

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ
РАСТИТЕЛЬНЫХ ПОЛИМЕРОВ»

В.М. Пестриков, Т.С. Смирнова, Н.Л. Леонова

КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ
ЗАДАЧ ТЕПЛОТЕХНИКИ

Учебно-методическое пособие
по курсовому проекту

Санкт-Петербург
2012

ББК 32.9я7
П 286
УДК 681.3(075)

Пестриков В.М., Смирнова Т.С., Леонова Н.Л. Компьютерное моделирование задач теплотехники: учебно-методическое пособие по курсовому проекту /СПб ГТУРП. - СПб.,2012.-69 с.

В пособии изложены теоретические и практические основы математического моделирования задач теплотехники. Рассмотрены метод наименьших квадратов, метод сеток решения уравнения Лапласа, изложены элементы работы с массивами и векторами, решение линейных и нелинейных уравнений в среде MathCad.

Пособие предназначено для студентов, обучающихся по направлению 140100 «Теплоэнергетика и теплотехника», а также может быть полезно при изучении таких дисциплин, как «Численные методы», «Вычислительная математика».

Рецензенты:

Подобед М.В., канд.техн.наук, доцент кафедры ПМиИ СПб ГТУРП
Тихов С.В., канд.техн.наук, доцент кафедры «Информатика» СПбГУСЭ.

Рекомендовано к изданию Редакционно-издательским советом университета в качестве учебно-методического пособия.

© Пестриков В.М., Смирнова Т.С.,
Леонова Н.Л., 2012

© Санкт-Петербургский государственный
технологический университет
растительных полимеров, 2012

Учебное издание

Виктор Михайлович Пестриков
Тамара Сергеевна Смирнова
Надежда Львовна Леонова

Компьютерное моделирование задач теплотехники

Учебно-методическое пособие по курсовому проекту

Редактор и корректор Н.П. Новикова

Техн. редактор Л.Я. Титова

Тем. план 2012 г., поз. 45

Подп. к печати 05.06.12.

Формат 60x84/16.

Бумага тип.№ 1.

Печать офсетная. 4,5 уч.-изд.л.; 4,5 усл. печ. л.

Тираж 50 экз. Изд.№ 45. Цена «С». Заказ

Ризограф Санкт-Петербургского государственного технологического университета растительных полимеров, 198095, СПб., ул. Ивана Черных, 4.

**В.М. Пестриков
Т.С. Смирнова
Н.Л. Леонова**

**Компьютерное
моделирование
задач теплотехники**

**Учебно-методическое пособие
по курсовому проекту**

**Санкт-Петербург
2012**

1. ОСНОВНЫЕ ПОНЯТИЯ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ

1.1. Компьютерное моделирование как метод научного познания

Компьютерное моделирование является междисциплинарным курсом. Для его успешного освоения требуется наличие самых разнообразных знаний: во-первых, знаний в выбранной предметной области - если мы моделируем физические процессы, мы должны обладать определенным уровнем знания законов физики, моделируя экологические процессы - биологических законов, моделируя экономические процессы - знанием законов экономики. Кроме того, так как компьютерное моделирование использует практически весь аппарат современной математики, предполагается знание основных математических дисциплин - алгебры, математического анализа, теории дифференциальных уравнений, математической статистики, теории вероятностей. Для решения математических задач на компьютере необходимо владеть в полном объеме численными методами решения нелинейных уравнений, систем линейных уравнений, дифференциальных уравнений, уметь аппроксимировать и интерполировать функции. И, конечно же, предполагается свободное владение современными информационными технологиями, знание языков программирования и владение навыками разработки прикладных программ [1,2].

Компьютерное моделирование, возникшее как одно из направлений математического моделирования, с развитием информационных компьютерных технологий стало самостоятельной и важной областью применения компьютеров. В настоящее время компьютерное моделирование в научных и практических исследованиях является одним из основных методов познания. Без компьютерного моделирования сейчас невозможно решение крупных научных и экономических задач. Выработана технология

исследования сложных проблем, основанная на построении и анализе с помощью вычислительной техники математической модели изучаемого объекта. Такой метод исследования называется вычислительным экспериментом. Вычислительный эксперимент применяется практически во всех отраслях науки - в физике, химии, астрономии, биологии, экологии, даже в таких сугубо гуманитарных науках, как психология, лингвистика и филология; вычислительные эксперименты широко применяются в экономике, в социологии, в промышленности, в управлении. Проведение вычислительного эксперимента имеет ряд преимуществ перед так называемым натурным экспериментом:

- для вычислительного эксперимента не требуется сложного лабораторного оборудования;
- существенное сокращение временных затрат на эксперимент;
- возможность свободного управления параметрами, произвольного их изменения, вплоть до придания им нереальных, неправдоподобных значений;
- возможность проведения вычислительного эксперимента там, где натурный эксперимент невозможен из-за удаленности исследуемого явления в пространстве (астрономия), либо из-за его значительной растянутости во времени (биология), либо из-за возможности внесения необратимых изменений в изучаемый процесс.

В этих случаях и используется компьютерное моделирование. Также компьютерное моделирование широко используется в образовательных и учебных целях. Компьютерное моделирование - наиболее адекватный подход при изучении предметов естественнонаучного цикла, изучение компьютерного моделирования открывает широкие возможности для осознания связи информатики с математикой и другими науками - естественными и социальными. Преподаватель может использовать на занятии готовые компьютерные модели для демонстрации изучаемого

явления, будь это движение астрономических объектов или движение атомов, или модель молекулы, или рост микробов и т.д. Также преподаватель может озадачить учащихся разработкой конкретных моделей: моделируя конкретное явление, студент не только освоит конкретный учебный материал, но и приобретет умение ставить проблемы и задачи, прогнозировать результаты исследования, проводить разумные оценки, выделять главные и второстепенные факторы для построения моделей, выбирать аналогии и математические формулировки, использовать компьютер для решения задач, проводить анализ вычислительных экспериментов.

Понятие моделирования - это очень широкое понятие, оно не ограничивается только математическим моделированием. Истоки моделирования обнаруживаются в далеком прошлом. Наскальные изображения мамонта, пронзенного копьем, на стене пещеры можно рассматривать как модель удачной охоты, созданную древним художником.

Элементы моделирования часто присутствуют в детских играх, любимое занятие детей - моделировать подручными средствами предметы и отношения из жизни взрослых. Взрослеют дети, взрослеет человечество. Человечество познает окружающий мир, модели становятся более абстрактными, теряют внешнее сходство с реальными объектами. В моделях отражаются глубинные закономерности, установленные в результате целенаправленных исследований. В роли моделей могут выступать самые разнообразные объекты: изображения, схемы, карты, графики, компьютерные программы, математические формулы и т.д. Если мы заменяем реальный объект математическими формулами (допустим, согласно 2-му закону Ньютона, опишем движение некоторого тела системой нелинейных уравнений, или, согласно закону теплопроводности, опишем процесс распространения тепла дифференциальным уравнение 2-го порядка),

то говорят о математическом моделировании, если реальный объект заменяем компьютерной программой - о компьютерном моделировании.

1.2. Классификация моделей

В зависимости от средств построения различают следующие классы моделей [2,3]:

- словесные или описательные модели их также иногда называют вербальными или текстовыми моделями;
- натурные модели (макет Солнечной системы, игрушечный кораблик);
- абстрактные или знаковые модели. Интересующие нас математические модели явлений и компьютерные модели относятся как раз к этому классу.

Можно классифицировать модели по предметной области:

- физические модели;
- биологические;
- социологические;
- экономические и т.д.

Классификация моделей по применяемому математическому аппарату:

- модели, основанные на применении обыкновенных дифференциальных уравнений;
- модели, основанные на применении уравнений в частных производных;
- вероятностные модели и т.д.

Также можно классифицировать модели по цели моделирования. В зависимости от целей моделирования различают:

- дескриптивные модели (описательные) описывают моделируемые объекты и явления и как бы фиксируют сведения человека о них. Примерами могут служить модель Солнечной системы или модель движения кометы, в которой мы моделируем траекторию ее полета, расстояние, на котором она пройдет от Земли. У нас нет никаких возможностей повлиять на движение кометы или движение планет Солнечной системы;

- оптимизационные модели служат для поиска наилучших решений при соблюдении определенных условий и ограничений. В этом случае в модель входит один или несколько параметров, доступных нашему влиянию, например, известная задача коммивояжера: оптимизируя его маршрут, мы снижаем стоимость перевозок. Часто приходится оптимизировать процесс по нескольким параметрам сразу, причем цели могут быть весьма противоречивы, например, головная боль любой хозяйки - как вкуснее, калорийнее и дешевле накормить семью;

- игровые модели (компьютерные игры);
- обучающие модели (всевозможные тренажеры);
- имитационные модели (модели, в которых сделана попытка более или менее полного и достоверного воспроизведения некоторого реального процесса, например, моделирование движения молекул в газе, поведение колонии микробов и т.д.).

Существует также классификация моделей в зависимости от их изменения во времени. Различают:

- статические модели - неизменные во времени;
- динамические модели - состояние которых меняется со временем.

1.3. Технология компьютерного моделирования

В технологии компьютерного моделирования можно выделить следующие основные понятия.

Модель - искусственно созданный объект, который воспроизводит в определенном виде реальный объект - оригинал.

Компьютерная модель - представление информации о моделируемой системе средствами компьютера.

Система - совокупность взаимосвязанных элементов, обладающая свойствами, отличными от свойств отдельных элементов.

Элемент - это объект, обладающий свойствами, важными для целей моделирования. В компьютерной модели свойства элемента представляются величинами - характеристиками элемента.

Связь между элементами описывается с помощью величин и алгоритмов, в частности, вычислительных формул.

Состояние системы представляется в компьютерной модели набором характеристик элементов и связей между элементами. Структура данных, описывающих состояние, не зависит от конкретного состояния и не меняется при смене состояний, меняется только значение характеристик.

Если состояния системы функционально зависят от некоторого параметра, то *процессом* называют набор состояний, соответствующий упорядоченному изменению параметра. Параметры в системе могут меняться как непрерывно, так и дискретно. В компьютерной модели изменение параметра всегда дискретно. Непрерывные процессы можно моделировать на компьютере, выбирая дискретную серию значений параметра так, чтобы

последовательные состояния мало чем отличались друг от друга, или, другими словами, минимизируя шаг по времени.

Модели могут принимать различную форму, в зависимости от способа мышления исследователя, его взгляда на мир, используемой алгебры. Использование различных математических аппаратов впоследствии приводит к различным возможностям в решении задач [1,2].

Модели могут быть:

- феноменологические и абстрактные;
- активные и пассивные;
- статические и динамические;
- дискретные и непрерывные;
- детерминированные и стохастические;
- распределенные, структурные и сосредоточенные;
- функциональные и объектные.

Феноменологические модели сильно привязаны к конкретному явлению. Изменение ситуации часто приводит к тому, что моделью воспользоваться в новых условиях достаточно сложно. Это происходит оттого, что при составлении модели её не удалось построить с точки зрения подобия внутреннему строению моделируемой системы. Феноменологическая модель передаёт внешнее подобие.

Абстрактная модель воспроизводит систему с точки зрения её внутреннего устройства, копирует её более точно. У неё больше возможностей, шире класс решаемых задач.

Активные модели взаимодействуют с пользователем; могут не только, как пассивные, выдавать ответы на вопросы пользователя, когда тот об этом попросит, но и сами активируют диалог, меняют его линию, имеют

собственные цели. Все это происходит за счёт того, что активные модели могут самоизменяться.

Статические модели описывают явления без развития. Динамические модели прослеживают поведение систем, поэтому используют в своей записи, например, дифференциальные уравнения с производными по времени.

Дискретные и непрерывные модели. Дискретные модели изменяют состояние переменных скачком, потому что не имеют детального описания связи причин и следствий, часть процесса скрыта от исследователя. Непрерывные модели более точны, содержат в себе информацию о деталях перехода.

Детерминированные и стохастические модели. Если следствие точно определено причиной, то модель представляет процесс детерминированно. Если из-за неизученности деталей не удаётся описать точно связь причин и следствий, а возможно только описание в целом, статистически (что часто и бывает для сложных систем), то модель строится с использованием понятия вероятности.

Распределённые, структурные, сосредоточенные модели. Если параметр, описывающий свойство объекта, в любых его точках имеет одинаковое значение (хотя может меняться во времени!), то это система с сосредоточенными параметрами. Если параметр принимает разные значения в разных точках объекта, то говорят, что он распределён, а модель, описывающая объект, — распределённая. Иногда модель копирует структуру объекта, но параметры объекта сосредоточены, тогда модель — структурная.

Функциональные и объектные модели. Если описание идёт с точки зрения поведения, то модель построена по функциональному признаку. Если описание каждого объекта отделено от описания другого объекта, если

описываются свойства объекта, из которых вытекает его поведение, то модель является объектно-ориентированной.

Каждый подход имеет свои достоинства и недостатки. Разные математические аппараты имеют разные возможности (мощность) для решения задач, разные потребности в вычислительных ресурсах. Один и тот же объект может быть описан различными способами. Инженер должен грамотно применять то или иное представление, исходя из текущих условий и стоящей перед ним проблемы.

1.4. Этапы компьютерного моделирования

В технологии компьютерного моделирования можно выделить несколько этапов. На рис 1.1 представлены этапы построения модели.

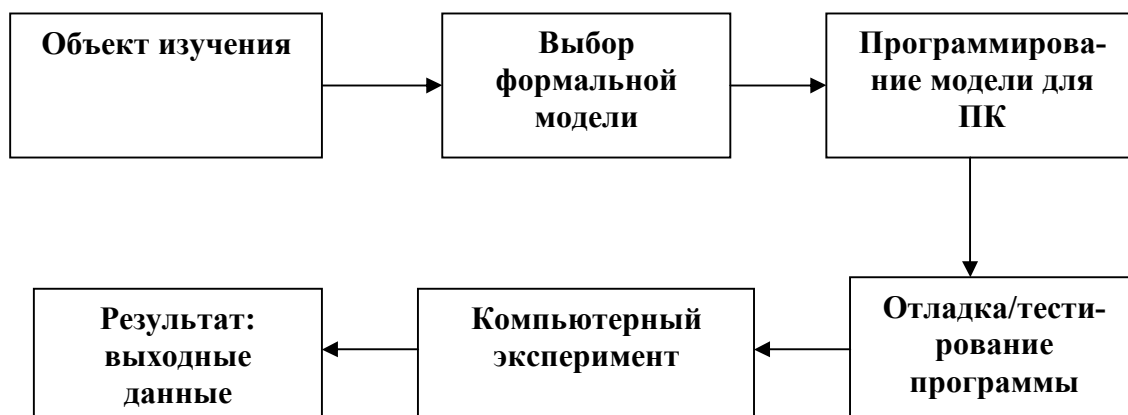


Рис.1.1. Этапы компьютерного моделирования

Моделирование начинается с *объекта изучения*, постановки проблемы. Затем формируются законы, управляющие исследованием, происходит отделение информации от реального объекта, формируется существенная информация, отбрасывается несущественная, происходит первый шаг абстракции. Преобразование информации определяется решаемой задачей. Информация, существенная для одной задачи, может оказаться несущественной для другой. Потеря существенной информации

приводит к неверному решению или не позволяет вообще получить решение. Учет несущественной информации вызывает излишние сложности, а иногда создает непреодолимые препятствия на пути к решению. Переход от реального объекта к информации о нем осмыслен только тогда, когда поставлена задача. В то же время постановка задачи уточняется по мере изучения объекта. Таким образом, параллельно идут процессы целенаправленного изучения объекта и уточнения задачи. Информация об объекте подготавливается к обработке на компьютере. Строится так называемая *формальная модель явления*, которая содержит:

- набор постоянных величин, констант, которые характеризуют моделируемый объект в целом и его составные части; называемых статическими или постоянными параметрами модели;
- набор переменных величин, меняя значение которых можно управлять поведением модели, называемых динамическим или управляющими параметрами;
- формулы и алгоритмы, связывающие величины в каждом из состояний моделируемого объекта;
- формулы и алгоритмы, описывающие процесс смены состояний моделируемого объекта.

На следующем этапе формальная модель реализуется на компьютере, выбираются подходящие программные средства для этого, строится *алгоритм решения проблемы*, пишется *программа*, реализующая этот алгоритм, затем написанная программа отлаживается и тестируется на специально подготовленных тестовых моделях. *Тестирование* - это процесс исполнения программы с целью выявления ошибок. Подбор тестовой модели - это своего рода искусство, хотя для этого разработаны и успешно применяются некоторые основные принципы тестирования. Тестирование - это процесс деструктивный, поэтому считается, что тест удачный, если обнаружена ошибка. Проверить компьютерную модель на соответствие

оригиналу, проверить, насколько хорошо или плохо отражает модель основные свойства объекта, часто удается с помощью простых модельных примеров, когда результат моделирования известен заранее.

Далее, работая с компьютерной моделью, мы осуществляем непосредственно *вычислительный эксперимент*. Исследуем, как поведет себя наша модель в том или ином случае, при тех или иных наборах динамических параметров, пытаемся прогнозировать или оптимизировать что-либо в зависимости от поставленной задачи.

Результатом компьютерного эксперимента будет являться информационная модель явления в виде графиков, зависимостей одних параметров от других, диаграмм, таблиц, демонстрации явления в реальном или виртуальном времени и т.п.

2. ПРИБЛИЖЕНИЕ ФУНКЦИЙ

2.1. Введение в теорию приближений

Аппроксимацией (приближением) функции $f(x)$ называется нахождение такой функции $g(x)$ (аппроксимирующей функции), которая была бы близка заданной. Критерии близости функций $f(x)$ и $g(x)$ могут быть различные [4,5].

В том случае, когда приближение строится на дискретном наборе точек, аппроксимацию называют точечной или дискретной.

В том случае, когда аппроксимация проводится на непрерывном множестве точек (отрезке), аппроксимация называется непрерывной или интегральной. Примером такой аппроксимации может служить разложение функции в ряд Тейлора, т.е. замена некоторой функции степенным многочленом.

Наиболее часто встречающимся видом точечной аппроксимации

является интерполяция (в широком смысле).

Пусть задан дискретный набор точек x_i ($i = 0, 1, \dots, n$), называемых узлами интерполяции, причем среди этих точек нет совпадающих, а также значения функции y_i в этих точках. Требуется построить функцию $g(x)$, проходящую через все заданные узлы. Таким образом, критерием близости функции является $g(x_i) = y_i$.

В качестве функции $g(x)$ обычно выбирается полином, который называют интерполяционным полиномом.

В том случае, когда полином един для всей области интерполяции, говорят, что интерполяция глобальная.

В тех случаях, когда между различными узлами полиномы различны, говорят о кусочной или локальной интерполяции.

Найдя интерполяционный полином, мы можем вычислить значения функции $f(x)$ между узлами (провести интерполяцию в узком смысле слова), а также определить значение функции $f(x)$ даже за пределами заданного интервала (провести экстраполяцию).

Следует иметь в виду, что точность экстраполяции обычно очень невелика [4].

2.2. Метод наименьших квадратов

2.2.1. Постановка задачи

Рассмотрим один из методов, позволяющих проанализировать и обработать данные, полученные в результате эксперимента (табл. 2.1). Пусть в результате измерений получена таблица зависимости одной величины y от другой x .

Таблица 2.1 Экспериментальные данные.

x_i	x_1	x_2	x_3	...	x_n
y_i	y_1	y_2	y_3	...	y_n

Необходимо найти формулу $y = f(x)$, выражающую таблично заданную зависимость аналитически. Применение интерполяции в данном случае нецелесообразно, так как значения y_i в узлах получены экспериментально и поэтому являются сомнительными (в ходе эксперимента возникает неустранимая погрешность, обусловленная неточностью измерений). Кроме того, совпадение значений в узлах не означает совпадения характеров поведения исходной и интерполирующей функций. Поэтому необходимо найти такой метод подбора эмпирической формулы, который позволяет не только найти саму формулу, но и оценить погрешность подгонки [4].

Найдем функцию заданного вида

$$y = f(x), \tag{2.1}$$

которая в точках $x_1, x_2, x_3, x_4, \dots, x_n$ принимает значения, как можно более близкие к табличным значениям $y_1, y_2, y_3, y_4, \dots, y_n$.

Практически вид приближающей функции можно определить визуально: по табл. 2.1 строится точечный график функции, а затем проводится кривая, по возможности наилучшим образом отражающая характер расположения точек (рис.2.1).

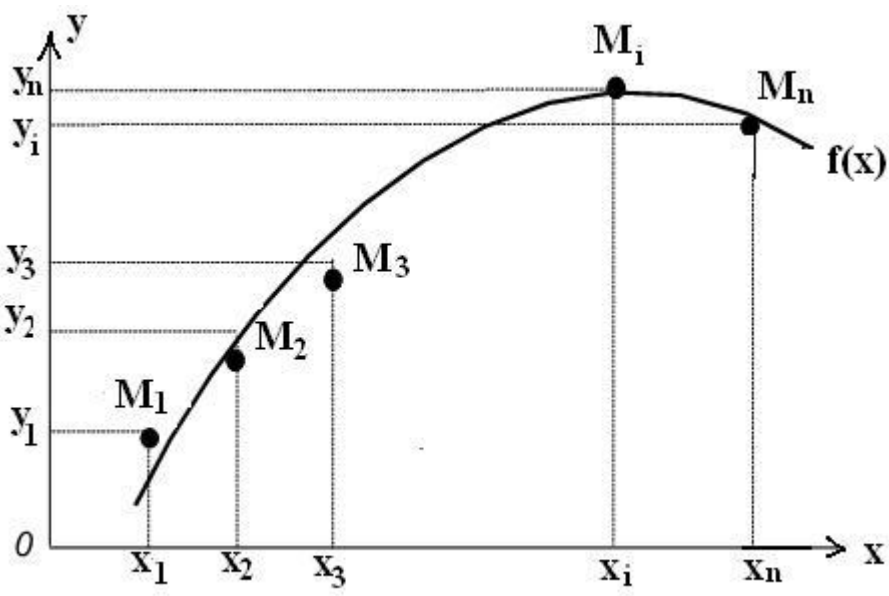


Рис.2.1. Определение вида приближающей функции

По полученной кривой устанавливается вид приближающей функции (обычно из числа простых по виду аналитических функций: линейная, степенная, экспоненциальная или показательная, логарифмическая, гипербола, дробно-рациональная и т.д.).

Заметим, что формула (2.1), называемая эмпирической формулой или уравнением регрессии y на x , позволяет находить значения функции $f(x)$ для нетабличных значений x , «сглаживая» результаты измерений величины y .

Из рис. 2.1 видно, что для каждого значения x_i экспериментальное y_i и расчетное $y_i^p = f(x_i)$ значения различаются на некоторую величину Δy_i , называемую абсолютной разностью. Потребовав, чтобы сумма квадратов абсолютных разностей для всех точек была минимальной, найдем оптимальные параметры функции $f(x)$: если выполняется условие

$$\left(\sum_{i=1}^n (\Delta y_i)^2\right) \rightarrow 0, \tag{2.2}$$

где $\Delta y_i = y_i - y_i^p = y_i - f(x_i)$, то считается, что функция $f(x)$ подобрана

наилучшим образом.

Рассмотрим все изложенное выше на примере линейной регрессии.

2.2.2. Линейная регрессия

Будем искать приближающую функцию в виде

$$y = f(x, k, b) = kx + b.$$

Абсолютная разность Δy_i для x_i определяется следующим образом:

$$\Delta y_i = y_i - y_i^p = y_i - f(x_i) = y_i - (kx_i + b), \quad (2.3)$$

формулу (2.3) перепишем в виде:

$$\sigma = \sum_{i=1}^n (\Delta y_i)^2 = \sum_{i=1}^n (y_i - (kx_i + b))^2.$$

Рассматриваемая сумма является функцией с двумя параметрами $\sigma = F(k, b)$. Задача сводится к отысканию минимума этой функции.

Используем необходимое условие экстремума:

$$\begin{aligned} \frac{\partial F(k, b)}{\partial k} &= 0; \\ \frac{\partial F(k, b)}{\partial b} &= 0, \end{aligned}$$

т.е.

$$\frac{\partial \sigma}{\partial k} = \frac{\partial \sum_{i=1}^n (\Delta y_i)^2}{\partial k} = \frac{\partial \sum_{i=1}^n (y_i - (kx_i + b))^2}{\partial k} = 0; \quad (2.4)$$

$$\frac{\partial \sigma}{\partial b} = \frac{\partial \sum_{i=1}^n (\Delta y_i)^2}{\partial b} = \frac{\partial \sum_{i=1}^n (y_i - (kx_i + b))^2}{\partial b} = 0.$$

Решив систему двух уравнений с двумя неизвестными относительно параметров k и b , получим конкретный вид искомой функции $y = kx + b$. Опуская математические выкладки, запишем выражения для искомых параметров:

$$k = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}; \quad (2.5)$$

$$b = \frac{1}{n} \left(\sum_{i=1}^n y_i - k \sum_{i=1}^n x_i \right).$$

Рассчитав значение σ , получим величину среднеквадратичной ошибки рассматриваемого приближения.

Найденные значения k и b определяют точку экстремума $\sigma = F(k, b)$, в этой точке функция принимает минимальное значение.

Как видно из рассмотренного примера, изменение количества параметров не приведет к искажению сущности самого подхода, изменится лишь количество уравнений в системе (2.4) (для n параметров соответственно будет записано n уравнений) [4].

2.2.3. Подбор эмпирических формул

Чтобы подобрать формулу, выражающую зависимость между двумя величинами, если это зависимость найдена опытным путем, строят график этой зависимости. Полученный график сравнивают по внешнему виду с графиками, построенными при помощи известных формул. Формулы содержат небольшое количество параметров (коэффициенты, показатели степеней и т.д.), изменением которых можно в той или иной степени менять вид кривой. Чтобы формула не оказалась слишком сложной, количество параметров не должно быть велико. Обычно берут два-три параметра. При

сравнении обращают внимание на наличие максимумов и минимумов, поведение функции при больших и малых значениях аргумента, выпуклость кривой вверх или вниз на отдельных участках и т.д. Выбрав среди известных графиков подходящий, следует подобрать такие значения параметров в формуле, чтобы разница между опытными значениями величины и значениями, найденными по формуле, не превышала ошибок эксперимента. Если эта разница получается слишком большой, берут другой подходящий график и повторяют попытку [4].

2.2.4. Степенная зависимость (геометрическая регрессия)

Степенная зависимость имеет вид

$$y = ax^b . \quad (2.6)$$

Во всех случаях $y = a$ при $x = 1$. При $b > 1$ в точке $x = 0$ кривая касается оси абсцисс. В этом случае, чем больше b , тем ближе подходит кривая к оси абсцисс при $0 < x < 1$ и тем быстрее она возрастает при $x > 1$. (рис.2.2.)

При $0 < b < 1$ в точке $x = 0$ кривая касается оси ординат. При $-1 < b < 0$ кривая ближе подходит к оси ординат, чем к оси абсцисс, при $b < -1$ наоборот.

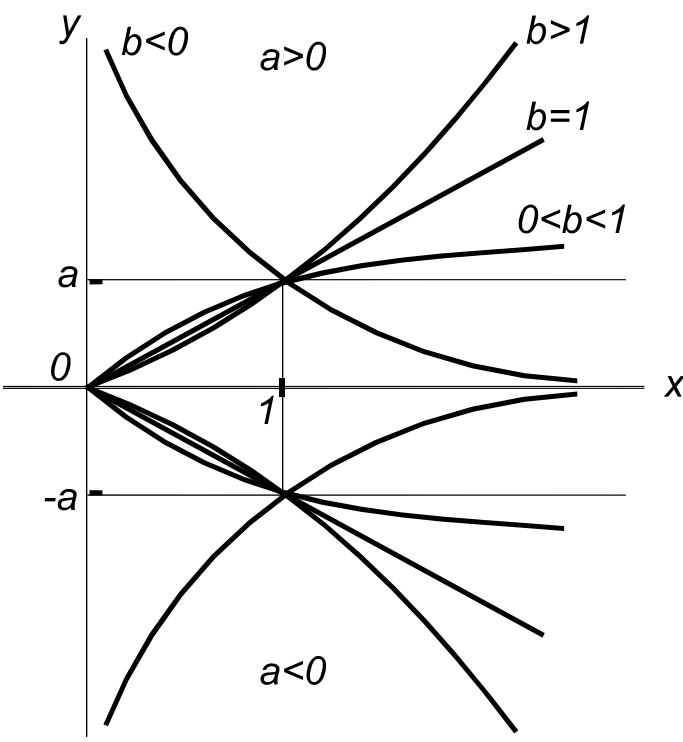


Рис. 2.2. График степенной зависимости

Покажем, как нахождение приближающей функции в виде геометрической регрессии может быть сведено к нахождению параметров линейной функции. Предполагая, что в исходной табл. 2.1 значения аргумента и функции положительны, прологарифмируем равенство (2.6) при условии $a > 0$:

$$\ln y = \ln a + b \ln x. \tag{2.7}$$

Введем новую переменную $t = \ln x$, тогда $\ln y$ будет функцией от t . Обозначим $A = \ln a$, $q = \ln y$, тогда равенство (2.7) примет вид

$$q(t) = A + bt,$$

т.е. задача свелась к отысканию приближающей функции в виде линейной.

Практически для нахождения приближающей функции в виде степенной (при сделанных выше предположениях) необходимо проделать следующие операции:

- 1) по данной табл. 2.1 составить новую табл. 2.2, прологарифмировав значения x и y в исходной таблице;

Таблица 2.2. Прологарифмированные экспериментальные данные

t	ln (x ₁)	ln (x ₂)	ln (x ₃)	...	ln (x _n)
Q(t)	ln (y ₁)	ln (y ₂)	ln (y ₃)	...	ln (y _n)

2) по новой табл. 2.2 найти параметры A и b приближающей функции вида $q(t) = A + bt$;

3) используя примененные обозначения, найти значения параметров a и b и подставить их в выражение (2.6).

Окончательно получаем:

$$b = \frac{n \sum_{i=1}^n \ln x_i \ln y_i - \sum_{i=1}^n \ln x_i \sum_{i=1}^n \ln y_i}{n \sum_{i=1}^n (\ln x_i)^2 - \left(\sum_{i=1}^n \ln x_i \right)^2}; \quad (2.8)$$

$$a = \exp \left(\frac{1}{n} \left(\sum_{i=1}^n \ln y_i - k \sum_{i=1}^n \ln x_i \right) \right).$$

2.2.5. Показательная зависимость

Показательная зависимость имеет вид

$$y = f(x, a, k) = a \exp(kx) = ae^{kx}. \quad (2.9)$$

Во всех случаях $y = a$ при $x = 0$. Если $a > 0$, то при $k > 0$ кривая растет с увеличением x тем быстрее, чем больше k . При $k < 0$ она приближается к оси абсцисс с возрастанием x тем быстрее, чем больше абсолютная величина k . (рис. 2.3.)

Если найденная на опыте зависимость y от x является показательной, то график зависимости $\ln y$ от x представляет собой прямую линию, тангенс угла наклона которой равен параметру k . Если значение y при $x = 0$ неизвестно, то величину параметра a можно найти по формуле $y = a \exp(kx)$

для ряда значений x , а затем взять среднее.

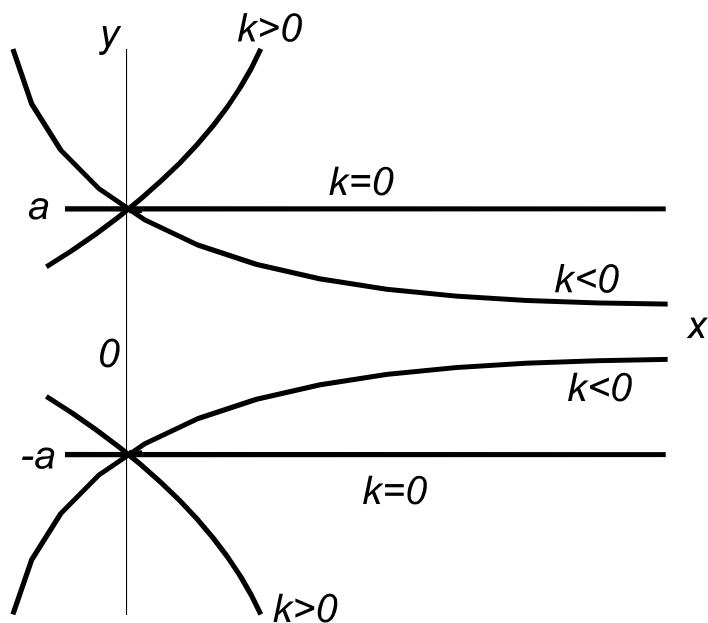


Рис. 2.3. График показательной функции

Найдем коэффициенты k и a для исходной табл. 2.1, если известно, что приближающую функцию целесообразно искать в виде показательной функции (2.9).

Прологарифмируем равенство (2.9) :

$$\ln y = \ln a + kx, \tag{2.10}$$

приняв обозначения $\ln y = q$, $\ln a = A$, перепишем (2.10) в виде:

$$q(x) = kx + A. \tag{2.11}$$

Таким образом приближающая показательная функция нехитрыми преобразованиями сведена к линейной, следовательно, для определения коэффициентов a и k показательной функции можно воспользоваться выведенной для линейной функции формулой (2.5).

Итак, для нахождения приближающей функции в виде (2.9) нужно прологарифмировать значения функции в исходной табл. 2.1 и, рассматривая их совместно с исходными значениями аргумента, построить для новой табл.2.3 приближающую функцию вида (2.11).

Таблица 2.3. Данные для нахождения приближающей функции в виде

$$y = f(x, a, k) = a \exp(kx) = ae^{kx}.$$

t	x ₁	x ₂	x ₃	...	x _n
q(t)	ln (y ₁)	ln (y ₂)	ln (y ₃)	...	ln (y _n)

Окончательно получаем:

$$k = \frac{n \sum_{i=1}^n x_i \ln y_i - \sum_{i=1}^n x_i \sum_{i=1}^n \ln y_i}{n \sum_{i=1}^n (x_i)^2 - \left(\sum_{i=1}^n x_i \right)^2}; \quad (2.12)$$

$$a = \exp \left(\frac{1}{n} \left(\sum_{i=1}^n \ln y_i - k \sum_{i=1}^n x_i \right) \right).$$

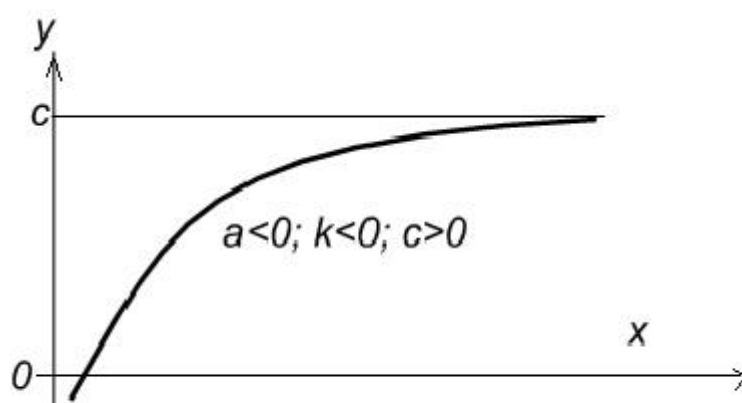


Рис. 2.4. График функции $y = a \exp(kx) + c$

Формулам

$$y = ax^k + c, \quad (2.13)$$

$$y = a \exp(kx) + c \quad (2.14)$$

соответствуют кривые, изображенные на рис. 2.2 и 2.3, сдвинутые вверх или вниз на величину c . Например, кривая, изображенная на рис. 2.4, соответствует формуле $y = a \exp(kx) + c$ при $a < 0, k < 0, c > 0$ и $|a| > c$. Чтобы

найти параметры этих формул, следует сначала определить значение c . Иногда величину c можно легко найти по значению, к которому стремится y при возрастании $y = kx + b$ (при $k < 0$) или по значению $y = cx^m$. при k, b, c (для формулы (2.13) при $k > 0$). Можно также воспользоваться формулой

$$c = \frac{y_1 y_2 - y_3^2}{y_1 + y_2 - 2y_3},$$

где y_1 и y_2 — ординаты произвольных (но достаточно далеких) точек с абсциссами x_1, x_2 , а ордината y_3 соответствует абсциссе $x_3 = \sqrt{x_1 x_2}$ в случае формулы (2.13) и абсциссе $x_3 = \frac{x_1 + x_2}{2}$ в случае формулы (2.14).

2.2.6. Пример поиска приближающей функции методом наименьших квадратов

Построим приближающую функцию методом наименьших квадратов для зависимости, заданной табл 2.4..

Таблица 2.4. Экспериментальные данные для построения приближающей функции

x	1.1	1.7	2.4	3.0	3.7	4.5	5.1	5.8
f(x)	0.3	0.6	1.1	1.7	2.3	3.0	3.8	4.6

Точечный график изображен на рис. 2.5. Вид приближающей кривой не очевиден, поэтому рассмотрим два способа приближения заданной

функции: в виде прямой $y = kx + b$ и в виде степенной функции $y = cx^m$. После нахождения значений параметров k, b, c и m найдем суммы квадратов отклонений (2.5) и по их значениям установим какое из двух приближений лучше.

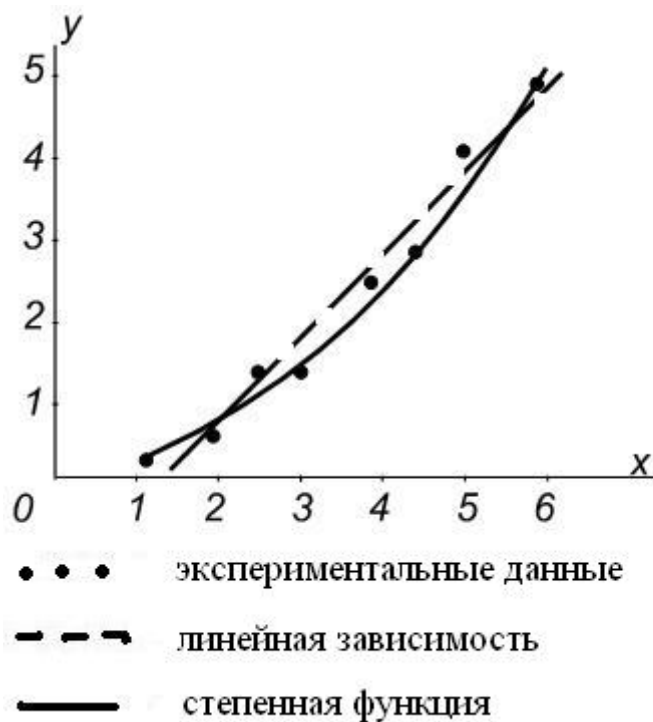


Рис. 2.5. Экспериментальные данные и подбор вида приближающей функции

Расчет сумм :

$$SX = \sum_{i=1}^n x_i, \quad SY = \sum_{i=1}^n y_i,$$

$$SXY = \sum_{i=1}^n x_i y_i, \quad SX^2 = \sum_{i=1}^n x_i^2$$

$$k = (N \cdot SXY - SX \cdot SY) / (N \cdot SX^2 - SX^2)$$

$$b = (SY - k \cdot SX) / N$$

Прделав необходимые вычисления, получаем:

$$k = 0,921, \quad b = -0,968, \quad \sigma(kb) = 0,2012.$$

Таким образом, приближающее линейное уравнение запишется в виде $y = 0,921x - 0,968$.

Для нахождения параметров c и m степенной функции воспользуемся формулой

$$c = \frac{n \sum_{i=1}^n \ln x_i \ln y_i - \sum_{i=1}^n \ln x_i \sum_{i=1}^n \ln y_i}{n \sum_{i=1}^n (\ln x_i)^2 - \left(\sum_{i=1}^n \ln x_i \right)^2},$$

$$m = \exp \left(\frac{1}{n} \left(\sum_{i=1}^n \ln y_i - k \sum_{i=1}^n \ln x_i \right) \right).$$

Подставив данные табл. 2.4, получим:

$$c = 0,257, \quad m = 1,656, \quad \sigma(cm) = 0,0425.$$

Таким образом, уравнение степенной регрессии имеет вид

$$y = 0,257x^{1,656}.$$

Как видно, сумма квадратов абсолютных погрешностей для линейной функции составляет $\sigma(kb) = 0,2012$, для степенной функции — $\sigma(cm) = 0,0425$. Видно, что приближение в виде степенной функции в данном случае предпочтительнее.

Для решения задачи приближения функции методом наименьших квадратов сформулируем основные шаги алгоритма:

- 1) ввод исходных данных;
- 2) выбор вида уравнения регрессии;
- 3) преобразование данных к линейному типу зависимости;
- 4) получение параметров уравнения регрессии;
- 5) обратное преобразование данных и вычисление суммы квадратов отклонений вычисленных значений функции от заданных.

2.2.7. Пример решения задачи курсового проекта в среде MathCad

Основная цель. Построить математическую модель, разработать алгоритм решения, составить программу решения. С помощью программы рассчитать заданный элемент теплоэнергетического оборудования [6,7].

Условие задачи

Подвеска пароперегревателя энергетического котла расположена вертикально в поворотном газоходе и представляет собой стержень длиной $L(\text{м}) = l$, выполненный из стали марки Ст. Верхний конец подвески крепится к каркасу котла и расположен за пределами обмуровки в воздушном пространстве котельного цеха. Нижний конец подвески крепится к трубам пароперегревателя для их поддержки и находится в потоке уходящих из топки котла дымовых газов. Котел находится в рабочем (горячем) состоянии, и в определенный момент времени температура известна в указанных точках подвески, считая от верхнего торца. Подобрать по исходным данным неизвестные коэффициенты функционального закона начального распределения температуры по длине подвески и на основании полученных результатов определить перераспределение температуры через заданные промежутки времени τ , если при аварийном останове котла значение температуры нижнего торца. $T_n = t_n$ °С подвески скачкообразно доходит до температуры, равной температуре верхнего торца подвески $T_0 = t_0$ °С. Рассчитать распределение температуры по длине подвески в заданные промежутки времени.

Что необходимо сделать:

1. Рассчитать распределение температуры по длине подвески в начальный момент времени.
2. Аппроксимировать полученное распределение температуры по длине подвески функцией, предложенной в задании.

3. Неизвестные коэффициенты аппроксимации найти по методу наименьших квадратов и с помощью одной из встроенных функций MathCad.
4. Построить график результатов аппроксимации и сделать выводы.
5. Аппроксимировать полученное распределение температуры по длине подвески наиболее подходящей функцией, предложенной в Excel.
6. Построить график результатов аппроксимации в Excel и сделать выводы.
7. Рассчитать распределение температуры по длине подвески в заданные промежутки времени.
8. Построить поле температур по длине подвески в фиксированные моменты времени.

Листинг решения задачи в MathCad:

Исходные данные:

Длина подвески – 3,25 м
 Нач. температура – 57,5 °С
 Кон. температура – 725 °С
 Ст.15
 $\lambda = 7682$
 $c = 0,682$

Вычислить распределение температуры по длине подвески через

$t_1 = 27,5$ с
 $t_2 = 22,5$ мин
 $t_3 = 9$ ч
 $t_4 = 25,5$ ч

Будем аппроксимировать полученные данные функцией

$$a \cdot k1 \cdot x + b \cdot k2 \cdot \exp(c \cdot k3 \cdot x)$$

где $k1 := 13$ $k2 := 1$ $k3 := 1$.

Скорректируем исходные данные с помощью функции

$$w(z) := 57.5 \cdot e^z + 13 \cdot z,$$

где $z := 0, 0.325 .. 3.25$

$$\begin{matrix} L := \\ \begin{pmatrix} 0 \\ 0.5 \\ 1 \\ 1.5 \\ 2 \\ 2.5 \\ 3 \\ 3.25 \end{pmatrix} \end{matrix} \qquad \begin{matrix} T := \\ \begin{pmatrix} 57.5 \\ 80 \\ 120 \\ 180 \\ 270 \\ 400 \\ 600 \\ 725 \end{pmatrix} \end{matrix}$$

Отообразим полученные данные и приближающую функцию графически (рис.2.6 и рис.2.7)

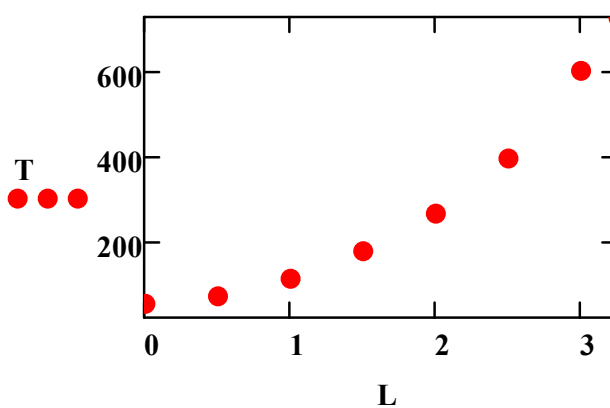


Рис 2.6. Распределение экспериментальных данных

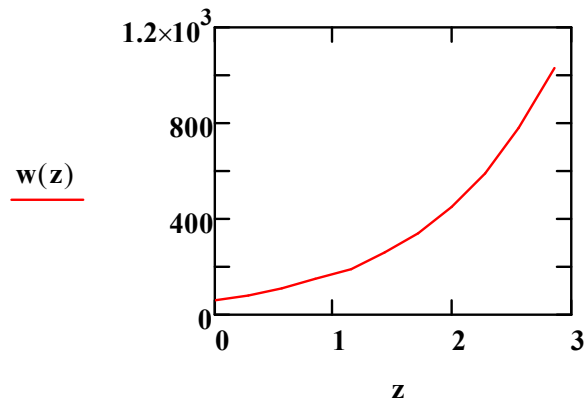


Рис. 2.7. Определение вида приближающей функции $w(z) := 57.5 \cdot e^z + 13 \cdot z$

Способ 1. Подбор приближающей функции методом наименьших квадратов с помощью встроенной функции genfit

Функция **genfit(X,Y, vs, F)** вычисляет параметры заданной функции *f*, определенной в *F* и наилучшим образом описывающей данные в *X,Y*. Вектор *vs* содержит начальные приближения параметров. *F* - вектор-функция, первый компонент которой есть функция *f*, а остальные - частные производные *f* по искомым параметрам [8].

Вычисляем частные производные по переменным:

$$F(x, a, b, c) := a \cdot k1 \cdot x + b \cdot k2 \cdot \exp(c \cdot k3 \cdot x)$$

$$\frac{d}{da} F(x, a, b, c) \rightarrow 13 \cdot x$$

$$\frac{d}{db} F(x, a, b, c) \rightarrow e^{c \cdot x}$$

$$\frac{d}{dc} F(x, a, b, c) \rightarrow b \cdot x \cdot e^{c \cdot x}$$

Составим вектор-функцию:

$$Z(x, a, b, c) := \begin{pmatrix} a \cdot k1 \cdot x + b \cdot k2 \cdot \exp(c \cdot k3 \cdot x) \\ 13 \cdot x \\ 1.0 \cdot e^{1.0 \cdot c \cdot x} \\ b \cdot x \cdot e^{1.0 \cdot c \cdot x} \end{pmatrix}$$

$$VS := \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix} \text{ - начальные приближения.}$$

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} := \text{genfit}(X, Y, VS, Z) = \begin{pmatrix} -0.135 \\ 55.831 \\ 0.792 \end{pmatrix}$$

Построим график полученного решения (рис.2.8):

$$f(x) := a \cdot k1 \cdot x + b \cdot k2 \cdot \exp(c \cdot k3 \cdot x)$$

$$x := 0, 0.325 .. 3.25$$

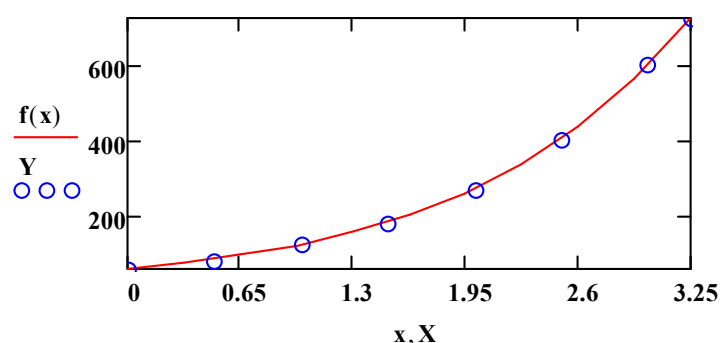


Рис. 2.8. Результат аппроксимации с помощью функции genfit

Способ 2. Метод наименьших квадратов и решение полученной системы уравнений с помощью блока Give-Minerr

Функция *Minerr* использует тот же самый алгоритм, что и функция *Find*. Различие состоит в следующем. Если в результате поиска решения не может быть получено дальнейшее уточнение текущего приближения к решению, *Minerr* возвращает это приближение.

Функция *Find*, в отличие от функции *Minerr*, возвращает в этом случае сообщение об ошибке “*решение не найдено*”. [8]

Метод наименьших квадратов:

ORIGIN:= 0

i := 0..7

Составим функцию, вычисляющую сумму квадратов отклонений ординат экспериментальных данных от приближающей функции.

$$\text{SSE}(\mathbf{a}, \mathbf{b}, \mathbf{c}) := \sum_i (Y_i - F(X_i, \mathbf{a}, \mathbf{b}, \mathbf{c}))^2$$

Given

$$\frac{d}{da} \text{SSE}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = 0$$

$$\frac{d}{db} \text{SSE}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = 0$$

$$\frac{d}{dc} \text{SSE}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = 0$$

$$\text{Minerr}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \begin{pmatrix} -0.135 \\ 55.831 \\ 0.792 \end{pmatrix}$$

Расчет распределения температур по длине подвески в разные моменты времени

ORIGIN:= 1

n := 1..20

Промежутки времени:

$$t_1 := 27.5$$

$$t_2 := 22.5 \cdot 60$$

$$t_3 := 9 \cdot 3600$$

$$t_4 := 25.5 \cdot 3600$$

Коэффициент теплопроводности:

$$a := \frac{\lambda \cdot 10^{-3}}{\rho \cdot C}$$

$$a = 8.169 \times 10^{-6}$$

В результате разложения функции начальной температуры в интеграл Фурье для стержня конечной длины с постоянными температурами на концах можно использовать решение в виде (рис. 2.9):

$$T_{11}(t, x) := T_1 + (T_2 - T_1) \cdot \frac{x}{L} + \frac{2}{\pi} \cdot \sum_n \left[(-1)^n \cdot \frac{T_2 - T_1}{n} \cdot \exp\left[-\left(\pi \cdot \frac{n}{L}\right) \cdot a \cdot t\right] \cdot \sin\left(\pi \cdot \frac{n}{L} \cdot x\right) \right] \dots$$

$$+ \frac{2}{L} \cdot \sum_n \left[\exp\left[-\left(\pi \cdot \frac{n}{L}\right) \cdot a \cdot t\right] \cdot \sin\left(\pi \cdot \frac{n}{L} \cdot x\right) \cdot \int_0^L f(x) \cdot \sin\left(\pi \cdot \frac{n}{L} \cdot x\right) dx \right]$$

$$T_{12}(t, x) := \text{if}(T_{11}(t, x) > 57.5, T_{11}(t, x), 57.5)$$

$$T_{12}(t_1, x) =$$

57.5
123.652
146.826
174.507
209.767
255.222
313.985
389.961
488.012
613.441
725

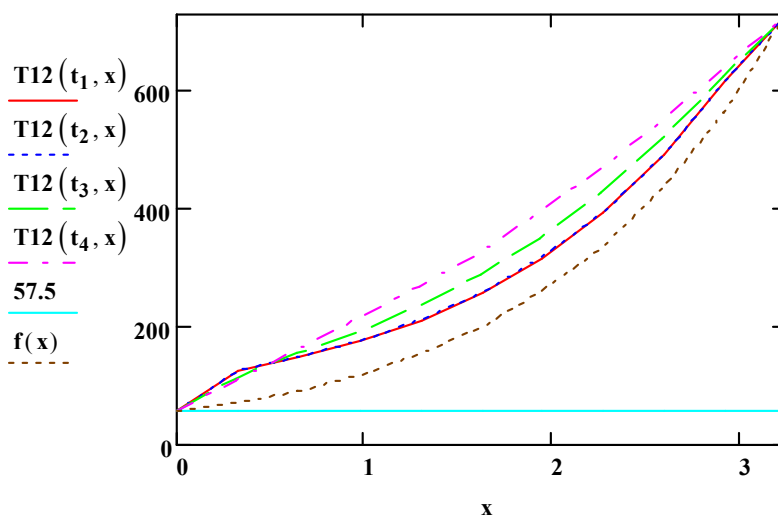


Рис. 2.9. Распределение температуры по длине подвески в разные промежутки времени

3. КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ЗАДАЧИ ДИРИХЛЕ

3.1. Численная постановка задачи

Уравнение Лапласа является модельным для эллиптических

уравнений в частных производных. Некоторые важные задачи, часто встречающиеся в приложениях, сводятся к решению одного эллиптического уравнения. К ним относятся задачи расчета дозвукового безвихревого (потенциального) течения газа и определения стационарного поля температуры в твердом теле [9,10].

Пусть требуется решить конечно-разностную задачу Дирихле для уравнения Лапласа в прямоугольной области, т.е. найти непрерывную функцию $u(x, y)$, удовлетворяющую внутри прямоугольной области $\Omega = \{(x,y) | 0 \leq x \leq a, 0 \leq y \leq b\}$ уравнению Лапласа

$$\Delta u \equiv \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (3.1)$$

и принимающую на границе области Ω заданные значения (рис.3.1), т. е.

$$u(0, y) = f_1(y), \quad y \in [0, b], \quad u(a, y) = f_2(y), \quad y \in [0, b],$$

$$u(x, 0) = f_3(x), \quad x \in [0, a], \quad u(x, b) = f_4(x), \quad x \in [0, a],$$

где f_1, f_2, f_3, f_4 — заданные функции.

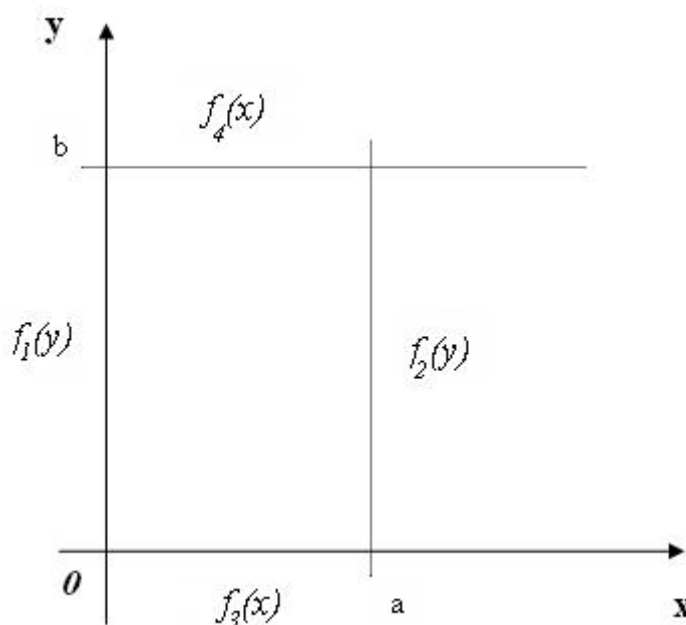


Рис 3.1. Граничные условия задачи Дирихле

Будем считать, что $u(x, y)$ непрерывна на границе области Ω , т. е.

$f_1(0) = f_3(0)$, $f_1(b) = f_4(0)$, $f_2(0) = f_3(a)$, $f_2(b) = f_4(a)$. Выбрав шаги h, l по x и y соответственно, строим сетку $x_i = ih$, $i = 0, 1, \dots, n$, $y_j = jl$, $j = 0, 1, \dots, m$, где $x_n = nh = a$, $y_m = ml = b$.

Вводя обозначения $u_{ij} = u(x_i, y_j)$, аппроксимируем частные производные $\frac{\partial^2 u}{\partial x^2}$ и $\frac{\partial^2 u}{\partial y^2}$ в каждом внутреннем узле сетки центральными разностными производными второго порядка:

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2} + O(h^2), \quad (3.2)$$

$$\frac{\partial^2 u}{\partial y^2} = \frac{u_{i,j+1} - 2u_{ij} + u_{i,j-1}}{l^2} + O(l^2)$$

и заменим уравнение Лапласа конечно-разностным уравнением

$$\frac{u_{i+1,j} - 2u_{ij} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{ij} + u_{i,j-1}}{l^2} = 0, \quad (3.3)$$

$$i = 1, \dots, n-1, \quad j = 1, \dots, m-1.$$

Погрешность замены дифференциального уравнения разностным составляет величину $O(h^2) + O(l^2)$.

Уравнения (3.3) вместе со значениями u_{ij} в граничных узлах образуют систему линейных алгебраических уравнений относительно приближенных значений функции $u(x, y)$ в узлах сетки (x_i, y_j) . Наиболее простой вид имеет эта система при $l = h$:

$$\begin{aligned} u_{ij} &= (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1})/4, \\ u_{i0} &= f_3(x_i), \quad u_{in} = f_4(x_i), \quad u_{0j} = f_1(y_j), \quad u_{mj} = f_2(y_j), \\ i &= 1, \dots, n-1, \quad j = 1, \dots, m-1. \end{aligned} \quad (3.4)$$

При получении сеточных уравнений (3.4) была использована схема узлов. Набор узлов, используемых для аппроксимации уравнения в точке, называется шаблоном. В данной работе используется шаблон типа «крест» (рис.3.2.).

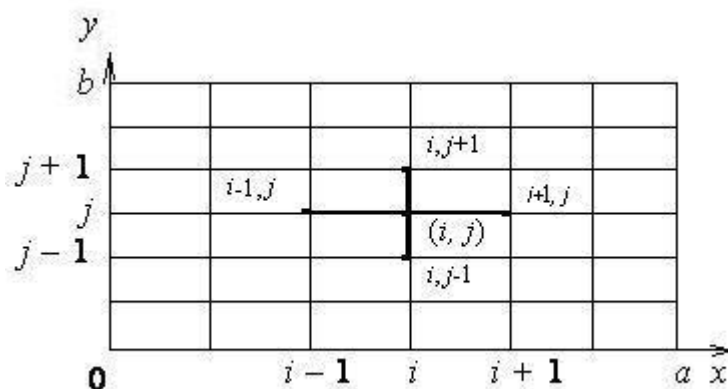


Рис. 3.2. Шаблон узлов типа «крест»

Численное решение задачи Дирихле для уравнения Лапласа в прямоугольнике состоит в нахождении приближенных значений u_{ij} искомой функции $u(x, y)$ во внутренних узлах сетки. Для определения величин u_{ij} требуется решить систему линейных алгебраических уравнений (3.4).

В данной работе она решается методом Зейделя, который состоит в построении последовательности итераций вида

$$u_{ij}^{(s+1)} = \frac{1}{4}(u_{i-1,j}^{(s+1)} + u_{i+1,j}^{(s)} + u_{i,j-1}^{(s)} + u_{i,j+1}^{(s+1)})$$

(верхним индексом s обозначен номер итерации). При $s \rightarrow \infty$ последовательность $u_{ij}^{(s)}$ сходится к точному решению системы (3.4). В качестве условия окончания итерационного процесса можно принять

$$\left| u_{ij}^{(s)} - u_{ij}^{(s+1)} \right| < \varepsilon, \quad 1 \leq i \leq n-1, \quad 1 \leq j \leq m-1.$$

Таким образом, погрешность приближенного решения, полученного методом сеток, складывается из двух погрешностей: погрешности аппроксимации дифференциального уравнения разностными уравнениями; погрешности, возникающей в результате приближенного решения системы разностных уравнений (3.4).

Известно, что описанная здесь разностная схема обладает свойствами устойчивости и сходимости. Устойчивость схемы означает, что малые изменения в начальных данных приводят к малым изменениям решения разностной задачи. Только такие схемы имеет смысл применять в реальных вычислениях. Сходимость схемы означает, что при стремлении шага сетки к нулю ($h \rightarrow 0$) решение разностной задачи стремится к решению исходной задачи. Таким образом, выбрав достаточно малый шаг h , можно как угодно точно решить исходную задачу.

3.2. Метод сеток

Используя метод сеток, составить приближенное решение задачи Дирихле для уравнения Лапласа (3.1).

Решение получить в квадрате ABCD с вершинами $A(0;0)$, $B(0;1)$, $C(1;1)$, $D(1;0)$, шаг $h=l=0,2$; $U|_{AB} = 45y(1-y)$; $U|_{BC} = 25x$; $U|_{CD} = 25$; $U|_{AD} = 25x \cdot \sin \pi x$. Систему линейных алгебраических уравнений решить по методу Зейделя, при $\varepsilon=0,01$.

Для решения этой задачи необходимо найти значение функции в ее узлах: $U1, U2, \dots, U16$.

1) Построим сетку с шагом $h=l=0,2$ (рис.3.3.)

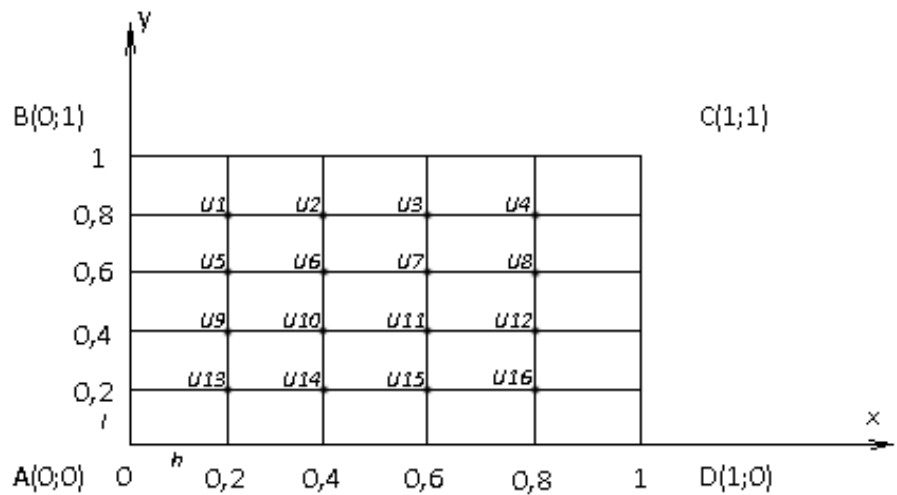


Рис 3.3. Графическое изображение метода сеток

2) Вычислим значение функции $U|_{AB}$, $U|_{BC}$, $U|_{CD}$, $U|_{AD}$, построим сетку с граничными условиями (рис.3.4.):

на стороне AB:

$$U(y=0)=0; \quad U(y=0,2)=7,2; \quad U(y=0,4)=10,8; \quad U(y=0,6)=10,8; \quad U(y=0,8)=7,2; \\ U(y=1)=0;$$

на стороне BC:

$$U(x=0,2)=5; \quad U(x=0,4)=10; \quad U(x=0,6)=15; \quad U(x=0,8)=20; \\ U(x=1)=25;$$

на стороне CD:

$$U(y=0,2)=25; \quad U(y=0,4)=25; \quad U(y=0,6)=25; \quad U(y=0,8)=25;$$

на стороне AD:

$$U(x=0,2)=1,544; \quad ; \quad U(x=0,4)=5,875; \quad U(x=0,6)=12,131; \\ U(x=0,8)=19,017; \quad U(x=1)=25.$$

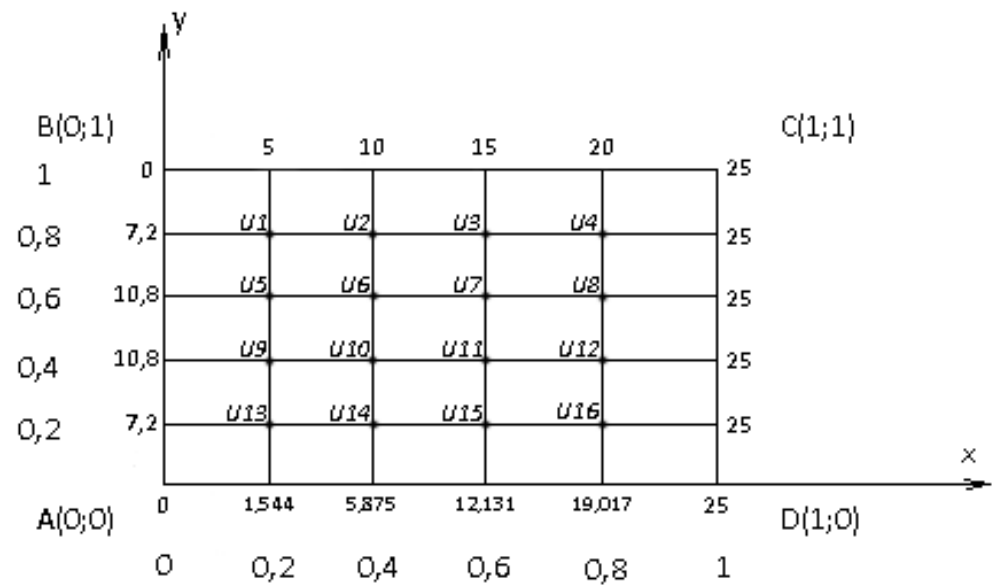


Рис. 3.4. Сетка с вычисленными значениями функции в узлах по границе

3) Составим систему линейных алгебраических уравнений (см. формулу 3.4)

Будем считать, что начальные условия - значения функции в узловых точках равны 0:

$$U_1, U_2, U_3, U_4, U_5, U_6, U_7, U_8, U_9, U_{10}, U_{11}, U_{12}, U_{13}, U_{14}, U_{15}, U_{16} = 0.$$

Составленная система решается, например, итерационным методом

Зейделя.

$$\left\{ \begin{array}{l} U_1 = \frac{1}{4}(5 + 7,2 + U_2 + U_5) \\ U_2 = \frac{1}{4}(10 + U_1 + U_6 + U_3) \\ U_3 = \frac{1}{4}(15 + U_2 + U_4 + U_7) \\ U_4 = \frac{1}{4}(20 + 25 + U_3 + U_8) \\ U_5 = \frac{1}{4}(10,8 + U_1 + U_6 + U_3) \\ U_6 = \frac{1}{4}(U_5 + U_2 + U_7 + U_{10}) \\ U_7 = \frac{1}{4}(U_3 + U_6 + U_8 + U_{11}) \\ U_8 = \frac{1}{4}(U_4 + U_7 + U_{12} + 25) \\ U_9 = \frac{1}{4}(10,8 + U_5 + U_{10} + U_{13}) \\ U_{10} = \frac{1}{4}(U_6 + U_9 + U_{11} + U_{14}) \\ U_{11} = \frac{1}{4}(U_7 + U_{12} + U_{15} + U_{10}) \\ U_{12} = \frac{1}{4}(U_8 + U_{11} + U_{16} + 25) \\ U_{13} = \frac{1}{4}(U_9 + U_{14} + 7,2 + 1,544) \\ U_{14} = \frac{1}{4}(U_{13} + U_{10} + U_{15} + 5,875) \\ U_{15} = \frac{1}{4}(U_{11} + U_{14} + U_{16} + 12,131) \\ U_{16} = \frac{1}{4}(U_{15} + U_{12} + 19,017 + 25) \end{array} \right.$$

3.3. Решение задачи курсового проекта в среде MathCad

Задача:

Построить математическую модель, разработать алгоритм, составить программу и рассчитать заданный элемент теплоэнергетического оборудования [6,7]. Варианты заданий представлены в табл. 3.1 -3.4 .

Формулировка задачи:

Топка энергетического котла изнутри экранирована газоплотными вертикальными панелями. Панель экрана представляет собой ряд параллельно расположенных экранных труб, соединенных между собой по образующим с помощью сварки стальными плоскими проставками (полосами), и имеет форму прямоугольника шириной b и высотой h . Питательная вода параллельными потоками подается в трубы на нижней границе экранной панели. Вследствие высоких значений коэффициентов теплоотдачи со стороны воды, можно принимать температуру стенки экранной панели равной температуре воды. В режиме устойчивого процесса горения в топочном объеме котла, вследствие неравномерного распределения температуры факела, температура воды на нижней границе экранной панели распределена по закону $t_0 = f_1(x)$. По мере продвижения воды вверх по трубам температура воды увеличивается и на верхней границе панели изменяется по закону $t_1 = f_2(x)$. При этом изменение (увеличение) температуры стенки на левой и правой сторонах экранной панели тождественно и описывается уравнениями $t_3 = f_4(y)$, $t_2 = f_3(y)$.

Определить распределение температуры по поверхности экранной панели в установившемся состоянии.

Варианты исходных данных

Таблица 3.1. Размеры и материал пластины

№ варианта	1	2	3	4	5	6	7	8	9	10
В, м	3,5	4,5	5,5	4,0	6,0	6,5	8,0	8,5	7,0	9,0
Н, м	3,5	4,5	5,5	4,0	6,0	6,5	8,0	8,5	7,0	9,0
Материал: котельная сталь	Ст20К	Ст22К	Ст20К	Ст22К	Ст20К	Ст22К	Ст20К	Ст22К	Ст20К	Ст22К

Таблица 3.2. Законы распределения температуры по границам панели

№ варианта	$t_0 = f_1(x)$ на нижней части панели	$t_1 = f_2(x)$ на верхней части панели	$t_2 = f_3(x)$ справа панели	$t_3 = f_4(x)$ слева панели
1	$a_1 \cdot k1 \cdot x + 4$	$a_2 \cdot k1 \cdot x + 4$	$a_3 \cdot k1 \cdot x^2 + b_3$	$a_4 \cdot k2 \cdot x^2 + b_4$
2	$a_1 \cdot k1 \cdot x^2 + 5$	$a_2 \cdot k1 \cdot x + 4$	$a_3 \cdot k1 \cdot x^2 + b_3$	$a_4 \cdot k2 \cdot x^2 + b_4$
3	$a_1 \cdot k1 \cdot x + 6$	$a_2 \cdot k1 \cdot x^2 + 4$	$a_3 \cdot k1 \cdot x^2 + b_3$	$a_4 \cdot k2 \cdot x + b_4$
4	$a_1 \cdot k1 \cdot x^2 + 8$	$a_2 \cdot k1 \cdot x + 4$	$a_3 \cdot k1 \cdot x + b_3$	$a_4 \cdot k2 \cdot x^2 + b_4$
5	$a_1 \cdot k1 \cdot x + 27$	$a_2 \cdot k1 \cdot x^2 + 4$	$a_3 \cdot k1 \cdot x^2 + b_3$	$a_4 \cdot k2 \cdot x + b_4$
6	$a_1 \cdot k1 \cdot x + 11$	$a_2 \cdot k1 \cdot x + 4$	$a_3 \cdot k1 \cdot x + b_3$	$a_4 \cdot k2 \cdot x^2 + b_4$
7	$a_1 \cdot k1 \cdot x^2 + 20$	$a_2 \cdot k1 \cdot x + 4$	$a_3 \cdot k1 \cdot x^2 + b_3$	$a_4 \cdot k2 \cdot x + b_4$
8	$a_1 \cdot k1 \cdot x^2 + 4$	$a_2 \cdot k1 \cdot x + 4$	$a_3 \cdot k1 \cdot x^2 + b_3$	$a_4 \cdot k2 \cdot x^2 + b_4$
9	$a_1 \cdot k1 \cdot x + 25$	$a_2 \cdot k1 \cdot x^2 + 40$	$a_3 \cdot k1 \cdot x + b_3$	$a_4 \cdot k2 \cdot x^2 + b_4$
10	$a_1 \cdot k1 \cdot x^2 + 1$	$a_2 \cdot k1 \cdot x + 4$	$a_3 \cdot k1 \cdot x + b_3$	$a_4 \cdot k2 \cdot x + b_4$

Таблица 3.3. Начальное и конечное значения температуры по длине стороны панели

№ вар.	1	2	3	4	5	6	7	8	9	10
t_0	110	120	130	140	150	160	170	180	160	140
t_1	200	220	240	240	250	250	260	280	280	200

Таблица 3.4. Теплофизические свойства материалов

Материал	λ , Вт/(м ⁰ С)	C , кДж/(кг ⁰ С)	ρ , кг/м ³
Ст20К	35,0	0,645	7840
Ст22К	32,0	0,635	7880

Распределение температуры по поверхности экранной панели в установившемся состоянии находится из численного решения уравнения Лапласа методом сеток двумя способами. Первый способ основывается на методе сеток, запрограммированном в MathCad. Второй способ является проверочным и основывается на использовании встроенных функций MathCad для решения уравнения Лапласа.

Описание решения в среде MathCad

Введем обозначения:

n - число разбиений сетки, $n:=4$

h – высота граничной области в форме прямоугольника, $h:=3.5$

b – ширина граничной области, $b:=3.5$

Предельное значение температуры по горизонтальной границе $t_{0,4}:=110$,

$t_{4,4}:=200$

i – номер строки сетки $i:=0..4$, $x_j := \frac{b}{4} \cdot j$;

j – номер столбца сетки $j:=0..4$, $y_i := \frac{h}{4} \cdot i$.

Законы распределения температуры по границам панели:

на нижней части панели: $t_0=f_1(x)=a_1 \cdot x+4$;

на верхней части панели: $t_1=f_2(x)=a_2 \cdot x+4$;

справа панели: $t_2=f_3(x)= a_3 \cdot x^2+b_3$;

слева панели: $t_3=f_4(x)= a_4 \cdot x^2+b_4$

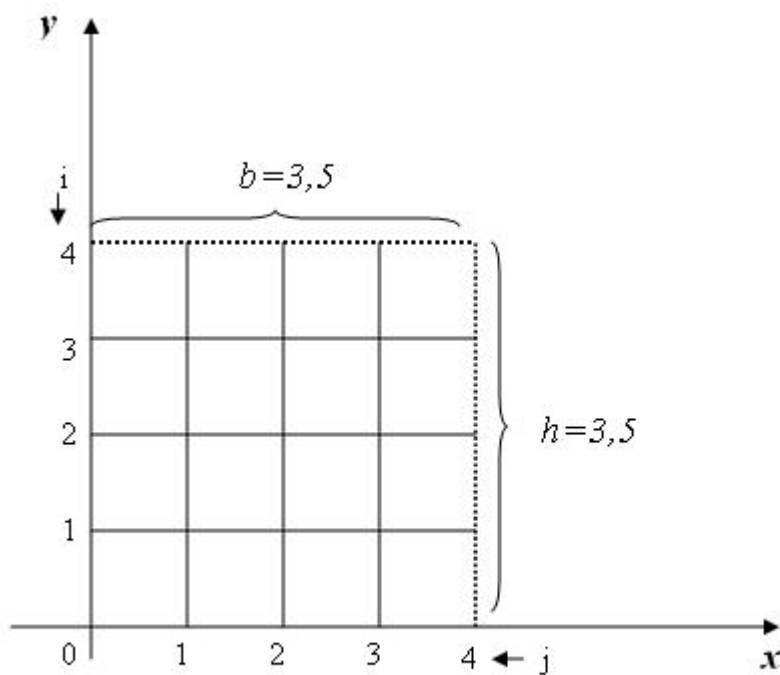


Рис. 3.5. Разбиение по методу сеток

Рассчитаем значения в узлах сетки (рис.3.5) на горизонтальных сторонах пластины:

$$F(t) := a_1 \cdot t + 4$$

$$F_1(t) := a_2 \cdot t + 4$$

Так как предельное значение температуры по горизонтальной границе

$$t_{0,4} := 110, t_{4,4} := 200:$$

$$a1 \cdot b + 4 = t0$$

$$a1 := \frac{t0 - 4}{b}$$

$$a2 \cdot b + 4 = t1$$

$$a2 := \frac{t1 - 4}{b}$$

$$t_{4,j} := F(x_j)$$

$$t_{0,j} := F1(x_j)$$

$$x_j := \frac{b}{4} \cdot j$$

Результат по горизонтали:

$$t = \begin{pmatrix} 4 & 53 & 102 & 151 & 200 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 4 & 30.5 & 57 & 83.5 & 110 \end{pmatrix}$$

Аналогично проводим расчет значений в узлах сетки (рис.3.5) по

вертикальным сторонам:

$$y_i := \frac{h}{4} \cdot i$$

$$a3 \cdot 0 + b3 = t0$$

$$b3 = t0$$

$$a3 := \frac{t0 - t1}{h^2}$$

$$a3 \cdot h^2 + b3 = t0$$

$$F(t) := a3 \cdot t^2 + t1 - \text{правая сторона};$$

$$F1(t) := \frac{4-t}{h^2} \cdot t^2 + 4$$

- левая сторона

$$t_{i,4} := F(y_i)$$

$$t_{i,0} := F1(y_i)$$

Результат по вертикали:

$$t = \begin{pmatrix} 4 & 53 & 102 & 151 & 200 \\ 4 & 0 & 0 & 0 & 194.375 \\ 4 & 0 & 0 & 0 & 177.5 \\ 4 & 0 & 0 & 0 & 149.375 \\ 4 & 30.5 & 57 & 83.5 & 110 \end{pmatrix}$$

Для решения системы линейных уравнений и расчета значений во внутренних узлах сетки используем возможности программирования в среде

MathCad:

```

S(n) :=
  eps ← 10-5
  T ← t
  for i ∈ 1..n-1
    for j ∈ 1..n-1
      Ti,j ← 0.25 · (ti-1,j + ti+1,j + ti,j-1 + ti,j+1)
  for i ∈ 1..n-1
    for j ∈ 1..n-1
      while |Ti,j - ti,j| > eps
        for i ∈ 1..n-1
          for j ∈ 1..n-1
            ti,j ← Ti,j
            Ti,j ← 0.25 · (ti-1,j + ti+1,j + ti,j-1 + ti,j+1)
  return T

```


Результат работы программы:

$$S(4) = \begin{pmatrix} 4 & 53 & 102 & 151 & 200 \\ 4 & 48.731 & 94.266 & 141.91 & 194.375 \\ 4 & 43.658 & 84.422 & 127.998 & 177.5 \\ 4 & 37.481 & 71.766 & 108.16 & 149.375 \\ 4 & 30.5 & 57 & 83.5 & 110 \end{pmatrix}$$

Проверка полученного результата посредством встроенной функции

relax.

$\text{relax}(a,b,c,d,e,F,v,rjас)$ — матрица решения дифференциального уравнения в частных производных на квадратной области, полученного с помощью алгоритма релаксации для метода сеток:

- a,b,c,d,e — квадратные матрицы коэффициентов разностной схемы, аппроксимирующей дифференциальное уравнение;
- F — квадратная матрица, задающая правую часть дифференциального уравнения;
- v — квадратная матрица граничных условий и начального приближения к решению;
- $rjас$ — параметр численного алгоритма (спектральный радиус итераций Якоби).

Параметр численного алгоритма характеризует скорость сходимости итераций. Он должен быть числом от 0 до 1. В матрице граничных условий v необходимо задать только граничные элементы, исходя из значений краевых условий по периметру расчетной области. Прочие (внутренние) элементы этой матрицы служат для задания начального приближения к решению.

Суть алгоритма релаксации сводится к тому, что в ходе итераций происходят проверка уравнений и соответствующая коррекция значений искомой функции в каждой точке. Если начальное приближение выбрано удачно, то можно надеяться, что алгоритм сойдется ("релаксирует") к правильному решению.

$i := 0..4$

$j := 0..4$

$a_{i,j} := 1$

$b_{i,j} := a$

$c_{i,j} := a$

$d := a$

$e_{i,j} := -4 \cdot a$

$v := t1$

$f_{i,j} := 0$

$s := \text{relax}(a, b, c, d, e, f, v, .95)$

Матрица решения:

$$s = \begin{pmatrix} 4 & 53 & 102 & 151 & 200 \\ 4 & 48.731 & 94.266 & 141.909 & 194.375 \\ 4 & 43.658 & 84.422 & 127.998 & 177.5 \\ 4 & 37.481 & 71.766 & 108.159 & 149.375 \\ 4 & 30.5 & 57 & 83.5 & 110 \end{pmatrix}$$

ПРИЛОЖЕНИЯ

Приложение 1.

Компьютерная математика и математические пакеты

При использовании вычислительной техники встала проблема реализации алгоритмов решения в виде так называемых программ. Для

решения этой проблемы в различные годы использовались следующие средства:

- программирование в машинных кодах (включая языки типа Ассемблер);
- программирование на языках высокого уровня (включая объектно-ориентированное программирование);
- системы компьютерной математики.

Разработка программы (даже с использованием языков высокого уровня с приставками Visual) требует и соответствующей подготовки (назовем ее «программистской»), и достаточно большого количества времени (и то, и другое часто отсутствует у «обычного пользователя»). Поэтому, начиная с 90-х гг. прошлого века, широкую известность и заслуженную популярность приобрели так называемые системы компьютерной математики или, проще, математические пакеты [3].

Наиболее популярными из них являются Maple, MatLab, MathCad.

Математический пакет MathCad: возможности, структура

Пакет MathCad можно отнести к математически универсальным системам. Объединение этих слов означает, что, помимо собственных вычислений (как численных, так и символьных), MathCad позволяет подготавливать качественные тексты - дипломные и курсовые проекты, диссертации, статьи с наглядным графическим представлением результатов вычислений (в том числе и анимации).

Применение библиотек и пакетов расширений обеспечивает профессиональную ориентацию MathCad на любую область науки, техники и образования.

Пакет MathCad, в отличие от других современных математических пакетов, построен в соответствии с принципом WYSIWYG (“What You See Is What You Get” – “Что видите, то и получите”). Поэтому он прост в использовании, в частности, из-за отсутствия необходимости сначала писать программу, а потом транслировать, компилировать и запускать ее на исполнение.

MathCad представляет собой интегрированную систему, включающую связанные между собой компоненты (такие как текстовый редактор, вычислительный процессор, символьный процессор) [3].

Приложение 2.

Массивы в MathCad

Столбец чисел называется вектором, а прямоугольная таблица чисел - матрицей. Общий термин для вектора или матрицы - массив. При работе с матрицами используется панель инструментов “Матрицы” (рис.4.1.):



Рис 4.1. Элементы панели «Массив»

Обращение к элементам массива

Обращение к элементу массива осуществляется путем записи имени массива и соответствующих индексных выражений, количество которых определяется размерностью массива.

Для ввода индексных выражений обязательно нажать клавишу [– левую квадратную скобку. При этом курсор перемещается вниз, и индексные выражения оказываются смещенными относительно имени массива.

Начальное значение индексных выражений определяется системной переменной **ORIGIN**, и по умолчанию ее значение равно 0.

Верхний индекс позволяет обратиться к отдельному столбцу массива.

Чтобы вставить верхний индекс, надо ввести имя массива, а затем нажать клавиши <ctrl>+6 или нажать на кнопку $M^{<>}$.

Функции размера и диапазона значений массива

В MathCad есть несколько функций, которые возвращают информацию относительно размеров массива и диапазона его элементов:

- **rows(A)** – число строк в массиве **A**;
- **cols(A)** – число столбцов в массиве **A**;
- **length(v)** – число элементов в векторе **v**;
- **last(v)** – индекс последнего элемента в векторе **v**;
- **max(A)** – определяет самый большой элемент в массиве **A**; если **A** имеет комплексные элементы, то возвращает наибольшую вещественную часть плюс i , умноженную на наибольшую мнимую часть;

- **min(A)** – определяет самый маленький элемент в массиве **A**; если **A** имеет комплексные элементы, то возвращает наименьшую вещественную часть плюс **i**, умноженную на наименьшую мнимую часть [11].

Функции формирования специального типа матриц

- **identity(n)** – формирует n x n единичную матрицу (матрица, все диагональные элементы которой равны 1, а все остальные элементы равны 0);

Пример 1:

$$\mathbf{identity}(3) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \leftarrow \text{Единичная матрица}$$

- **geninv(A)** – формирует левую обратную к **A** матрицу **L** такую, что **L·A = I**, где **I** - единичная матрица, имеющая то же самое число столбцов, что и **A**.

Матрица **A** - m x n вещественная матрица, где m >= n;

Пример 2

$$\mathbf{L} := \text{geninv}(\mathbf{M}) \quad \mathbf{L} = \begin{pmatrix} -1.1 & -1.5 & 1.7 \\ 0.6 & 0 & -0.2 \\ -0.1 & 0.5 & -0.3 \end{pmatrix}$$
$$\mathbf{L} \cdot \mathbf{M} = \begin{pmatrix} 1 & 0 & -2.442 \times 10^{-15} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- **Re(A)** – формирует массив, состоящий из элементов, которые являются вещественными частями элементов A;
- **Im(A)** – формирует массив, состоящий из элементов, которые являются мнимыми частями элементов A;
- **diag(v)** – формирует диагональную матрицу, содержащую на диагонали элементы вектора **v**;

Пример 3

$$\mathbf{b} = \begin{pmatrix} 4 \\ 5 \\ 7 \end{pmatrix} \quad \text{diag}(\mathbf{b}) = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 7 \end{pmatrix} \quad \leftarrow \text{Диагональная матрица, полученная из вектора } \mathbf{b}$$

- **matrix(m,n,f)** – создает и заполняет матрицу размером $m \times n$, элемент которой, расположенный в строке i и столбце j , равен $f(i,j)$ [11].

Функции формирования новых массивов из существующих

- **augment (A, B)** - формирует массив расположением **A** и **B** бок о бок, причем массивы **A** и **B** должны иметь одинаковое число строк;
- **stack (A, B)** - формирует массив, расположением **A** над **B**, причем массивы **A** и **B** должны иметь одинаковое число столбцов;
- **submatrix (A, ir, jr, ic, jc)** - формирует подматрицу, содержащую строки с **ir** по **jr** и столбцы с **ic** по **jc** матрицы **A**;

Приложение 3.

Решение нелинейных уравнений и систем уравнений в пакете MathCad

Вычисление корней численными методами включает два основных этапа:

- *отделение корней;*
- *уточнение корней до заданной точности.*

Рассмотрим эти два этапа подробно.

Отделение корней нелинейного уравнения

Учитывая легкость построения графиков функций в MathCad, в дальнейшем будет использоваться графический метод отделения корней.

Пример 4. Дано алгебраическое уравнение

$$x^3 + 3x^2 - 3 = 0$$

Определить интервалы локализации корней этого уравнения (рис. 4.2.).



Рис 4.2. График функции $f(x) = x^3 + 3x^2 - 3$

Пример 5. Дано алгебраическое уравнение

$$x^3 - 6x^2 + 21x + 52 = 0.$$

Определить интервалы локализации корней этого уравнения.

На рис. 4.3. приведен график функции $f(x) = x^3 - 6x^2 + 21x + 52$, построенный в MathCad. Видно, что в качестве интервала локализации можно принять интервал $[-2, 0]$.

Однако уравнение имеет три корня. Следовательно, можно сделать вывод о наличии еще двух комплексных корней.

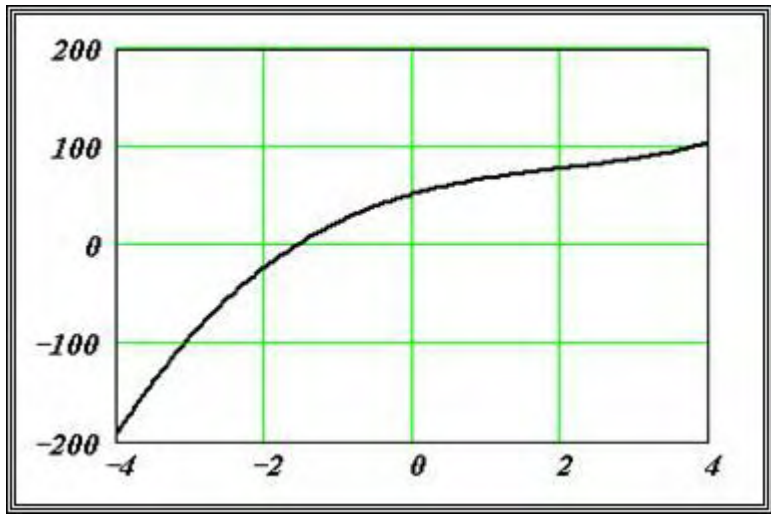


Рис.4.3. График функции $f(x) = x^3 - 6x^2 + 21x + 52$

Уточнение корней нелинейного уравнения

Для уточнения корня используются специальные вычислительные методы такие, как метод деления отрезка пополам, метод хорд, метод касательных (метод Ньютона) и многие другие [11].

Функция root. В MathCad для уточнения *корней любого нелинейного уравнения* (не обязательно только алгебраического) введена функция *root*,

которая может иметь два или четыре аргумента, т.е. $\boxed{\boxed{root(f(x),x)}}$

или $\boxed{\boxed{root(f(x),x,a,b)}}$, где $f(x)$ – имя функции или арифметическое выражение, соответствующее решаемому нелинейному уравнению, x – скалярная переменная, относительно которой решается уравнение, a, b – границы интервала локализации корня.

Пример 6. Используя функцию $root(f(x),x)$, найти все три корня уравнения $x^3 - 6x^2 + 21x + 52 = 0$, включая и два комплексных.

Решение (рис. 4.4):

$$a_3 := 1 \quad a_2 := -6 \quad a_1 := 21 \quad a_0 := 52$$

$$f(x) := a_3 \cdot x^3 + a_2 \cdot x^2 + a_1 \cdot x + a_0$$

Вычисление первого корня

$$x := -2 \quad \text{Задание "стартового" значения для первого корня}$$

$$x_1 := \text{root}(f(x), x) \quad x_1 = -1.578 \quad |f(x_1)| = 9.404 \times 10^{-7}$$

Вычисление второго корня

$$i := \sqrt{-1} \quad x := 1 + i \quad \text{Задание "стартового" значения для второго корня}$$

$$x_2 := \text{root}\left(\frac{f(x)}{x - x_1}, x\right) \quad x_2 = 3.789 + 4.313i$$

$$|f(x_2)| = 8.191 \times 10^{-6}$$

Вычисление третьего корня

$$i := \sqrt{-1} \quad x := 1 - i \quad \text{Задание "стартового" значения для третьего корня}$$

$$x_3 := \text{root}\left[\frac{f(x)}{(x - x_1) \cdot (x - x_2)}, x\right] \quad x_3 = 3.789 - 4.313i$$

$$|f(x_3)| = 2.047 \times 10^{-6}$$

Рис. 4.4. Решение примера 6 в среде MathCad

Функция *root* с двумя аргументами требует задания (до обращения к функции) переменной *x* начального значения корня из интервала локализации.

Пример 7. Используя функцию *root*, вычислить изменения корня нелинейного уравнения $e^x - ax^2 = 0$ при изменении коэффициента *a* от 1 до 10 с шагом 1.

Решение (рис. 4.5):

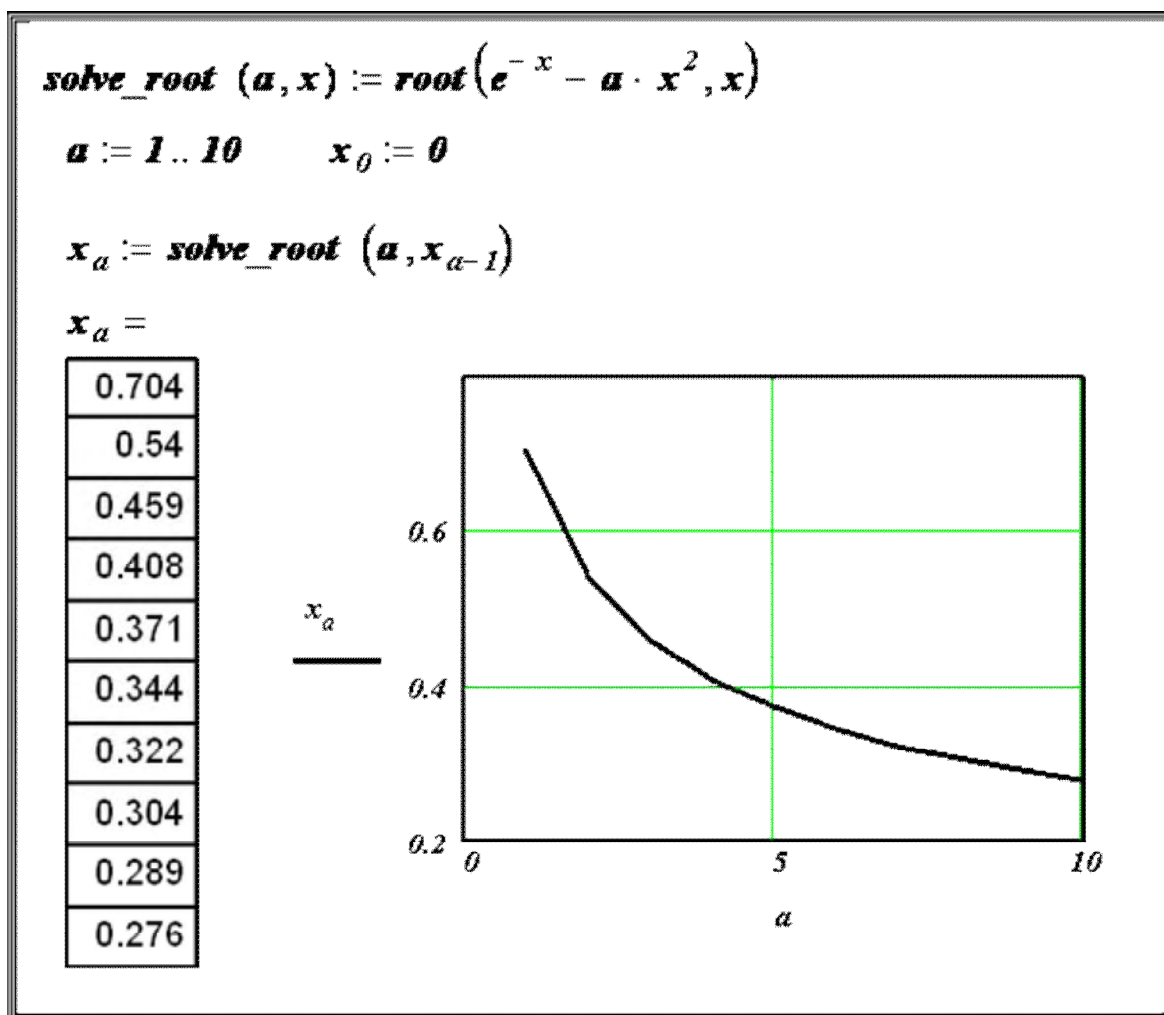


Рис. 4.5. Решение примера 7 в среде MathCad

Функция *polyroots*. Для вычисления всех корней алгебраического уравнения порядка n (не выше 5) рекомендуется использовать функцию *polyroots*. Обращение к этой функции имеет вид $\mathbf{polyroots(v)}$, где v – вектор, состоящий из $n+1$ проекций, равных коэффициентам алгебраического уравнения, т.е. $\mathbf{v_0 = a_0, v_1 = a_1, \dots, v_n = a_n}$. Эта функция не требует проведения процедуры локализации корней.

Пример 8. Используя функцию *polyroots*, найти все три корня уравнения $x^3 - 6x^2 + 21x + 52 = 0$, включая и два комплексных

Решение (рис. 4.6):

$$a_3 := 1 \quad a_2 := -6 \quad a_1 := 21 \quad a_0 := 52$$

$$v := \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} \quad \text{polyroots}(v) = \begin{pmatrix} -1.578 \\ 3.789 - 4.313i \\ 3.789 + 4.313i \end{pmatrix}$$

Рис. 4.6. Решение примера 8 в среде MathCad

Блок Given. При уточнении корня нелинейного уравнения можно использовать специальный вычислительный блок *Given*, имеющий следующую структуру:

```

⟨ Начальные условия ⟩
Given
⟨ Равенство ⟩
⟨ Ограничения ⟩
⟨ Вызов функции Find или Minerr ⟩

```

Решаемое уравнение задается в виде равенства, в котором используется «жирный» знак равно, вводимый с палитры ЛОГИЧЕСКИЙ.

Ограничения содержат равенства или неравенства, которым должен удовлетворять искомый корень.

Функция *Find* уточняет корень уравнения, вызов этой функции имеет вид *Find(x)*, где x – переменная, по которой уточняется корень. Если корня уравнения на заданном интервале не существует, то следует вызвать функцию *Minerr(x)*, которая возвращает *приближенное значение корня*.

Для выбора алгоритма уточнения корня необходимо щелкнуть правой кнопкой мыши на имени функции *Find(x)* и в появившемся контекстном меню (рис 4.7) выбрать подходящий алгоритм.

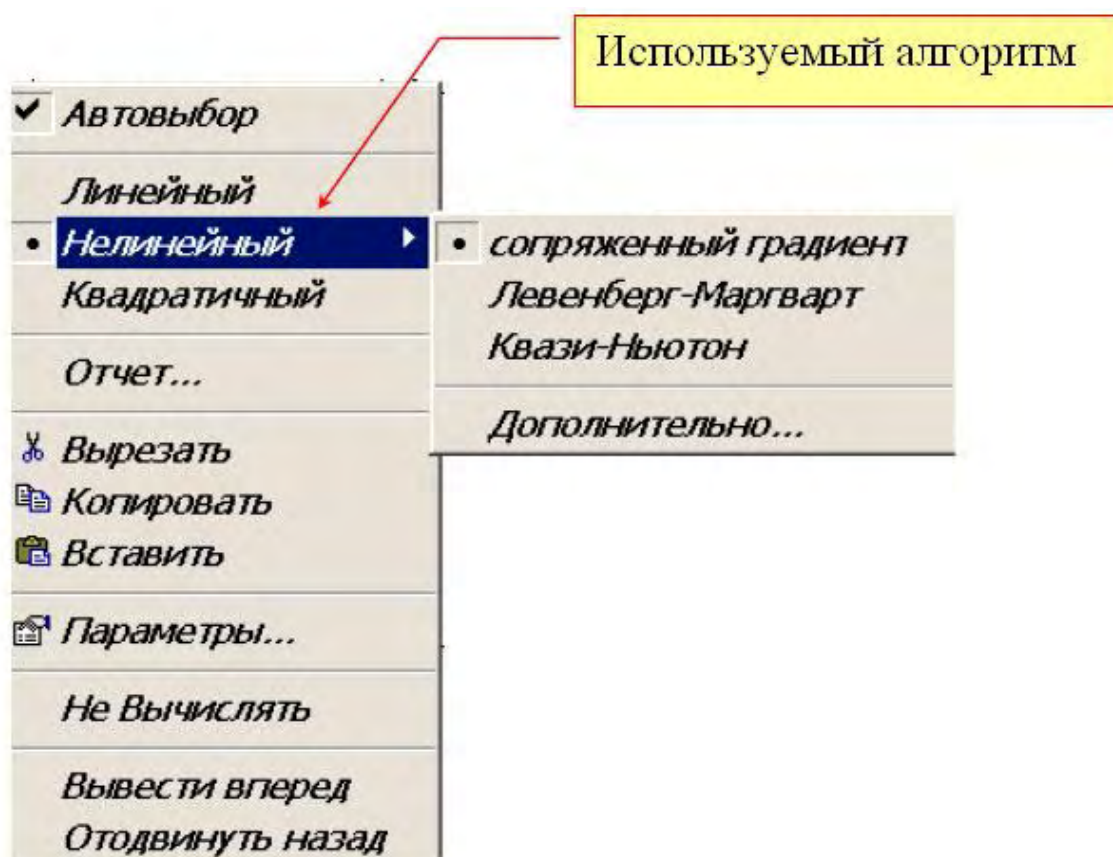


Рис. 4.7. Выбор алгоритма уточнения корня

Аналогично можно задать алгоритм решения и для функции *Minerr(x)*.

Использование численных методов в функциях *Find(x)*, *Minerr(x)* требует перед блоком *Given* задать начальные значения переменным, по которым осуществляется поиск корней уравнения.

Пример 9. Используя блок *Given*, вычислите корень уравнения $x^3 - 6x^2 + 21x + 52 = 0$ в интервале от деления $[-5, -1]$.

Решение (рис. 4.8):

```
a3 := 1    a2 := -6    a1 := 21    a0 := 52
f(x) := a3 · x^3 + a2 · x^2 + a1 · x + a0
x := -2    Задание начального значения
Given
f(x) = 0    Задание нелинейного уравнения
-4 ≤ x ≤ -1    Задание интервала от деления
Find(x) = -1.578    Вычисленное значение корня
```

Рис. 4.8. Решение примера 10 в среде MathCad

Приложение 4.

Решение систем уравнений

В зависимости от того, какие функции входят в систему уравнений, можно выделить два класса систем:

- алгебраические системы уравнений;
- трансцендентные системы уравнений.

Среди алгебраических систем уравнений особое место занимают системы линейных алгебраических уравнений (СЛАУ) [11].

Системы линейных алгебраических уравнений

Системой линейных алгебраических уравнений /СЛАУ/ называется система вида:

$$\begin{aligned} a_{1,1}x_1 + a_{1,2}x_2 + \dots + a_{1,m}x_m &= b_1; \\ a_{2,1}x_1 + a_{2,2}x_2 + \dots + a_{2,m}x_m &= b_2; \\ &\dots \\ a_{n,1}x_1 + a_{n,2}x_2 + \dots + a_{n,m}x_m &= b_n, \end{aligned}$$

В матричном виде систему можно записать как

$$Ax = b,$$

где A – матрица размерности $n \times m$, b – вектор с m проекциями.

Для вычисления решения СЛАУ следует использовать функцию *lsolve*, обращение к которой имеет вид:

lsolve(A,b), где A – матрица системы; b – вектор правой части.

Решение систем нелинейных уравнений

MathCad дает возможность находить решение системы уравнений численными методами, при этом максимальное число уравнений в MathCad2001i доведено до 200.

Для решения системы уравнений необходимо выполнить следующие этапы.

Задание начального приближения для всех неизвестных, входящих в систему уравнений. При небольшом числе неизвестных этот этап можно выполнить графически, как показано в примере 10

Пример 10. Дана система уравнений:

$$y = x^2;$$

$$y = 8 + 3x.$$

Определить начальные приближения для решений этой системы.

Решение (рис. 4.9):

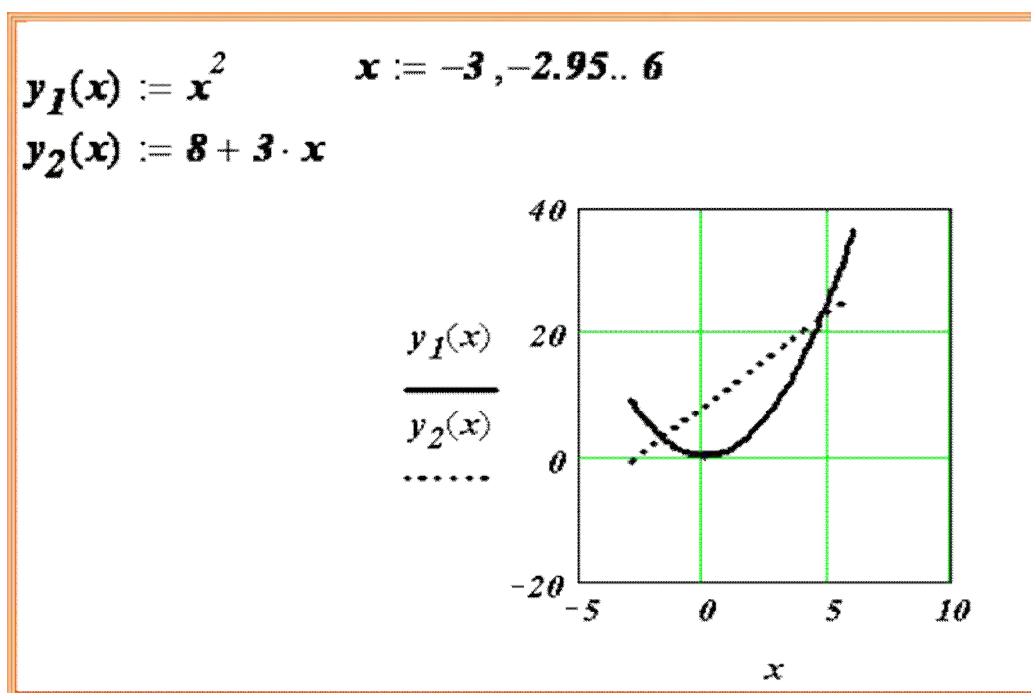


Рис. 4.9. Решение примера 10 в среде MathCad

Видно, что система имеет два решения: для первого решения в качестве начального приближения может быть принята точка $(-2, 2)$, а для второго решения – точка $(5, 20)$.

Вычисление решения системы уравнений с заданной точностью

Для вычисления решения системы уравнений с заданной точностью используется уже известный вычислительный блок **Given**.

Функция **Find** вычисляет решение системы уравнений с заданной точностью, и вызов этой функции имеет вид **Find(x)**, где x – список переменных, по которым ищется решение. Начальные значения этим переменным задаются в блоке < Начальные условия >. Число аргументов функции должно быть равно числу неизвестных.

Следующие выражения недопустимы внутри блока решения:

- ограничения со знаком ¹;
- дискретная переменная или выражения, содержащие дискретную переменную в любой форме;
- блоки решения уравнений не могут быть вложены друг в друга, каждый блок может иметь только одно ключевое слово **Given** и имя функции **Find** (или **Minerr**).

Пример 11. Используя блок **Given**, вычислить все решения системы предыдущего примера. Выполнить проверку найденных решений.

Решение (рис. 4.10):

$x := -2 \quad y := 2$ Начальное приближение для
первого решения

Given

$y = 8 + 3 \cdot x$

$y = x^2$

$s_A := \text{Find}(x, y)$

$s_A = \begin{pmatrix} -1.702 \\ 2.895 \end{pmatrix}$ Проекция первого решения

$x := 5 \quad y := 20$ Начальное приближение для
второго решения

Given

$y = 8 + 3 \cdot x$

$y = x^2$

$x > 0$ Ограничение на положительность
проекции x второго решения

$s_B := \text{Find}(x, y)$

$s_B = \begin{pmatrix} 4.702 \\ 22.105 \end{pmatrix}$ Проекция второго решения

Рис. 4.10. Решение примера 11 в среде MathCad

Пример 12. Используя функцию *Minerr*, вычислите решение системы уравнений

$$x + y = 0.95;$$

$$(x^2 + 1)^2 + (y^2 + 1)^2 = 5.5.$$

Решение (рис. 4.11):

$$x := 0 \quad y := 1$$

Given

$$(x^2 + 1)^2 + (y^2 + 1)^2 = 5.5$$

Жирный знак =

$$x + y = 0.95$$

Жирный знак =

$$z := \text{Minerr}(x, y)$$

$$z = \begin{pmatrix} -0.106 \\ 1.056 \end{pmatrix} \quad \text{Найденное решение}$$

$$z_0 + z_1 = 0.95$$

Проверка найденного решения

$$\lfloor (z_0)^2 + 1 \rfloor^2 + \lfloor (z_1)^2 + 1 \rfloor^2 = 5.5$$

Рис. 4.11. Решение примера 12 в среде MathCad

Библиографический список

1. <http://www.exponenta.ru> - образовательный математический сайт .
 2. <http://primat.org/> - сайт «Прикладная математика» .
 3. <http://ru.wikipedia.org/wiki/> -Википедия – свободная энциклопедия.
-
4. Бахвалов Н.С., Жидков Н.П., Кобельков Г.Г. Численные методы. - 8-е изд. – М.: Лаборатория Базовых Знаний, 2000. – 624 с.
 5. Турчак Л. И. Основы численных методов. – М.: Наука, 1987. – 320 с.
-
6. Дмитриева С.И., Федоров О.К., Смирнова Т.О. и др. Расчет схем паротурбинных и парогазовых установок и их элементов: учебное пособие /СПбГТУРП. -СПб., 2000. - 41 с.
 7. Пестриков В.М. Математическое моделирование теплотехнических задач в программируемых средах: учебное пособие / СПбГТУРП. - СПб., 2009. – 104 с.
 8. <http://pers.narod.ru> Софт с исходниками, обучение, алгоритмы, PHP, JavaScript, тексты, графика
 9. Демидович Б.П., Марон И.А., Шувалова Э.З. Численные методы анализа. Приближение функций, дифференциальные и интегральные уравнения. - М.: Наука. 1967. - 368 с.
 10. Вержбицкий В.М. Численные методы. Математический анализ и обыкновенные дифференциальные уравнения. - М.: Высшая школа, 2001. – 384 с.
 11. <http://window.edu.ru/> Единое окно доступа к образовательным ресурсам .
-

ОГЛАВЛЕНИЕ

1. ОСНОВНЫЕ ПОНЯТИЯ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ	3
1.1. Компьютерное моделирование как метод научного познания.....	3
1.2. Классификация моделей	6
1.3. Технология компьютерного моделирования.....	8
1.4. Этапы компьютерного моделирования.....	11
2. ПРИБЛИЖЕНИЕ ФУНКЦИЙ.....	13
2.1. Введение в теорию приближений.....	13
2.2. Метод наименьших квадратов	14
2.2.1. Постановка задачи.....	14
2.2.2. Линейная регрессия.....	17
2.2.3. Подбор эмпирических формул.....	18
2.2.4. Степенная зависимость (геометрическая регрессия).....	19
2.2.5. Показательная зависимость.....	21
2.2.6. Пример поиска приближающей функции методом наименьших квадратов.....	24
2.2.7. Пример решения задачи курсового проекта в среде MathCad	27
3. КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ЗАДАЧИ ДИРИХЛЕ.....	33
3.1. Численная постановка задачи.....	33
3.2. Метод сеток	37
3.3. Решение задачи курсового проекта в среде MathCad.....	41
ПРИЛОЖЕНИЯ.....	48
Приложение 1. Компьютерная математика и математические пакеты.....	48
Приложение 2. Массивы в MathCad.....	50
Приложение 3. Решение нелинейных уравнений и систем уравнений в пакете MathCad.....	54
Приложение 4. Решение систем уравнений.....	61
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	67