покоординатной минимизации 10 раз: 5,41033493665787E-06 (7,78533091640299E+15)
Точность аппроксимации при числе функций 40 =
до использования покоординатной минимизации: 8,56396041143931E-05 (9,70228753727443E+17) - в скобках указана невязка системы
после использования покоординатной минимизации 1 раз: 7,65109469970922E-05 (1,81810882664766E+18)
после использования покоординатной минимизации 2 раз: 4,22286701917703E-05 (5,74315391110152E+17)
после использования покоординатной минимизации 3 раз: 3,69853465216583E-05 (1,00155029104024E+17)
после использования покоординатной минимизации 4 раз: 3,67161099588483E-05 (9,51764977736196E+16)
после использования покоординатной минимизации 5 раз: 3,64797004507684E-05 (1,77573445513704E+17)
после использования покоординатной минимизации 6 раз: 3,58695539168662E-05 (2,10994215025247E+17)
после использования покоординатной минимизации 7 раз: 3,50296041961064E-05 (2,2252657026658E+17)
после использования покоординатной минимизации 8 раз: 3,40876651935968E-05 (2,2326549915482E+17)
после использования покоординатной минимизации 9 раз: 3,31162541216469E-05 (2,19509832955796E+17)
после использования покоординатной минимизации 10 раз: 3,21523966379465E-05 (2,13564203370026E+17)

```

Рис. 16: Многократная минимизация медленно уточняет решение

тать интегралы идеально точно нельзя, поэтому в подсчёте интегралов требуется достигнуть компромисса между точностью и быстродействием, а избавляться от неустойчивости требуется независимо от точности интегрирования.

- Классический и модифицированный методы Гаусса не способны точно решать плохо обусловленные СЛАУ большой размерности. Поскольку при большой (от 12-15) размерности матри-

```

Точность аппроксимации при числе функций 145 =
до использования покоординатной минимизации: 2.90477e-007
после использования покоординатной минимизации 1 раз: 3.73571e-007
после использования покоординатной минимизации 2 раз: 1.97405e-007
после использования покоординатной минимизации 3 раз: 3.87e-007
после использования покоординатной минимизации 4 раз: 5.81909e-008
после использования покоординатной минимизации 5 раз: 1.49384e-007
Точность аппроксимации при числе функций 146 =
до использования покоординатной минимизации: 5.9614e-007
после использования покоординатной минимизации 1 раз: 2.88656e-007
после использования покоординатной минимизации 2 раз: 3.47072e-007
после использования покоординатной минимизации 3 раз: 4.78116e-007
после использования покоординатной минимизации 4 раз: 6.6989e-007
после использования покоординатной минимизации 5 раз: 4.97351e-007
Точность аппроксимации при числе функций 147 =
до использования покоординатной минимизации: 6.40923e-008
после использования покоординатной минимизации 1 раз: 1.90828e-007
после использования покоординатной минимизации 2 раз: 1.666e-007
после использования покоординатной минимизации 3 раз: 1.07712e-007
после использования покоординатной минимизации 4 раз: 4.59949e-007
после использования покоординатной минимизации 5 раз: 4.0604e-007
Точность аппроксимации при числе функций 148 =
до использования покоординатной минимизации: 4.23898e-007
после использования покоординатной минимизации 1 раз: 2.75973e-007
после использования покоординатной минимизации 2 раз: 1.33488e-007

```

Рис. 17: Многократная минимизация не обязательно уточняет решение

после использования покоординатной минимизации 24 раз:	1,10875005200797E-11	(2,19471598394291E-11)
после использования покоординатной минимизации 25 раз:	1,1086696395135E-11	(2,91756980533271E-11)
после использования покоординатной минимизации 26 раз:	1,10867648616327E-11	(2,91189570007656E-11)
после использования покоординатной минимизации 27 раз:	1,10864128684942E-11	(2,91767953545428E-11)
после использования покоординатной минимизации 28 раз:	1,10855135287539E-11	(2,91082746957955E-11)
после использования покоординатной минимизации 29 раз:	1,10856374198637E-11	(3,64373083558124E-11)
после использования покоординатной минимизации 30 раз:	1,10858390417385E-11	(4,36966037338271E-11)
после использования покоординатной минимизации 31 раз:	1,10854233039735E-11	(3,63828188535278E-11)
после использования покоординатной минимизации 32 раз:	1,10853266772754E-11	(3,63918659864094E-11)
после использования покоординатной минимизации 33 раз:	1,10849232685034E-11	(2,91756885358485E-11)
после использования покоординатной минимизации 34 раз:	1,10849228786053E-11	(4,36965921798159E-11)
после использования покоординатной минимизации 35 раз:	1,10846281864235E-11	(2,91189394458092E-11)
после использования покоординатной минимизации 36 раз:	1,10843372594392E-11	(3,6383351712285E-11)
после использования покоординатной минимизации 37 раз:	1,10841954098308E-11	(3,6392756113562E-11)
после использования покоординатной минимизации 38 раз:	1,1084344368474E-11	(3,6436775766525E-11)
после использования покоординатной минимизации 39 раз:	1,10832670296505E-11	(3,63918763912944E-11)
после использования покоординатной минимизации 40 раз:	1,10839738121078E-11	(4,37036833129842E-11)
после использования покоординатной минимизации 41 раз:	1,10833327432641E-11	(4,38076779919092E-11)
после использования покоординатной минимизации 42 раз:	1,10830398187004E-11	(3,63918682408014E-11)
после использования покоординатной минимизации 43 раз:	1,10828923541243E-11	(3,64287854731206E-11)
после использования покоординатной минимизации 44 раз:	1,10823638295385E-11	(4,37036811469583E-11)
после использования покоординатной минимизации 45 раз:	1,10817428565572E-11	(3,64287936153539E-11)
после использования покоординатной минимизации 46 раз:	1,10826542965721E-11	(4,38076657468984E-11)
после использования покоординатной минимизации 47 раз:	1,10820116632932E-11	(2,9118926442131E-11)
после использования покоординатной минимизации 48 раз:	1,10815955715934E-11	(3,64287854731206E-11)
после использования покоординатной минимизации 49 раз:	1,10815760523815E-11	(4,38076651706625E-11)
после использования покоординатной минимизации 50 раз:	1,10817150573194E-11	(3,63918687610457E-11)

Рис. 18: Негарантированное убывание погрешности при многократном использовании метода покоординатной минимизации для аппроксимации функции $y = x^3 - x + 4$ системой мономов на отрезке от 0 до 5

ца Грама перестаёт быть численно положительно определённой, метод Холецкого на большой размерности работать не может, а метод наискорейшего спуска чаще всего расходится. Решая СЛАУ, можно найти лишь приближённое решение, которое придётся как-нибудь модифицировать с целью уменьшить погрешность аппроксимации, а не невязку.

4. Метод покоординатной минимизации почти гарантирует убывание погрешности аппроксимации, пусть медленное. Его можно использовать для улучшения аппроксимации при конкретном

числе функций, но устойчивости от также не гарантирует. Возможно, устойчивый алгоритм не может не быть многошаговым.

7 Многошаговый алгоритм аппроксимации, гарантирующий устойчивость

Модификация и слияние нескольких описанных ранее методов, а также учёт некоторых замечаний, привели к созданию устойчивого многошагового алгоритма решения задачи аппроксимации; он никак не привязан к какой-либо системе функций и должен эффективно работать для любой линейно независимой системы. Назовём этот метод *ультра-гибридным*. Вот в чём он заключается.

Допустим, требуется минимизировать функционал

$$F(c^M) = \left\| \varphi - \sum_{m=1}^M c_m \alpha_m \right\|_{L_2(\partial Q)}, \quad (20)$$

причём его минимизация должна удовлетворять условию

$$F(c^{M-1}) \geq F(c^M) \geq F(c^{M+1}), \quad M \geq 2. \quad (21)$$

Такую минимизацию можно осуществить по индукции:

Если $M = 1$, решение очевидно: $c_1 = (\alpha_1, \varphi) / (\alpha_1, \alpha_1)$. В противном случае требуется искать решения для c^2, c^3, \dots, c^{M-1} последовательно; допустим, они найдены, тогда:

Шаг 1 (СПИДГАУСС). Найти вектор $c^M = (c_1, c_2, \dots, c_M)$, решив известную СЛАУ описанным ранее гибридным методом (либо любым другим).

Шаг 2 (МИНИМАКА СПИДГАУССА). Если $F(c^{M-1}) < F(c^M)$, то применить ко всем компонентам c^M покоординатную минимизацию (возможно, несколько раз).

Шаг 3 (ПОЛНАЯ МИНИМАКА). Если снова $F(c^{M-1}) < F(c^M)$, то вектор c^M заменить на вектор $(c_1^{M-1}, c_2^{M-1}, \dots, c_{M-1}^{M-1}, 0)$ и провести покоординатную минимизацию по всем компонентам этого вектора.

Шаг 4 (МИНИМАКА НА КОНЦЕ). Если снова $F(c^{M-1}) < F(c^M)$, то вектор c^M снова заменить на вектор $(c_1^{M-1}, c_2^{M-1}, \dots, c_{M-1}^{M-1}, 0)$ и применить покоординатную минимизацию только для последнего элемента

$$c^M = \left(c_1^{M-1}, c_2^{M-1}, \dots, c_{M-1}^{M-1}, \frac{(\alpha_M, \varphi) - \sum_{m=1}^{M-1} c_m (\alpha_M, \alpha_m)}{(\alpha_M, \alpha_M)} \right). \quad (22)$$

Шаг 5. Если и в таком случае $F(c^{M-1}) < F(c^M)$, то

$$c^M = (c_1^{M-1}, c_2^{M-1}, \dots, c_{M-1}^{M-1}, 0) \rightarrow F(c^{M-1}) = F(c^M). \quad (23)$$

Аппроксимативные свойства этого метода, очевидно, не хуже, чем у гибридного; при этом устойчивость гарантирована. Практика показывает, что шаги 1-4 не являются равнозначными и в каждом конкретном случае любой из них может уменьшить погрешность аппроксимации там, где остальные не могут.

Следующие графики (Рис. 19-24) показывают, насколько метод устойчив; обратите внимание, что зачастую ультра-гибрид позволяет аппроксимировать не только устойчиво, но и точнее любого другого метода (кривая ультра-гибрида опускается ниже минимального значения для другого метода).

Ультра-гибрид исправляет в том числе неустойчивость аппроксимации по ортогональным функциям (Рис. 22-23).

А вот так метод «выглядит изнутри» (Рис. 24).

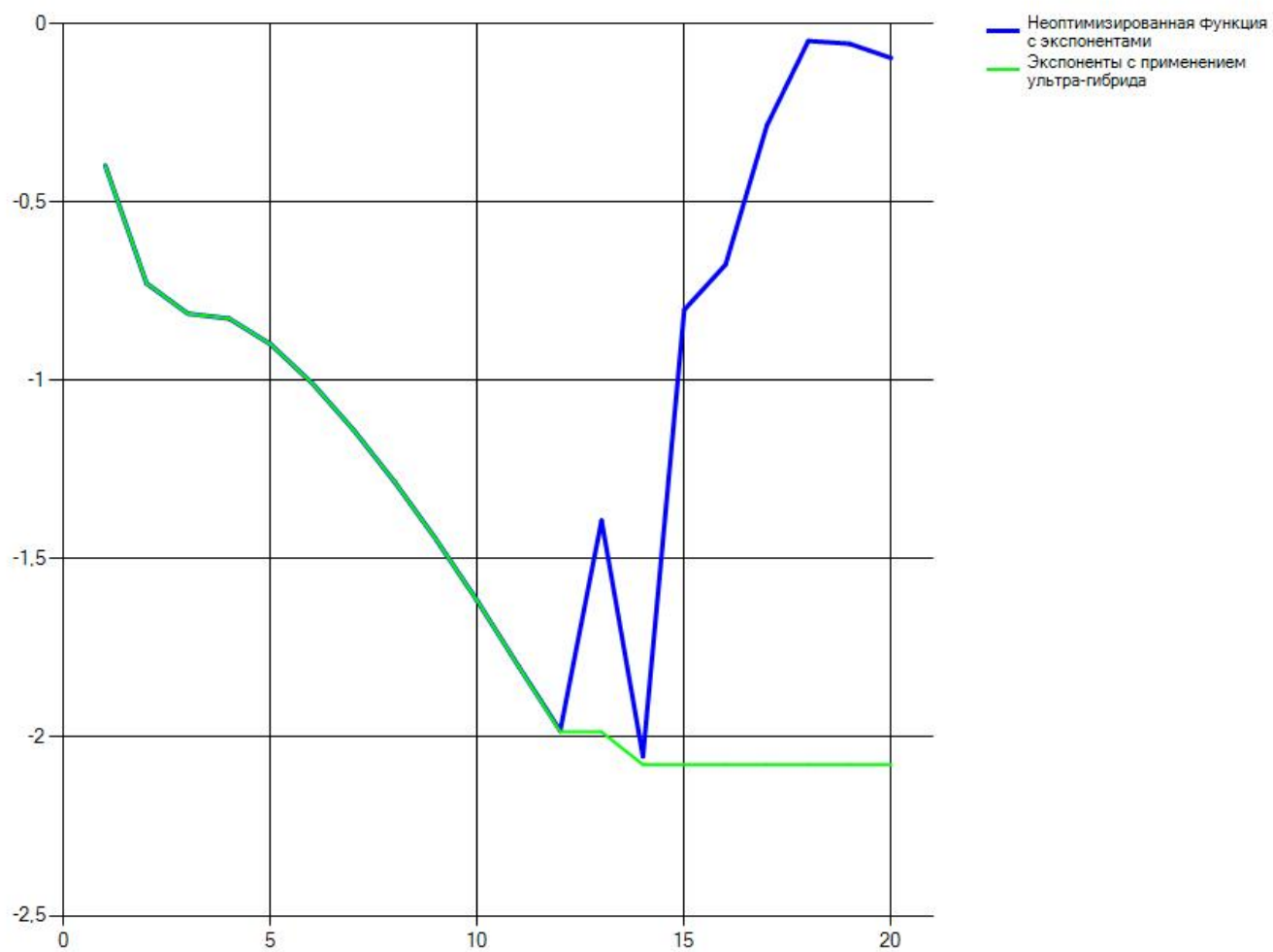


Рис. 19: Устойчивая аппроксимация экспонентами с помощью ультра-гибрида

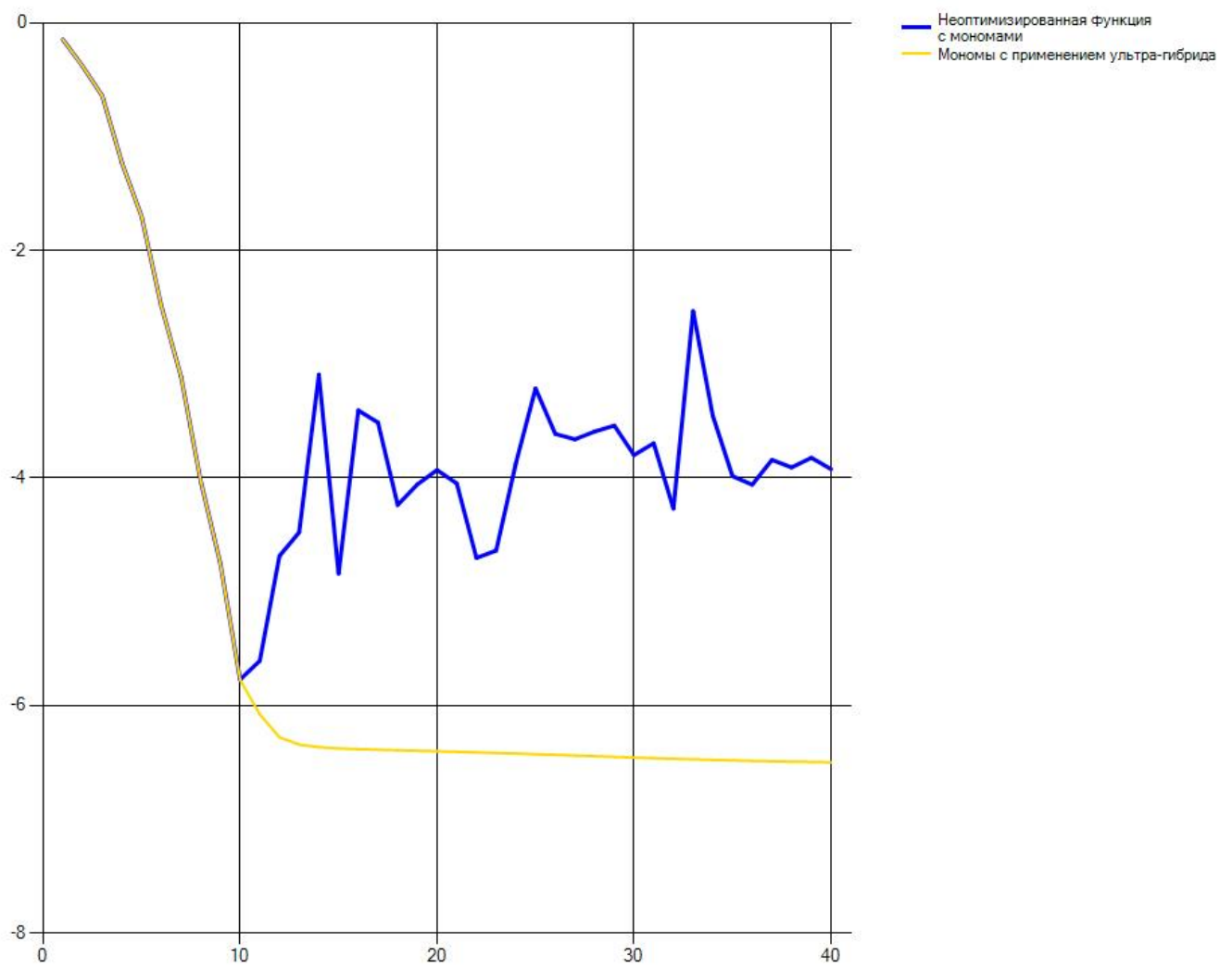


Рис. 20: Устойчивая аппроксимация мономами с помощью ультра-гибрида

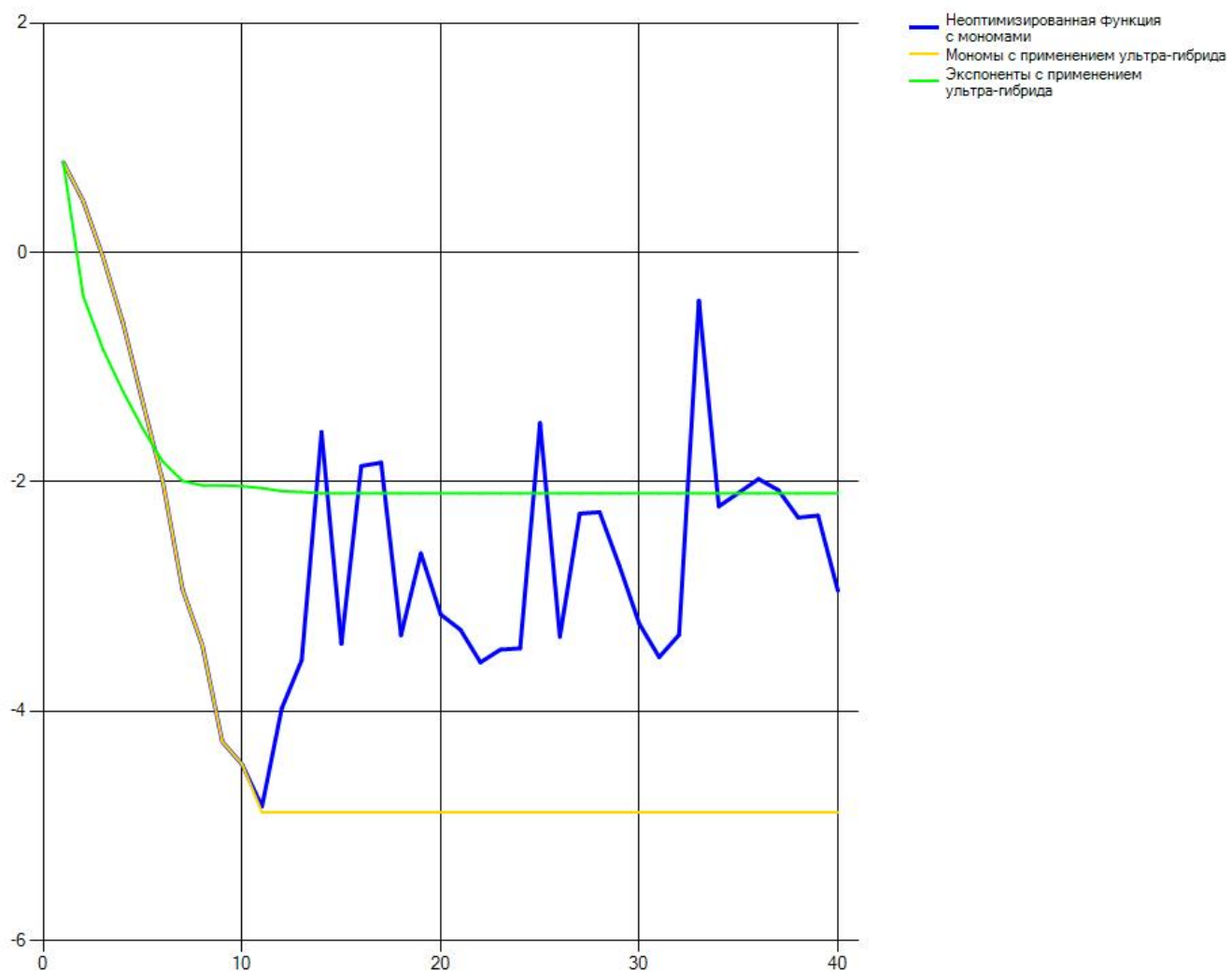


Рис. 21: Устойчивая аппроксимация экспонентами и мономами с помощью ультра-гибрида

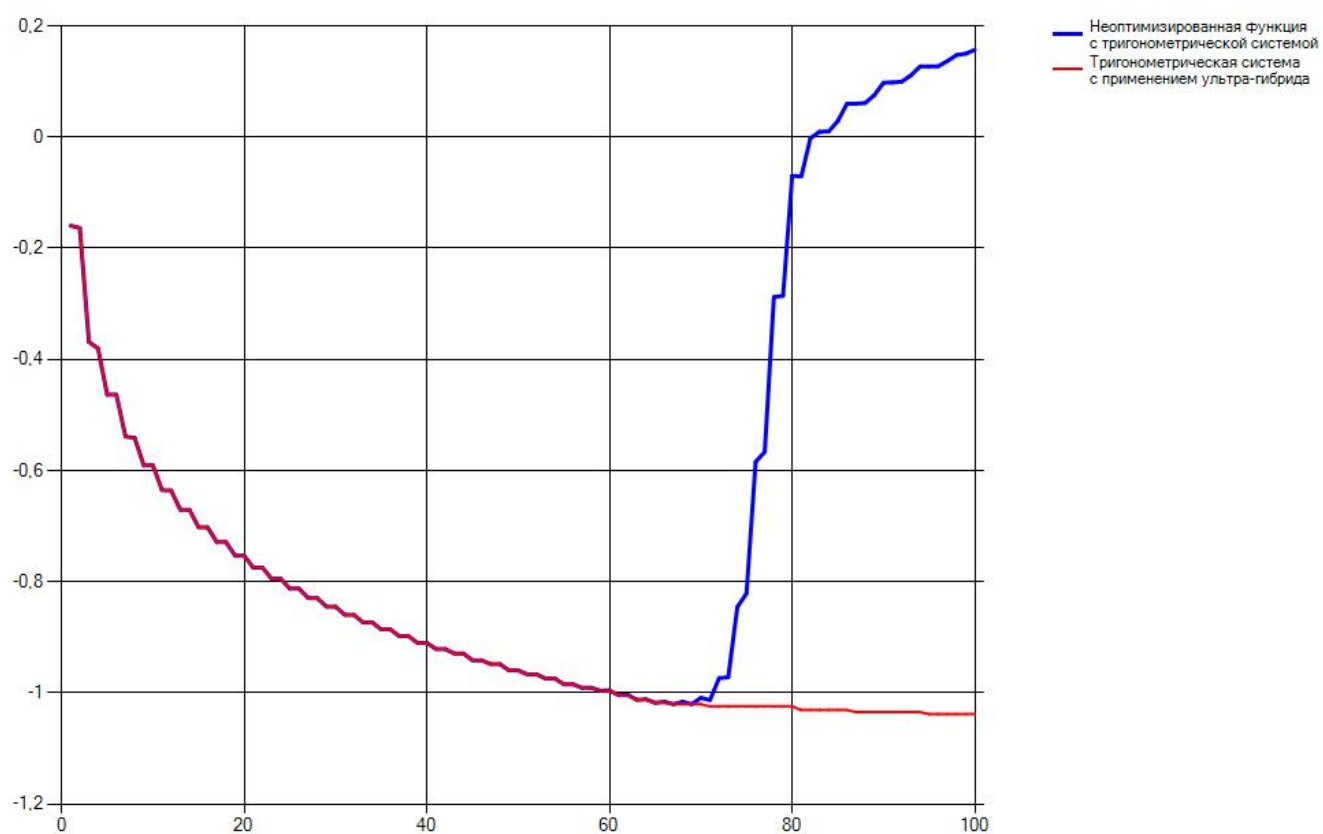


Рис. 22: Устойчивая аппроксимация системой синусов и косинусов с помощью ультра-гибрида

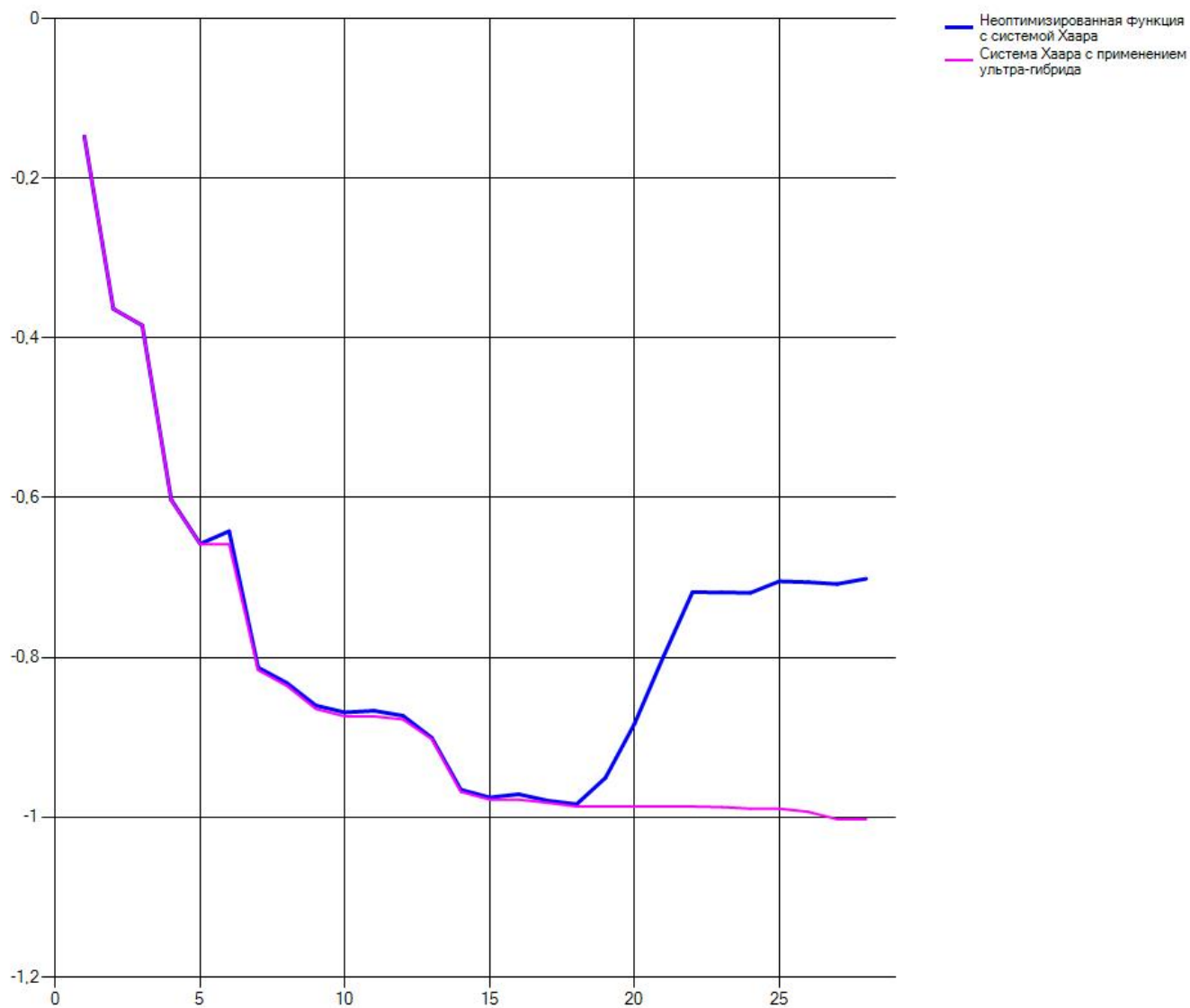


Рис. 23: Устойчивая аппроксимация системой Хаара с помощью ультра-гибрида

```

Погрешность уменьшена СПИДГАУССОМ на 35,4244330463701 %
Погрешность уменьшена СПИДГАУССОМ на 95,5434702632401 %
0,0394216901547653 < 0,0760986460258508 при t = 4 (до покоординатной минимизации результата СПИДГАУССА)
0,0394216901547653 < 1,28111285653612 при t = 4 (до полно покоординатной минимизации вектора c1 c2 ... 0)
0,0394216901547653 < 0,0395061869203051 при t = 4 (до полно покоординатной минимизации на конце)
0,0394216901547653 < 0,0422346685754631 при t = 5 (до покоординатной минимизации результата СПИДГАУССА)
0,0394216901547653 < 1,14135297391236 при t = 5 (до полно покоординатной минимизации вектора c1 c2 ... 0)
0,0394216901547653 < 0,0395170587255971 при t = 5 (до полно покоординатной минимизации на конце)
Погрешность уменьшена МИНИМАКОЙ НА КОНЦЕ на 0,131497424050403 %
Поток 0x1bb8 завершился с кодом 0 (0x0).
0,0393698516476946 < 0,0570449704756105 при t = 6 (до покоординатной минимизации результата СПИДГАУССА)
0,0393698516476946 < 1,19007985872472 при t = 6 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 0,393004344552288 %
0,0392151264202754 < 0,0668502173712445 при t = 7 (до покоординатной минимизации результата СПИДГАУССА)
0,0392151264202754 < 12,3108150611806 при t = 7 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 0,702497247103531 %
0,0389396412367248 < 0,151529195549245 при t = 8 (до покоординатной минимизации результата СПИДГАУССА)
0,0389396412367248 < 45,7967993569312 при t = 8 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 0,355606842916728 %
0,0388011692078798 < 0,826853010019592 при t = 9 (до покоординатной минимизации результата СПИДГАУССА)
0,0388011692078798 < 91,2374148462599 при t = 9 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 0,0241725790560827 %
0,0387917899645783 < 20,5506792008336 при t = 10 (до покоординатной минимизации результата СПИДГАУССА)
0,0387917899645783 < 53791,9843889955 при t = 10 (до полно покоординатной минимизации вектора c1 c2 ... 0)
0,0387917899645783 < 0,0388122771182666 при t = 10 (до полно покоординатной минимизации на конце)
0,0387917899645783 < 28,101157060125 при t = 11 (до покоординатной минимизации результата СПИДГАУССА)
0,0387917899645783 < 30313,8458322724 при t = 11 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 0,0306414514830782 %
0,0387799035970769 < 42,7612699230159 при t = 12 (до покоординатной минимизации результата СПИДГАУССА)
0,0387799035970769 < 22502,8354749216 при t = 12 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 0,226902127650211 %
0,0386919111707145 < 109,771876495148 при t = 13 (до покоординатной минимизации результата СПИДГАУССА)
0,0386919111707145 < 5838516,27918602 при t = 13 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 1,75725526923085 %
0,038011995522901 < 586,403434414773 при t = 14 (до покоординатной минимизации результата СПИДГАУССА)
0,038011995522901 < 92834881,2913163 при t = 14 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 1,03523015612276 %
0,0376184838823038 < 2788,45840834194 при t = 15 (до покоординатной минимизации результата СПИДГАУССА)
0,0376184838823038 < 97410369,2685901 при t = 15 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 1,57777610271359 %
0,0370249484334057 < 250,784949577033 при t = 16 (до покоординатной минимизации результата СПИДГАУССА)
0,0370249484334057 < 243,227487910481 при t = 16 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 1,14615723112372 %
0,0366005843096164 < 546,332951317169 при t = 17 (до покоординатной минимизации результата СПИДГАУССА)
0,0366005843096164 < 51262841,7404047 при t = 17 (до полно покоординатной минимизации вектора c1 c2 ... 0)
Погрешность уменьшена ПОЛНОЙ МИНИМАКОЙ на 2,53862977376224 %

```

Рис. 24: Как с помощью ультра-гибрида происходит уточнение решения. В неравенствах слева указана наилучшая достигнутая на данный момент аппроксимация, справа – достигнутая на текущей итерации при текущем шаге, t – количество функций на текущей итерации. Как видно, на отдельных шагах аппроксимация может существенно возрасти, иногда ни один из шагов не даёт результата в нескольких итерациях, однако раз в несколько итераций аппроксимация улучшается каким-либо отдельным методом и в общем происходит убывание погрешности

8 Замечание об ортогональных системах

Известно, что для ортогональных систем матрица Грама диагональна, поэтому проекция аппроксимируемой функции на исходное подпространство выражается аналитически рядом Фурье и тогда:

$$\left\| \varphi - \sum_{m=1}^N c_m \alpha_m \right\|_{L_2(Q)} = \left\| \varphi - \sum_{m=1}^N (\varphi, \alpha_m) / (\alpha_m, \alpha_m) \alpha_m \right\|_{L_2(Q)} = \min. \quad (24)$$

Поэтому рекомендуется по возможности использовать ортогональные системы, чтобы минимизировать время вычислений и погрешности. Ранее было показано, что такая аппроксимация убывает очень медленно и не застрахована от неустойчивости. Также было показано, что в свете этого зачастую эффективнее аппроксимировать по неортогональной системе небольшой размерности (пусть даже по неустойчивому алгоритму), чем вычислять решение в виде ряда $\sum_{m=1}^N (\varphi, \alpha_m) / (\alpha_m, \alpha_m) \alpha_m$ по нескольким сотням функций. Не в пользу одного из методов говорят и новые данные: в ходе тестирования ультра-гибридного метода было обнаружено, что между записью решения в виде ряда $\sum_{m=1}^N (\varphi, \alpha_m) / (\alpha_m, \alpha_m) \alpha_m$ и решением диагональной СЛАУ, соответствующей этому ряду, на практике существует разница: по неизвестным причинам решение задачи аппроксимации для ортогональных систем иногда становится точнее, если решать её описанным выше традиционным методом (через решение СЛАУ), не обращая внимания на то, что в теории решение можно записать сразу же; кроме того, убывание погрешности происходит быстрее, а неустойчивость менее выражена. Следующие графики (Рис. 25-26) это подтверждают; в обоих случаях аппроксимация происходит по неустойчивому алгоритму для небольшого числа функций, потому что цель – показать разницу двух способов решения одной и той же задачи; графики представлены для системы Хаара, на которой это явление и было замечено; с тригонометрической системой этот способ работает в точности наоборот; к слову, попытки ускорить вычисления многочленов Лежандра ранее привели к усиленной неустойчивости, так что описанное явление явно связано с погрешностями вычислений.

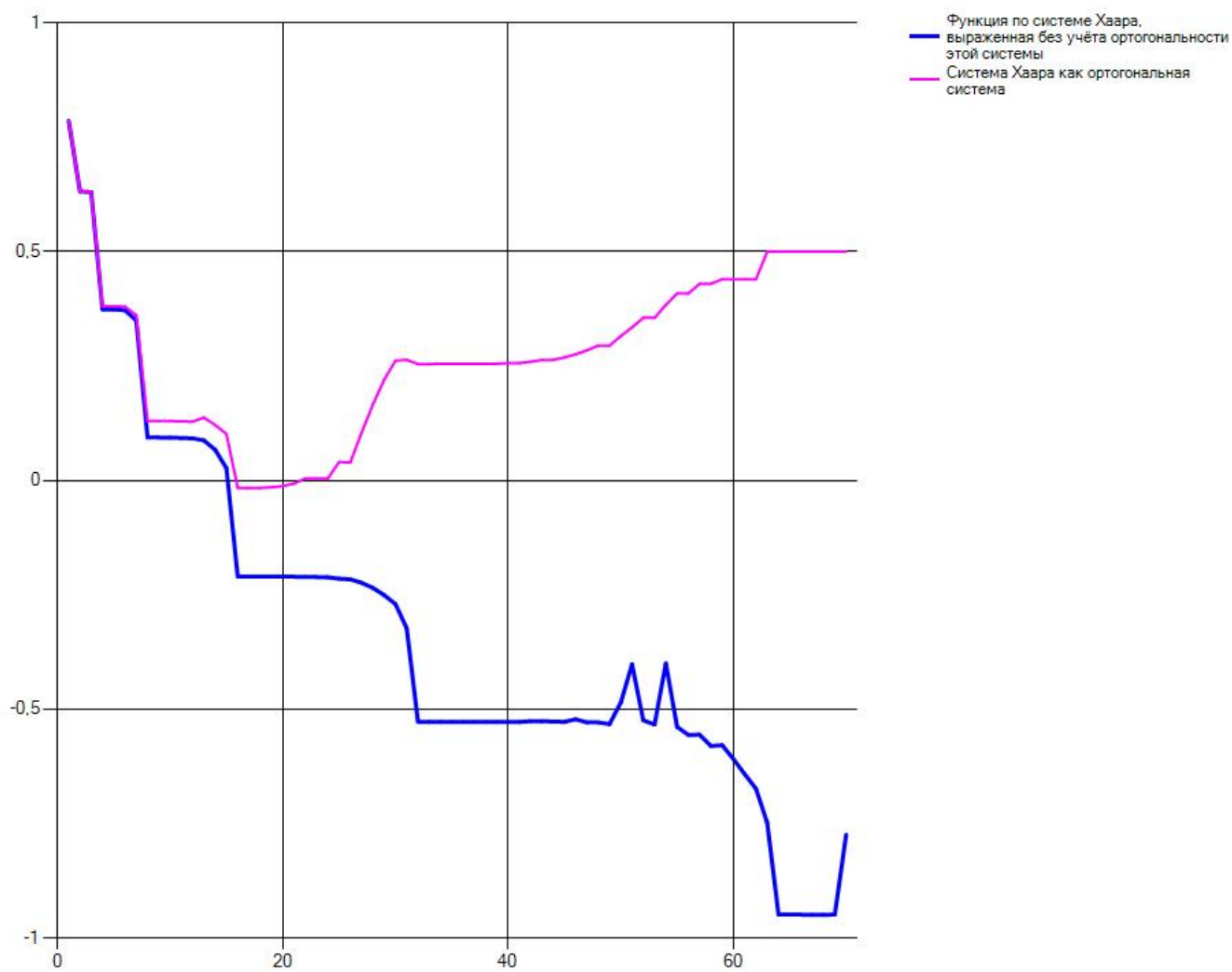


Рис. 25: Разница между двумя способами аппроксимации системой Хаара

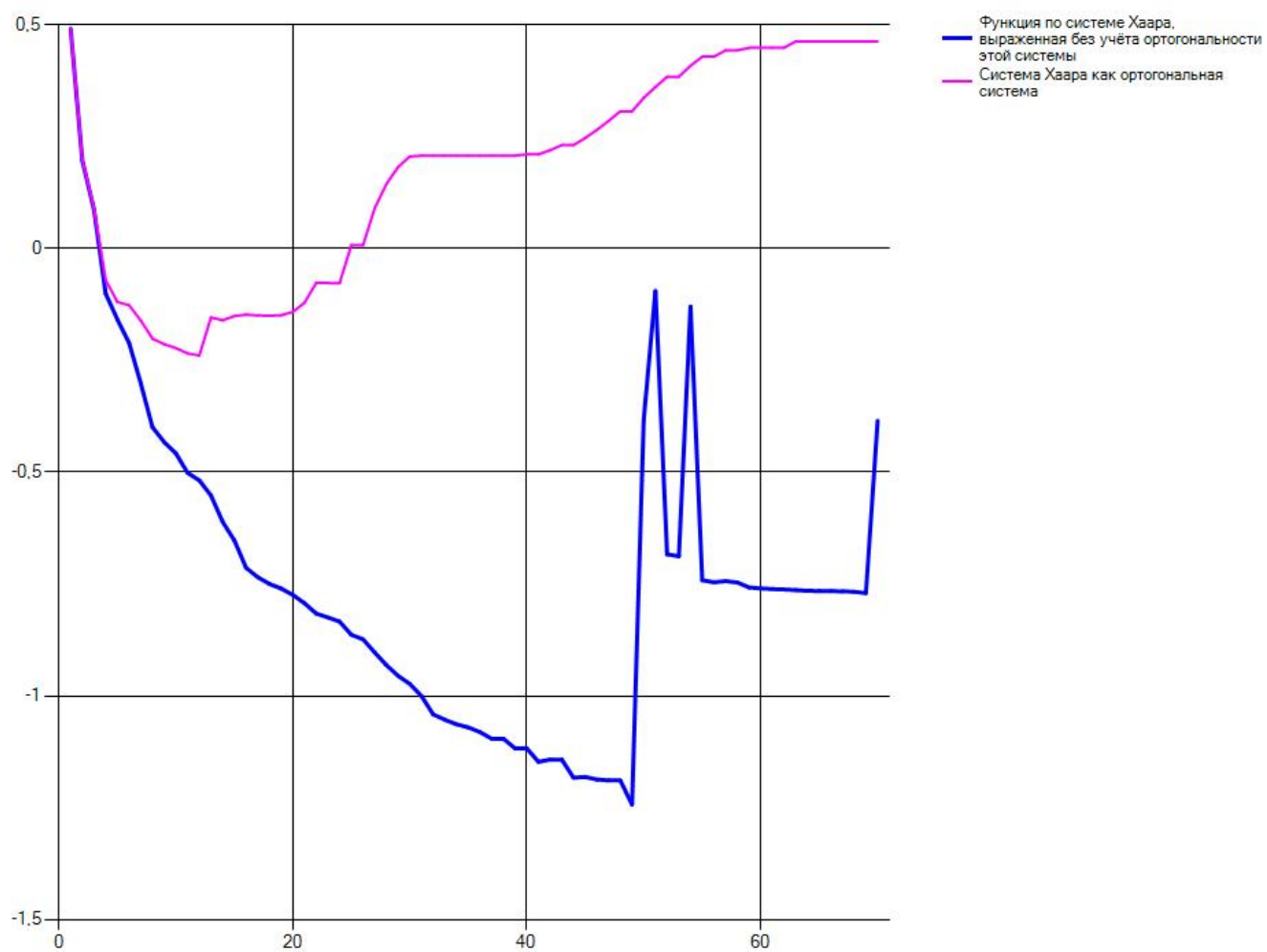


Рис. 26: Такая же разница при других начальных данных задачи

9 Примеры использования ультра-гибридного метода для аппроксимации в пространствах, отличных от $L_2[a, b]$

При практических расчётах аппроксимируемая функция обычно известна либо по набору значений (причём приближённых), либо по своему образу, полученному через дифференциальный или интегральный оператор. Рассмотрим, какую пользу приносит ультра-гибридный метод при решении таких задач на примере МНК и обратной задачи потенциала.

В методе наименьших квадратов происходит дискретное среднеквадратическое приближение сеточной функции, заданной в n точках, измеримыми функциями. В таком случае выкладки получаются абсолютно такими же, как при аппроксимации в $L_2(Q)$, но используются другие выражения для скалярного произведения

$$(f, \varphi)_{R_n^{[a,b]}} = \frac{1}{n} \sum_{i=1}^n f(x_i) \varphi(x_i) \quad (25)$$

и нормы

$$\|f\|_{R_n^{[a,b]}} = \sqrt{\frac{1}{n} \sum_{i=1}^n f(x_i)^2}. \quad (26)$$

Несмотря на более точное вычисление скалярных произведений в случае сеточной функции, в МНК так же встречается неустойчивость. Аппроксимируем, например, фиксированную сеточную функцию (проекцию действительной функции) некоторыми системами размерности 4 (Рис. 27).

Качество аппроксимации для системы Хаара следующее: в интегральной среднеквадратичной норме равна 3,41841902293629, в равномерной норме равна 11,1746183943873, в дискретной среднеквадратичной норме равна 4,53800130664926. Если увеличить количество функций до 6, получим заметную неустойчивость (Рис. 28).

Те же данные об аппроксимации: погрешность аппроксимации в интегральной среднеквадратичной норме равна 1,50959632325169E+17, в равномерной норме равна 4,35248462504321E+17, в дискретной среднеквадратичной норме) равна 6,21783517863316E+16.

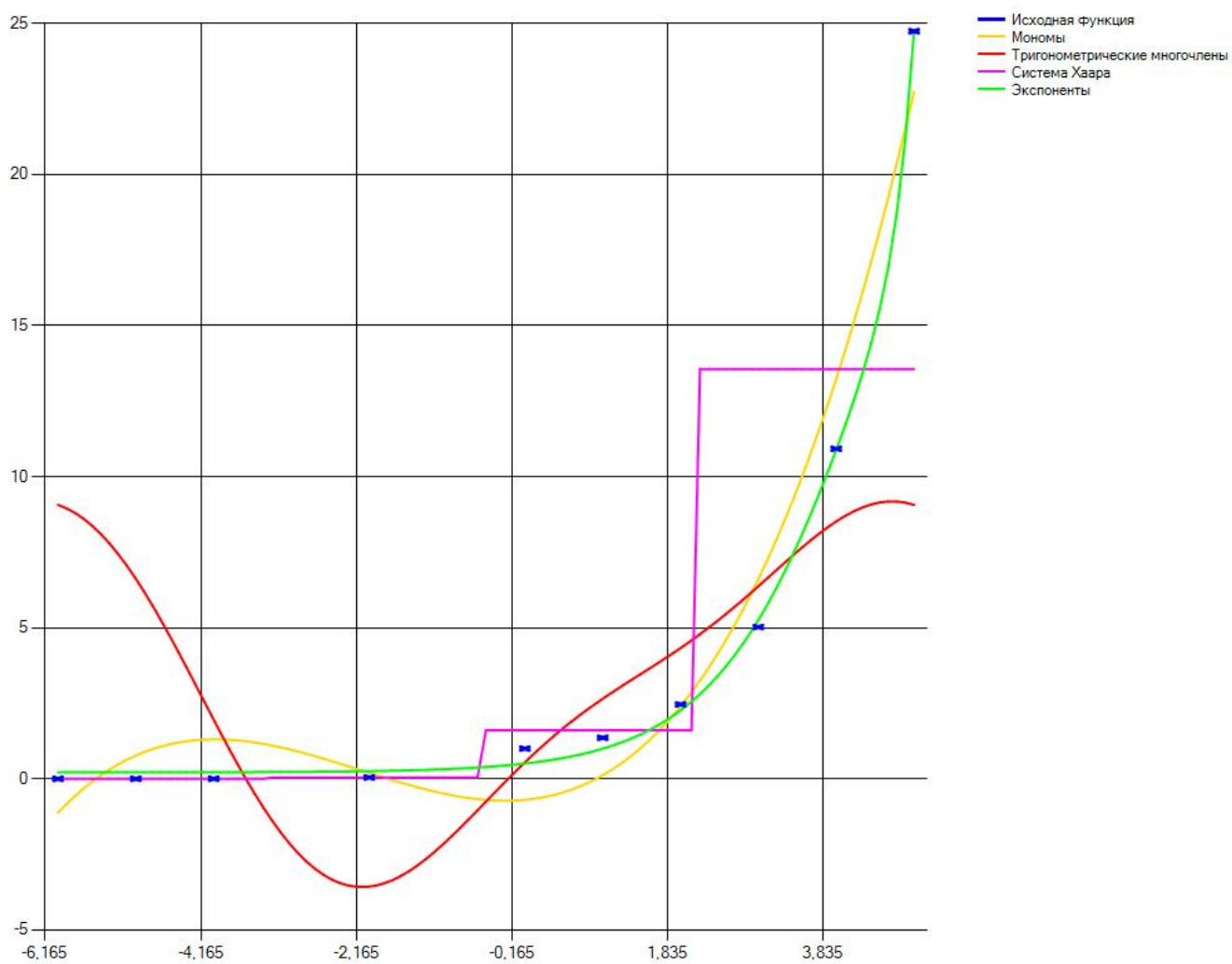


Рис. 27: Аппроксимация сеточной функции несколькими системами размерности 4

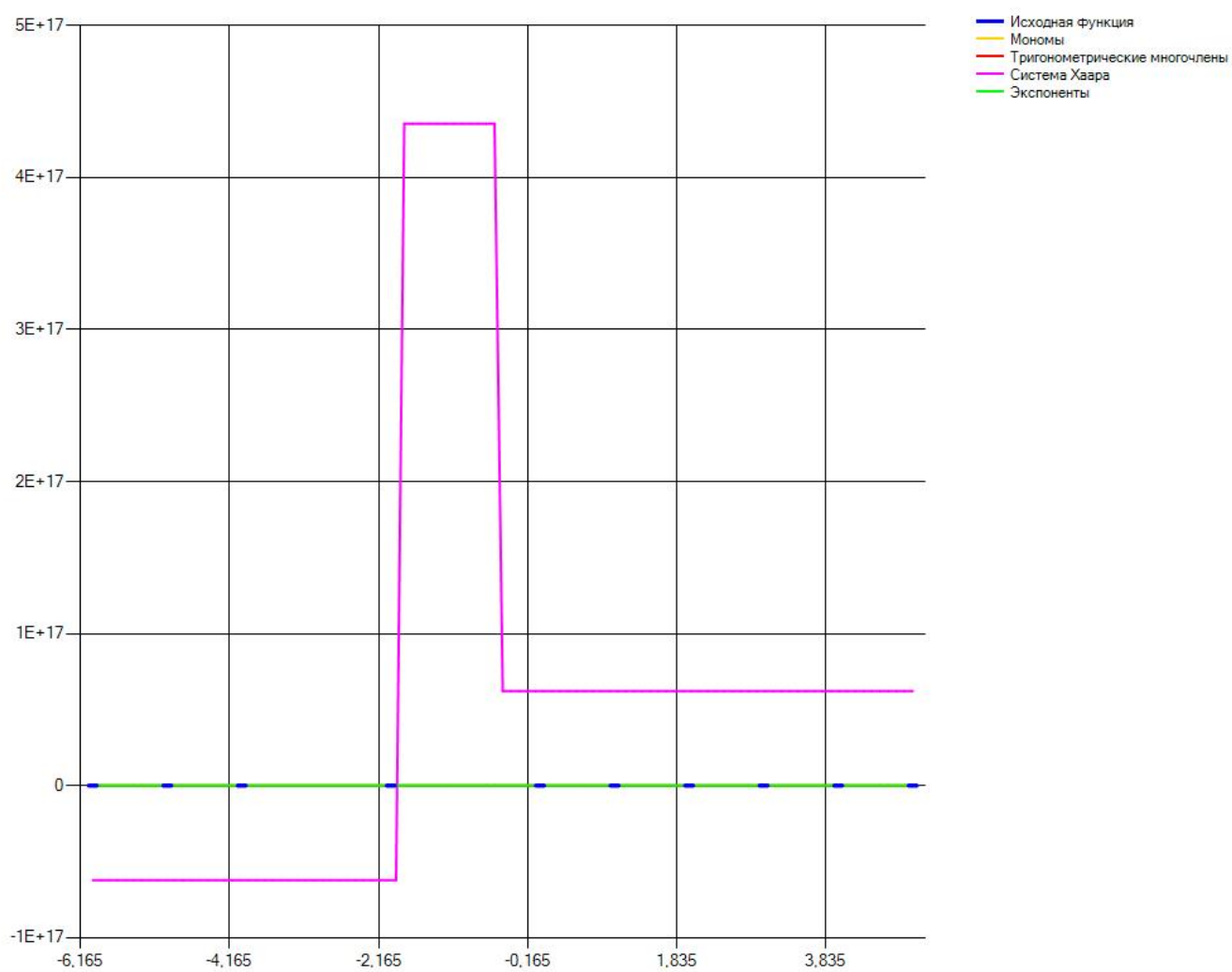


Рис. 28: Неустойчивость системы Хаара при размерности 6

10 Обзор результатов

Список литературы

- [1] С. Дасгупта, Х. Пападимитриу, У. Вазирани. Алгоритмы. — М.: МЦНМО, 2014.
- [2] Колмогоров А. Н., Фомин С. В.. Элементы теории функции и функционального анализа. — М.: Наука. Гл. ред. физ.-мат. лит. 1989.
- [3] Н. П. Корнейчук. Экстремальные задачи теории приближения. — М.: Наука. Гл. ред. физ.-мат. лит. 1976.
- [4] Н. И. Ахиезер. Лекции по теории аппроксимации. — М.: Наука. Гл. ред. физ.-мат. лит. 1964.
- [5] Г. С. Шевцов, О. Г. Крюкова, Б. И. Мызникова. Численные методы линейной алгебры: Учеб. пособие. — М.: Финансы и статистика: ИНФРА-М, 2008.
- [6] И. Б. Петров, А. И. Лобанов.. Лекции по вычислительной математике: Учебное пособие. — М.: Интернет-Университет Информационных Технологий; БИНОМ. Лабиринт знаний, 2006.
- [7] Справочник алгоритмов на языке АЛГОЛ. Линейная алгебра. — М.: Машиностроение, 1976.