

• РАДИО И СВЯЗЬ •

# СПРАВОЧНИК

---

В.П. ДЬЯКОНОВ

СИСТЕМА  
MATHCAD



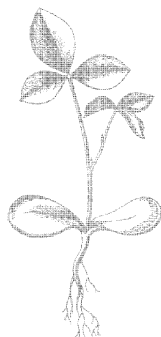
# СПРАВОЧНИК

В.П. ДЬЯКОНОВ

## СИСТЕМА MATHCAD



МОСКВА „РАДИО И СВЯЗЬ”  
1993



Scan AAW

ББК 32.973  
Д 93  
УДК 681.3.06

**Федеральная целевая программа книгоиздания России**

**Редакция литературы по информатике и вычислительной технике**

**Дьяконов В. П.**

Д 93 Система MathCAD: Справочник.— М.: Радио и связь,  
1993 — 128 с.: ил.

ISBN 5-256-01094-8.

Описана интегрированная система автоматизации математических и инженерно-технических расчетов MathCAD, ориентированная на IBM-совместимые персональные компьютеры. Система содержит текстовый редактор, мощный вычислитель и удобные средства математической графики. Значительное внимание уделено возможностям системы (матричные и векторные операции, действия с комплексными числами, реализация быстрого преобразования Фурье и др.) в практике типовых расчетов, включая моделирование электрических и электронных цепей. Большинство примеров иллюстрировано графикой.

Для широкого круга читателей — студентов, аспирантов, инженеров и научных работников.

Д 2404020000-056 31-93  
046(01)-93

ББК 32.973

**Справочное издание**

**Дьяконов Владимир Павлович**

**СИСТЕМА MathCAD**

**Справочник**

Заведующий редакцией Ю. Г. Ивашов  
Редактор Н. Г. Давыдова  
Обложка художника Н. А. Пашуро  
Художественный редактор В. И. Мусиенко  
Технический редактор А. Н. Золотарева  
Корректор Т. С. Власкина

**ИБ № 2543**

Сдано в набор 12.11.92. Подписано в печать 6.05.93. Формат 60×88<sup>1</sup>/<sub>16</sub>. Бумага тип. № 2. Гарнитура литературная. Печать офсетная. Усл. печ. л. 7,84. Усл. кр.-отт. 8,09. Уч.-изд. л. 7,50. Тираж 10 000 экз. Изд. № 23700. Зак. № 1988. С-056.

Издательство «Радио и связь». 101000 Москва, Почтамт, а/я 693  
Московская типография № 4 Министерства печати и информации РФ.  
129041, Москва, Б. Переяславская, 46

ISBN 5-256-01094-8

© Дьяконов В. П., 1993

## Предисловие

Для автоматизации математических расчетов используются разнообразные вычислительные средства — от программируемых микрокалькуляторов и персональных ЭВМ [1—4] до сверхмощных суперЭВМ. И тем не менее такие расчеты остаются сложным делом. Более того, применение компьютеров внесло новые трудности: прежде чем начинать расчеты, пользователь должен был освоить основы программирования, изучить один, а то и несколько языков программирования и достаточно специфические численные методы расчетов [5—8].

Положение стало меняться к лучшему после разработки специализированных программных комплексов для автоматизации математических и инженерно-технических расчетов. К таким комплексам относятся пакеты программ MathCAD, MatLAB, Eureka и др. Данная книга посвящена описанию одной из наиболее мощных и универсальных интегрированных систем — системы MathCAD [9], ориентированной на IBM-совместимые персональные компьютеры.

Система MathCAD содержит текстовый редактор, мощный вычислитель и очень простой в применении графический процессор. Это позволяет готовить документы, по виду напоминающие статьи или разделы из книг. Язык общения с пользователем в системе MathCAD предельно приближен к обычному математическому языку. Вычислитель системы содержит множество математических функций — от всем известных элементарных до столь специфических, как быстрое преобразование Фурье или сплайн-интерполяция.

Система MathCAD универсальна. Ее с одинаковым успехом можно использовать для выполнения как школьных, так и сложных расчетов, вполне достойных включения в серьезные труды по математике. При этом центр тяжести расчетов перемещается с вопросов программирования на вопросы естественного математического описания алгоритмов решения математических и инженерно-технических задач.

Обширный набор векторных и матричных операций и выполнение практически всех операций как с действительными, так и с комплексными числами делают систему MathCAD очень удобной для физических и электротехнических расчетов, анализа и моделирования электрических и электронных цепей и устройств.

Система MathCAD довольно распространена и у нас в стране. Но большинство пользователей не располагает ее фирменным техническим описанием, без которого практическое применение системы просто невозможно.

Эта книга является кратким, но полным справочником по системе MathCAD. В то же время книга оригинальна и не является переводом крайне дефицитного фирменного описания [9]. По рубрикации, набору примеров и направленности на расчеты электрических и электронных цепей она существенно отличается от описания [9]. Скорее, она на принципиально новой основе продолжает цикл справочников автора по математическим и инженерно-техническим расчетам [1—4].

Книга рассчитана на широкий круг читателей — всех, кто в своей работе встречается с необходимостью вести математические расчеты различной сложности. В рекламе, приведенной в конце книги, указан адрес для переписки с автором и отмечается возможность приобретения дискет со всеми документами, описанными в книге, что заметно облегчает освоение системы.

# 1. ОСНОВЫ РАБОТЫ В СИСТЕМЕ MathCAD

## 1.1. Назначение системы

MathCAD является интегрированной системой программирования, ориентированной на проведение математических и инженерно-технических расчетов. MathCAD содержит текстовый редактор, вычислитель и графический процессор.

*Текстовый редактор* служит для ввода и редактирования текстов. Тексты являются комментариями, и входящие в них математические выражения не исполняются. Текст может состоять из слов, математических выражений и формул, спецзнаков. Отличительная черта системы MathCAD — использование общепринятой в математике символики. Например, знак деления обозначается горизонтальной чертой, а не наклонной. Таких малопонятных функций, как  $SQR(X)$  или  $SQRT(X)$ , применяемых в языках программирования Бейсик или Паскаль, в системе MathCAD нет.

*Вычислитель* обладает поистине уникальными возможностями. Он обеспечивает вычисления по сложным математическим формулам, имеет большой набор встроенных математических функций, позволяет вычислять ряды, суммы и произведения, определенные интегралы и производные, работать с комплексными числами, а также решать линейные и нелинейные уравнения, проводить минимизацию функций, выполнять векторные и матричные операции. В вычислитель входят и такие мощные средства, как линейная и сплайн-интерполяция, регрессия, прямое и обратное быстрое преобразование Фурье, статистические расчеты. Легко можно менять разрядность чисел и погрешность итерационных методов.

*Графический процессор* служит для создания графиков. Он, несомненно, порадует всех, кто любит графическое представление результатов математических расчетов. Графический процессор сочетает чрезвычайную простоту общения с пользователем с самыми изысканными возможностями графических средств. Простые графики нескольких функций пользователь может начать строить буквально в первые секунды знакомства с системой. По мере приобретения навыков работы с графическим процессором легко осваиваются и другие графические средства — графики в логарифмическом масштабе, масштабные сетки с любым числом делений, линии, отмеченные точками, прямоугольниками и ромбиками, гистограммы и др. Графика ориентирована на решение типичных математических задач. Возможно быстрое изменение размеров графиков, наложение их на текстовые надписи и перемещение в любое место документа.

Система MathCAD ориентирована на IBM-совместимые персональные компьютеры, например IBM PC, XT, AT, PS/2 и им подобные. Она используется совместно с операционными системами PC-DOS или MS-DOS с версиями от 2.0 и выше. Поддерживаются основные типы видеоадаптеров и дисплеев — CGA, EGA, Hercules и др. Минимальный объем памяти оперативного запоминающего устройства 512 Кбайт. Систему можно разместить на двух дискетах с емкостью 360 Кбайт или на одной дискете с емкостью 720 Кбайт или выше.

MathCAD автоматически поддерживает работу с математическим сопроцессором. Последний заметно повышает скорость расчетов и вывода графиков, что

существенно в связи с тем, что MathCAD всегда работает в графическом режиме. Последнее связано с тем, что только в этом режиме можно формировать на экране дисплея специальные математические символы (например, квадратный корень любой длины или знак интеграла) и одновременно применять их вместе с графиками и текстом.

MathCAD поддерживает работу с многими типами принтеров — от стандартных класса Epson до самых современных лазерных принтеров с исключительно высоким качеством печати. Поддерживается и работа с плоттерами, способными создавать графики с плавными линиями, а не ступенчатыми или точечными, как у обычных матричных принтеров.

MathCAD — система универсальная, т. е. она может использоваться в любой области науки или техники, везде, где применяются математические методы. Запись команд в системе MathCAD на языке, очень близком к стандартному языку математических расчетов, резко упрощает постановку и решение задач. Перефразируя слова создателя языка Лого Пайперта, можно сказать, что, работая в системе MathCAD, пользователь обучает свой компьютер, как решать нужные ему задачи, а не компьютер обучает пользователя основам программирования.

Тем самым главные аспекты решения математических задач смещаются с их программирования на алгоритмическое и математическое описание. В сочетании с обширными встроенными возможностями вычислений и прекрасными графическими средствами это делает MathCAD гибким и удобным инструментом в руках математика и физика, ученого и инженера, студента и школьника.

## 1.2. Первые шаги

Предположим, что файлы системы MathCAD скопированы на жесткий или гибкий диск. Вначале нужно настроить компьютер на работу с соответствующим накопителем. Допустим, это накопитель на гибком диске с именем A. Для работы компьютера с ним введем команду

A:⟨ENTER⟩

где ⟨ENTER⟩ означает нажатие клавиши ввода. Она может иметь и другое обозначение — его нужно уточнить. Далее вводится команда

MCAD⟨ENTER⟩

и происходит загрузка системы в память компьютера. Если на компьютере установлена специальная система для работы с файлами, например Norton Commander, то нужно установить метку файла на имя MCAD.EXE и нажать клавишу ENTER. Спустя некоторое время на экране дисплея появится окно системы с ее именем. Нажатие клавиши F10 вызовет появление главного меню (рис. 1.1). С его помощью можно задать выполнение большого числа различных команд управления систе-

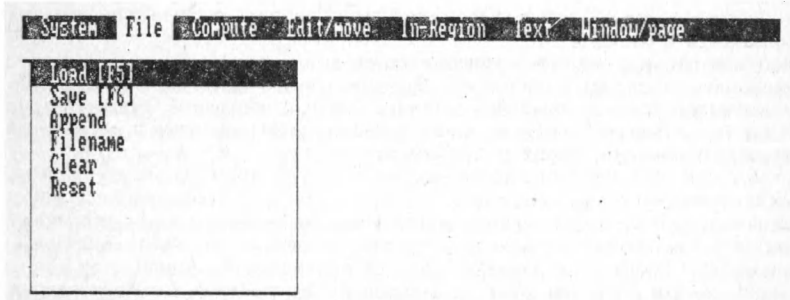


Рис. 1.1. Главное меню системы MathCAD

Математика

ПРИМЕРЫ ЭЛЕМЕНТАРНЫХ ВЫЧИСЛЕНИЙ

$$12 + \frac{48}{2} = 36 \quad (12 + 48) \cdot 2 = 120 \quad -8 + \frac{24}{6} = -4$$

$$\sin(1) = 0.841 \quad \cos(1) = 0.54 \quad \tan(1) = 1.557$$

$$\sinh(1) = 1.175 \quad \cosh(1) = 1.543 \quad \tanh(1) = 0.762$$

$$\log(1000) = 3 \quad \ln(2) = 0.693 \quad e = 7.389$$

$$\sqrt{2} = 1.414 \quad \sin\left[\frac{\sqrt{2}}{2}\right] = 0.65 \quad 2 \cdot \pi = 6.283$$

$$\text{angle}(1,1) \cdot \frac{180}{\pi} = 45 \quad \text{asin}(0.841) = 0.999$$

$$\text{acos}(0) = 1.571 \quad \text{atanh}(0.762) = 1.001$$

Рис. 1.2. Примеры вычислений в системе MathCAD

мой. При этом система практически готова к работе. Нажмите клавишу Esc — меню исчезнет, и можно приступать к оперативной работе.

Простейшие вычисления можно выполнить, используя знак вывода результатов вычислений (= (равенство)), что соответствует схеме

Выражение =

На рис. 1.2 представлен ряд примеров простых вычислений. Из них видно, что в левой части равенства могут стоять любые математические выражения, содержащие встроенные в систему функции.

MathCAD реализует вычисления в строго определенном порядке, как это делает человек, читая страницу книги, т. е. слева направо и сверху вниз. Чтобы лучше понять специфику работы системы, нужно познакомиться с некоторыми понятиями.

**Документом** в системе MathCAD называется полное математическое описание алгоритмов решения задач. Документ, в свою очередь, состоит из **блоков**, т. е. отдельных частей. Блоки могут быть трех типов — текстовые, вычислительные и графические. Каждый блок занимает на экране некоторое пространство, ограниченное прямоугольной областью. Указанный выше порядок выполнения вычислений относится и к блокам.

Текстовые блоки играют роль неисполняемых комментариев. Они служат лишь для повышения наглядности документа. Вычислительные блоки состоят из исполняемых математических выражений, например формул, равенств, равенств и неравенств и т. д. Графические блоки также являются исполняемыми и служат для вывода результатов вычислений в графическом виде.

Правильный порядок выполнения блоков — основа правильного функционирования системы. Например, если в некотором блоке содержатся операции, требующие данных из другого блока, то этот другой блок обязательно должен выполняться первым и располагаться перед использующим его блоком. Иная ситуация приведет к появлению ошибки. Ее характер зависит от решаемой задачи.

**Сигнал ошибки** в системе имеет вид надписи, заключенной в прямоугольник. От него отходит черта, указывающая место ошибки. Таким образом, место ошибки легко найти.

Размеры блоков устанавливаются автоматически в зависимости от числа входящих в них знаков и математических выражений либо от заданных размеров графиков. Обычно границы блоков не видны. Однако по виду курсора системы можно судить о переходе от одного блока к другому. В текстовых блоках и в промежутках между блоками курсор имеет вид мигающей горизонтальной черточки. Когда он входит в текстовый блок, последний обрамляется кавычками (вообще, для задания текстового блока следует ввести открывающие его кавычки, закрывающие кавычки появляются автоматически и отслеживают за концом последней строки текста). Когда курсор входит в формульный или графический блок, его форма напоминает развернутую букву L. Если блок графический, то в нем имеется *шаблон* блока, т. е. прямоугольник, в котором затем строится график. Указанные признаки позволяют легко определять границы и тип блоков.

Блоки не должны налагаться друг на друга. Исключением может быть наложение графического блока на текстовый блок с тем, чтобы получить на графике надписи. В остальных случаях система откажется от ввода блока в неподходящее место. В верхней строке при этом появляется надпись

Can't edit blank space  
(Нет места для редактирования блока).

Система MathCAD имеет ряд *режимов работы*. При первом включении устанавливается режим *auto* (автоматический). Такой режим позволяет выполнять вычисления сразу по мере ввода и редактирования документа. Однако это создает специфическую «медлительность» системы, поскольку на вычисления, нередко довольно сложные, система вынуждена затрачивать время. Медлительности способствует и работа системы в графическом режиме: стоит изменить хоть одну точку изображения, как придется повторить построение изображения на всем экране. Естественно, чем выше производительность персонального компьютера, тем меньше сказывается эта медлительность.

Если загрузить систему командой

MCAD /M

то будет установлен другой режим — *manual*. В этом режиме ввод и редактирование документа происходят без выполнения вычислений. Реакция системы на действия пользователя становится более быстрой, и редактирование оказывается более удобным. Для перехода к режиму вычислений при этом достаточно нажать функциональную клавишу F9. Вычисления охватят те блоки, которые расположены сверху от текущего положения курсора.

Имеется и ряд других команд управления системой. С ними можно ознакомиться, нажав клавишу F10. В верхней строке появится *основное меню* системы. Из него «выпадают» *подменю*, содержащие команды управления системой. Эти команды мы рассмотрим более подробно по мере дальнейшего знакомства с системой.

### 1.3. Ввод текстов

Для ввода текстов, т. е. создания текстовых блоков документов, достаточно ввести знак « (кавычки). Текущее положение курсора окажется в обрамлении из пары кавычек, т. е. знака «». Между этими кавычками можно вводить текст. По мере ввода текста расстояние между кавычками автоматически увеличивается. Если надо перейти на новую строку текста, то следует нажать клавишу ввода ENTER. При этом текстовый блок расширяется на одну строку.

Закрывающие кавычки отслеживают за максимальной длиной строки. Таким образом, кавычки всегда указывают на верхний левый и нижний правый углы четырехугольника, обрамляющие текстовый блок. Внутри текстового блока можно пользоваться стандартными приемами редактирования текста — перемещением курсора вверх и вниз, вправо и влево, уничтожением и вставкой символов. Более подробно приемы редактирования будут описаны в дальнейшем. Пробелы в текст вводятся нажатием клавиши пробела. Как отмечалось, математические выраже-



ния в текстовом блоке играют роль комментариев и не выполняются. Например, надпись «Вычисляется  $\sin(1) \Rightarrow$ » не приведет к вычислению значения  $\sin(1)$ . Просто эта надпись будет выведена на экран дисплея.

Текстовый блок можно покинуть, выведя за его пределы курсор. Для этого используются клавиши перемещения курсора. Введенный текст остается на экране дисплея, но обрамляющие его кавычки исчезают.

Если в память компьютера загружен драйвер кириллицы, т. е. букв русского алфавита, то в текстовых блоках можно пользоваться этими буквами. Переключение алфавитов с латинского на русский и обратно соответствует указанному в примененном драйвере. Однако в ряде случаев знаки кириллицы занимают в памяти место, отведенное под греческие буквы. В этом случае часть из них не может использоваться. Вообще, для вывода ряда греческих букв необходимо одновременно нажать клавишу Alt и одну из буквенных клавиш со сходным звучанием буквы, например Alt и A для буквы  $\alpha$ .

Мы вернемся к более детальному знакомству с приемами редактирования текстов в системе MathCAD. Пока лишь отметим, что они включают в себя режимы ввода больших фрагментов текста, центрирования и выравнивания правого края строк, вставки и удаления строк, работы с блоками и другие приемы, характерные для текстовых редакторов. Впрочем, надо отметить, что команды редактирования текстов отличаются от таких команд у наиболее распространенных текстовых редакторов (например, WD, WORKS, Multi-EDIT, Лексикон и др.). Последнее нельзя считать серьезным недостатком, так как текстовый режим для системы MathCAD не является основным. Гораздо важнее, что документ в системе MathCAD может сочетать тексты с математическими формулами в их естественном виде и графиками.

## 1.4. Азы вычислений

Продолжим элементарное знакомство с возможностями системы MathCAD и рассмотрим работу с вычислительными блоками. Ранее мы убедились, что любое выражение вычисляется, стоит после него поставить знак равенства  $=$ . Данный знак — один из множества специальных знаков, включенных в систему.

Чтобы придать вычислениям общность, в математике обычно оперируют как числовыми, так и символьными данными. К ним относятся *переменные* — имеющие имена хранилища числовых данных. Полезно понимать, что этими хранилищами являются определенные участки памяти компьютера. А их имена по существу указывают положение (адрес) соответствующего участка в общем адресном пространстве компьютера.

Обычно присваивание переменной определенного значения отождествляют с ее равенством этому значению. Например, если  $X=5$ , то говорят, что значение переменной  $X$  равно пяти. Но в системе MathCAD знак равенства уже использован по другому назначению — для указания о выводе значения переменной или арифметического выражения на экран дисплея.

Присваивание в системе MathCAD реализуется с помощью знака  $:=$  (двоеточие с равенством). Практически для этого достаточно ввести знак двоеточия. Итак, если ввести

$X:=5$

то на экране дисплея появится

$X:=5$  (т. е. переменной  $X$  присваивается значение 5).

*Константой* в системе MathCAD называют неизменные числовые значения. Например, константа 5 имеет значение, равное пяти, в любом месте программы.

Константа в виде *системной переменной* — это предварительно определенная переменная, значение которой задается в начале загрузки системы. Например, такой константой является число  $e=2.718...$ , которое служит основанием натуральных логарифмов. Другой пример константы — число  $\pi=3.141...$

Было бы логично предположить, что и такие константы не должны менять свое назначение по ходу выполнения вычислений. Но как быть с ситуацией, когда буквой  $e$  обозначают заряд электрона или электродвижущую силу? Видимо, это и привело разработчиков системы к тому, что ряд констант имеет функции системных пере-

менных. С одной стороны, их значения предварительно определены как типовые константы, но, с другой стороны, чересчур настойчивый пользователь может использовать их как переменные, меняющие свои значения по ходу исполнения программы.

Проведите такой эксперимент. После загрузки системы введите  $e=$ . Тут же получите ожидаемое  $e=2.718$ . Далее введите  $e:=10$ , а затем опять  $e=$ . Получите  $e=10$ . Разумеется, теперь число  $e$  уже нельзя использовать как основание натуральных логарифмов.

Важным понятием в математике является *функция*, т. е. некоторая зависимость одной переменной от другой или ряда переменных, или констант. В системе MathCAD имеется множество *встроенных функций*, т. е. функций, заблаговременно введенных в нее разработчиками. Здесь и тривиальные алгебраические и тригонометрические функции, более редкие гиперболические функции, специальные математические и статистические функции и даже известные далеко не каждому специалисту функции прямого и обратного быстрого преобразования Фурье или сплайн-интерполяции.

Главным признаком функции является *возврат значения*, т. е. функция в ответ на обращение к ней по имени с указанием ее аргументов должна вернуть свое значение. Например, если где-то используется функция  $\sin(X)$ , то вместо нее будет подставлено число, равное значению синуса с аргументом, равным значению переменной  $X$ .

Пример:

$X:=1$

$\sin(X)=0.841$

$2+\sin(X)=2.841$

Переменной  $X$  (аргументу) присвоено значение 1.

Функция возвратила значение  $\sin(1)=0.841$ .

Вычислено выражение  $2+\sin(X)=2.841$ .

Еще одной важной особенностью системы является возможность задания *внешних функций*, или *функций пользователя*. Фактически это означает возможность расширения системы. Хотя система содержит множество встроенных функций, где гарантия того, что кому-нибудь не понадобится какая-либо новая функция, например такая:

$es(X)=\exp(\sin(X))$  или  $ses(X, Y):=\exp(X)+\sin(Y)$ .

В этих примерах видно, что функция пользователя задается своим именем, списком аргументов в круглых скобках и арифметическим выражением после знака присваивания. Заданная таким образом функция ведет себя как встроенная функция, т. е. в ответ на обращение к ней возвращает свое значение.

Для идентификации, т. е. распознавания переменных, констант и функций, служат их имена — *идентификаторы*. Они должны начинаться с латинской буквы и могут содержать в себе цифры. Идентификаторы должны быть уникальны, т. е. не повторять имен ранее введенных констант, переменных и функций в том числе встроенных.

Другим важным определением системы являются *операторы* — специальные знаки или слова, вызывающие определенные действия. Например, операторами являются знаки сложения  $+$  и вычитания  $-$ , вывода  $=$ , присваивания  $:=$  и многие другие. К операторам относятся и специальные знаки: квадратного корня, интеграла, вычисления производной и т. д. Поэтому язык программирования системы MathCAD оказывается очень близким к обычному математическому языку.

Необходимо отметить еще одну особенность — задание переменных с пределами изменения, что фактически определяет возможность проведения *циклических вычислений*. Целочисленная переменная, меняющаяся с шагом 1 от значения Start до значения End, задается следующим образом:

Name: Start; End что дает Name:=Start..End.

Итак, начальное значение переменной отделяется от конечного знаком «точка с запятой», но на экране дисплея он трансформируется в две точки.. на одной строке. Например, если задать  $1:=0.5$ , то переменная 1 будет принимать значения от 0 до 5 с шагом 1, т. е. 0, 1, 2, 3, 4 и 5. Шаг может быть равен и  $-1$ , например, если задать  $X:=5.0$ . Тогда  $X$  будет меняться от 5 до 0, т. е. принимать значения 5, 4, 3, 2, 1 и 0. Далее будет показана возможность задания любого, но обязательно целочисленного шага.

## 1.5. Построение графиков

Те, кому пришлось составлять программы графического вывода результатов, знают, сколь непросто запрограммировать построение графика заданной функции на фоне координатных осей или масштабной сетки, особенно если необходимо автоматическое изменение масштаба графиков или придание им особых свойств, например выделение отдельных кривых, представление графиков в логарифмическом масштабе и др. Даже профессионально сделанные мощные интегрированные системы, такие как SuperCalc или Lotus-123, страдают примитивностью графиков, особенно относящихся к представлению результатов математических, физических или инженерно-технических приложений.

Тем более приятно, что система MathCAD не только избавляет пользователя от утомительных процедур программирования графических операций, но и дает ему возможность крайне просто и изящно выводить результаты вычислений в наглядной и изысканной графической форме. У многих коллег автора восторг вызывали не столько мощные возможности графики, сколько поразительная простота их реализации. Трудно себе представить, сколько времени и сил могли бы сэкономить авторы книг по математическим кривым, например [10], располагая они системой MathCAD.

Графики в системе MathCAD могут размещаться в любом логически дозволенном месте документа, иметь любой размер и представлять результаты многих расчетов. Под логически дозволенным местом подразумевается то, что графики могут помещаться после тех вычислительных блоков системы, которые готовят исходные данные для построения графиков. Это принципиальное отличие от графиков в книгах или журналах вполне очевидно любому пользователю — даже с умеренными математическими способностями.

Рассмотрим несколько типовых примеров реализации графических возможностей. Допустим, нужно построить график параболы при аргументе  $x$ , меняющемся от 0 до 5. Рисунок 1.3 иллюстрирует создание документа, обеспечивающего построение такого графика.

В левой части документа задана переменная  $X$  с пределами изменения от 0 до 5 и функция — парабола. Выводятся табличные значения функции. Здесь важно отметить, что все вычисления после задания переменной и ее пределов изменения повторяются столько раз, сколько раз меняется переменная. Таким образом, значения как переменной, так и функции образуют одномерные массивы — векторы. Каждый элемент вектора заданной функции является индексированной переменной. Для ввода подстрочного индекса необходимо после набора буквы  $X$  указать знак [ (открывающая квадратная скобка). Для набора степени — надстрочного индекса следует ввести обычный знак  $\wedge$  возведения числа в степень и затем набрать показатель степени.

Теперь приступим к созданию графика. Для этого надо вывести курсор из вычислительных блоков и установить его в место, которое будет левым верхним углом

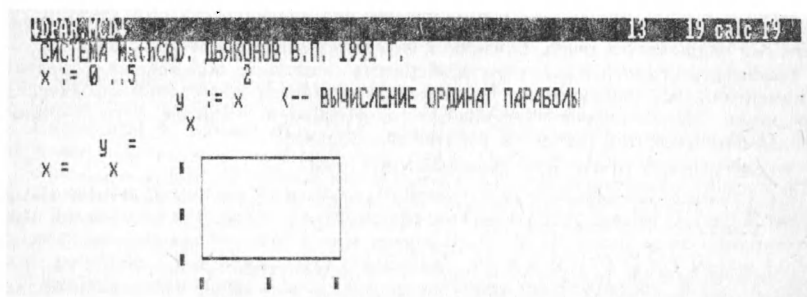


Рис. 1.3. Начало построения графика параболы

прямоугольного блока графики. Затем введем символ @, обеспечивающий включение графического процессора. На экране появится шаблон будущего графика в виде прямоугольной рамки с маленькими прямоугольниками, расположенными вдоль осей X и Y будущего графика (рис. 1.3). Крайние прямоугольники задают масштабы графиков, а средние служат для указания переменных, изменения которых отображаются графически. Проставив имена этих переменных, введем курсор в пределы шаблона и нажмем клавишу ENTER. Будет построен график, показанный на рис. 1.4.

В данном случае мы не вводили масштабы графиков по осям X и Y. Система сделала это автоматически, отметив (на экране дисплея) масштабы маленькими уголками. Можно, разумеется, установить масштабы и вручную — введя нужные значения на место шаблонов. В этом случае уголки будут отсутствовать.

Допустим, нас не удовлетворяет вид графика — скажем, его слишком малые размеры. Для изменения вида графика имеется ряд команд. Ограничимся простейшим их применением. Введем курсор в поле графика и нажмем клавишу F. Поскольку курсор был в поле графика, это введет команду plot format (формат графиков). В верхней строке появится сообщение, указывающее на основные параметры графика:

logs=0,0 subdivs=1,1 size=5,15 type=1

Мы вернемся к подробному обсуждению параметров графики, а пока лишь отметим, что параметр subdivs указывает на число делений масштабной сетки по вертикали и горизонтали, тогда как параметр size — на размер графика в знаках-местах. Персональные компьютеры класса IBM PC имеют размер экрана в 25 строк по 80 символов в каждой. В то же время система MathCAD способна готовить документы и большего размера, хотя видимой будет часть документа (25 строк по 80 символов).

Итак, если нужно увеличить размеры графика вдвое, то исходные значения параметра size=5,15 надо изменить на новые size=10,30. Это делается с помощью обычного редактирования верхней строки с применением клавиш передвижения курсора, удаления знаков и набора цифр. Теперь, нажав клавишу ENTER, можно получить график удвоенного размера — см. правый график на рис. 1.4.

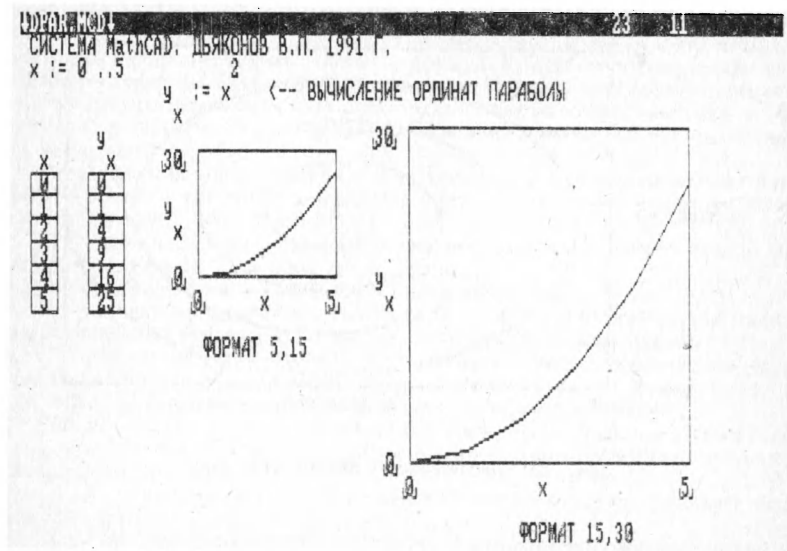
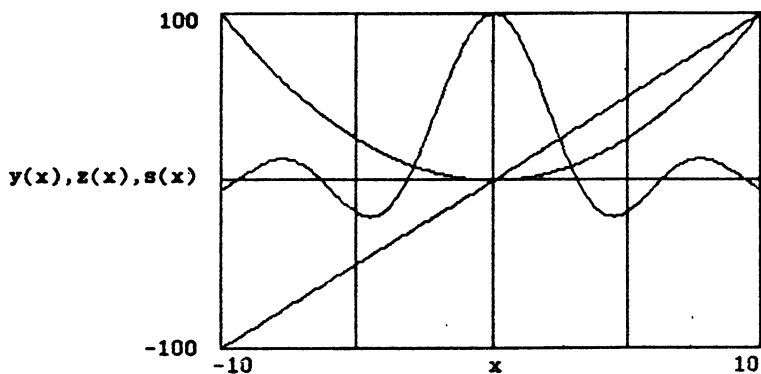


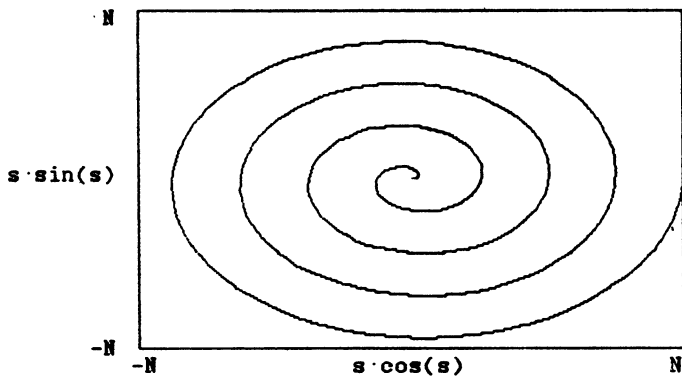
Рис. 1.4. Конец построения графика параболы

$y(x) := 10 \cdot x$                       функция 1- прямая линия  
 $z(x) := x^2$                       функция 2- парабола  
 $s(x) := 100 \cdot \frac{\sin(x)}{x}$                       функция 3- имеет особенность при  $x \rightarrow 0$  в виде неопределенности  
 $x := -10, -9.9 \dots 10$                        $0/0 \rightarrow 1$



Док. 1.1. Построение графиков трех функций

$N := 8 \cdot 3.14$                        $s := 0, .1 \dots N$



Док. 1.2. Построение графиков спирали

Надо ли это делать каждый раз, когда строится график? Разумеется, нет — так надо действовать лишь в процессе отладки решения Вашей задачи. Система MathCAD запоминает все изменения параметров каждого графика, так что в последующем он будет строиться в виде и формате, полученных после окончательной отладки.

Столь же легко можно построить графики нескольких функций — для этого достаточно определить их и перечислить в виде списка в шаблоне графика. Документ 1.1 иллюстрирует построение графиков трех функций в пределах одного графического блока.

Возможно и построение графиков в полярной системе координат. Примером может служить построение графика спирали, которое иллюстрирует док. 1.2. Математически спираль описывается уравнениями

$$x(s) = s \cdot \cos(s),$$

$$y(s) = s \cdot \sin(s),$$

где параметр  $s$  плавно увеличивается, начиная с нуля.

## 1.6. Редактирование, запись и считывание документов

Редактирование документа — это изменение его вида и параметров. Предполагается, что редактируемый документ каким-либо образом введен, например с клавиатуры или путем его считывания с диска. С некоторыми приемами редактирования мы уже столкнулись выше — это снабжение документа текстовыми комментариями, правила набора идентификаторов и некоторых операторов, изменение параметров графиков и т. д.

Система MathCAD имеет мощный редактор документов. Точнее, их даже три — это редакторы текстовых, вычислительных и графических блоков. Полному рассмотрению приемов редактирования посвящена отдельная глава данной книги. Здесь же мы опишем лишь главные приемы редактирования, впрочем, порой вполне достаточные для практической работы с системой пользователей, для которых математика — все же не самое главное в их жизни.

Общие приемы редактирования с помощью курсора и операций вставки и удаления символов настолько известны и интуитивно предсказуемы, что на них даже не стоит останавливаться. Пара минут эксперимента, и Вы освоите премудрости простейшего редактирования текстов и формул. Кратко эти премудрости таковы: установите курсор на неверный знак и нажмите клавишу стирания Del, после этого увидите исправленный знак. Есть еще одна клавиша стирания — над клавишей ENTER. С ее помощью можно удалить любой знак слева от курсора или даже подряд несколько знаков.

Редактирование резко ускоряется и упрощается при использовании *блочных команд*. Отметим три такие важнейшие команды, вводимые нажатием функциональных клавиш:

F2 — помещает копию блока в специальный буфер, т. е. место в памяти, хранящее данные о блоке (сам блок сохраняется);

F3 — стирает блок и помещает в буфер его копию;

F4 — помещает копию блока из буфера на место документа, указанное текущим положением курсора, если это место не занято другим блоком.

С помощью этих трех команд можно творить чудеса с документами, стирая и перемещая блоки и тем самым добиваясь нужного вида документа. Следует, однако, помнить о правиле логической последовательности блоков.

Еще две полезные и часто встречаемые операции относятся к пустым строкам, т. е. к межстрочным промежуткам. Они вводятся одновременным нажатием двух указанных ниже клавиш:

Ctrl F9 — вводит пустую строку между блоками в месте, указанном положением курсора,

Ctrl F10 — уничтожает пустую строку, в которой расположен курсор.

После того как документ отредактирован и проверен в работе, его обычно требуется сохранить. Документы хранятся на магнитных дисках в виде *файлов*,

т. е. специально закодированных данных, несущих информацию о виде документа и его параметрах, задающих алгоритм вычислений. Файлы документов имеют расширение .MCD. Их можно просмотреть с помощью текстового редактора и убедиться, что кодирование выполняется весьма наглядными средствами — большая часть файла почти повторяет текстовое описание документа с включением достаточно понятных служебных команд. После определенных навыков работы с системой MathCAD несложно научиться выправлять файлы системы с помощью текстового редактора, что иногда полезно.

Простейший способ записи документа — ввод команды Save путем нажатия функциональной клавиши F6. В верхней строке появится запрос имени файла Save as (записать как). Введите желательное имя документа и нажмите клавишу ENTER. Если имя документа оригинально, то файл с данными документа будет тут же записан на диск. Однако, если файл с таким именем уже есть на диске, в верхней строке появится запрос

File exist—OK to overwrite?

(Файл уже есть, писать поверх него?)

Ответом должны быть буквы Y (от yes — да) или N (от no — нет). Ответ Y приведет к замене существующего файла новым, а N — к отказу от записи. В последнем случае запись можно повторить.

## 2. УПРАВЛЕНИЕ СИСТЕМОЙ MathCAD

### 2.1. Основное меню

Управление системой MathCAD осуществляется рядом команд. Для удобства работы с системой они вводятся с помощью меню команд. Последнее имеет древовидную структуру и содержит основное меню, из которого выпадают подменю.

Для ввода основного меню нужно нажать клавишу F10. Основное меню появляется в верхней строке экрана — *командной строке*. Одна из позиций основного меню выделена цветом и фоном, т. е. активна (см. рис. 1.1, на котором показано переднее панно системы с активной позицией File основного меню и командой Load подменю). Используя клавиши перемещения курсора влево и вправо, можно выбрать нужную позицию основного меню. При нажатии клавиши ENTER наблюдается выпадение из основного меню подменю. Нужная позиция в подменю выбирается с помощью клавиш перемещения курсора вверх и вниз.

Основное меню содержит следующие позиции:

System (система) — группа команд по управлению системными возможностями системы;

File (файл) — группа команд по управлению файлами;

Compute (вычисление) — группа команд по управлению вычислительными процессами;

Edit/move (редактирование/перемещение) — группа команд редактирования документов и перемещения блоков с одного места на другое;

In-Region (входная область) — группа команд по редактированию отмеченных фрагментов документов;

Text (текст) — группа команд редактирования и ввода текстов;

Window/page (окно/страница) — группа команд задания окон и управления форматом страниц.

Существует более быстрый и менее монотонный способ задания требуемой команды, чем ее поиск в меню с помощью курсора. Для этого после нажатия клавиши F10 следует нажать последовательно на две клавиши — одну с первой буквой имени команды основного меню, а другую с первой буквой имени команды подменю. Например, если нужно ввести команду Save из подменю, относящегося к позиции File основного меню, то следует нажать клавиши F10 F S ENTER

При нажатии клавиши F будет выделена позиция File основного меню, а при

нажатии клавиши S — позиция Save подменю. Нажатие клавиши ENTER фиксирует ввод команды.

Наиболее распространенные команды можно вводить более просто с помощью функциональных клавиш или их комбинаций с клавишами Ctrl и Alt. Такая возможность будет каждый раз особо отмечаться. Например, для ввода команды Save достаточно нажать клавишу F6, а команды Hoad — клавишу F5.

Отметим и еще один способ ввода команд — с помощью командного режима, устанавливаемого нажатием клавиши Esc. Это возможно в процессе редактирования документа. После нажатия указанной клавиши в командной строке появляется надпись Command: и система ожидает ввода имени соответствующей команды. Его нужно указать полностью, например Save, если вводится команда записи на диск.

Таким образом, управление системой MathCAD производится различными способами, и пользователь волен выбирать любой понравившийся ему способ. Вначале данным простым и наглядным кажется способ с применением меню. Однако по мере освоения системы привлекательными становятся более быстрые способы ввода команд. Переход к ним осуществляется ненавязчиво и постепенно. Далее мы подробно рассмотрим основные команды системы, руководствуясь в основном их местоположением на дереве меню.

## 2.2. Системные команды (System)

Основные системные команды сгруппированы в подменю, относящиеся к позиции System основного меню. Рассмотрим их назначение и возможности.

Help [F1] — помощь. Эта команда вводит с диска файл MCAD.HLP, содержащий очень краткое изложение основных возможностей системы. Подобный файл есть и в текстовом формате — README.TXT. Помощь организована также по типу меню и позволяет быстро навести справки о той или иной команде, функции или операторе. Помощь рассчитана на пользователя, знакомого с системой, и не заменяет ее подробного описания.

Quit [Q] — выход. Данная команда обеспечивает окончание работы с системой и выход из нее в операционную систему MS-DOS или управляющую программу типа Norton Commander. Знаком ^ здесь г далее отмечено нажатие клавиши Ctrl совместно с далее указанной клавишей. Такое сокращение принято в литературе по программированию, и мы не будем нарушать традиции.

Dos. Эта команда дает доступ к командам дисковой операционной системы DOS без выхода из среды MathCAD и потери текущего редактируемого документа. При вводе этой команды в командной строке появляется приглашение DOS. Если в ответ на него нажать клавишу ENTER, то произойдет временный выход из системы MathCAD в систему MS-DOS. Для возврата в среду системы MathCAD необходимо ввести команду EXIT и нажать клавишу ENTER.

В системе MathCAD предусмотрена возможность исполнения команд операционной системы MS-DOS без выхода из среды MathCAD. Для этого достаточно после приглашения DOS указать нужную команду. Можно, например, просмотреть каталог всех файлов, используя команду dir MS-DOS, или вывести на экран дисплея файл, к которому нет доступа изнутри системы. В конце исполнения подобных команд появляется надпись Press any key to return MathCAD (Нажмите любую клавишу для возврата в MathCAD). Выполнив эту рекомендацию, можно вернуться в систему.

Memory — память. Данная команда выдает в командной строке информацию об объеме памяти, занимаемой текущим документом, и объеме свободной памяти. Например, в виде 14 K of 127 K bytes used (14 Кбайт из 127 используется). Следует отметить, что объем памяти, занимаемой документом в оперативном запоминающем устройстве персонального компьютера, заметно больше объема, занимаемого документом на диске. Связано это с тем, что дополнительная память резервируется под значения переменных и массивов, которые в дисковом файле фигурируют лишь в виде своих имен. Для компьютеров с расширенной свыше 640 Кбайт памятью система MathCAD обеспечивает перенос в расширенную область памяти массивов



векторов, матриц и графиков. При этом основная область памяти используется более экономно.

Redraw [R] — перерисовка. Изображение на экране дисплея иногда может портиться в процессе скроллинга (перемещения) или из-за наложения друг на друга блоков и сообщений об ошибках. Данная команда придает изображению его нормальный вид.

Print [O] — печать. Эта команда служит для печати документа принтером, который выбирается с помощью команды Select Printer. В процессе подготовки к печати в командной строке выдается информация о формате печати и порте вывода. Пока Вам неясны запросы системы, нажимайте просто клавишу ENTER в ответ на них, и документ будет распечатан.

Select printer — селекция принтера. Данная команда обеспечивает выбор одного из ряда печатающих устройств — принтеров или графопостроителей (плоттеров). При ее вводе появляется подменю с наименованиями печатающих устройств. Они включают в себя 9- и 24-игольчатые принтеры с шириной строки 80 и 132 символа, лазерные принтеры и плоттеры фирмы Hewlett-Packard. Наборы шрифтов для них хранятся в файлах с расширением .MCF. При селекции принтера нужный файл загружается с диска, а выбранный тип принтера запоминается и при последующих загрузках системы загружается автоматически.

Config save — запись конфигурации системы. *Конфигурацией* системы называется набор выбранных в данный момент команд, тип принтера и дисплея и прочие данные о состоянии системы. Вся эта информация с помощью данной команды может быть записана в специальный конфигурационный файл, который хранится на диске под именем msad.msc. После такой записи система запоминает информацию о ее текущем состоянии и последующая загрузка системы приводит ее в это состояние. При вводе команды система запрашивает имя конфигурационного файла: Save as:msad.msc. При необходимости конфигурационный файл может быть записан под измененным именем. Если файл с указанным именем уже существует, в командную строку будет выдан запрос File exists — OK to overwrite? (Файл существует, писать ли поверх него?). Ответ Y приведет к записи нового конфигурационного файла, а ответ N — к отказу от записи.

Execute — выполнение. Эта команда обеспечивает выполнение специальных командных файлов системы.

Командными файлами в системе MathCAD являются файлы, записанные в текстовом формате. Для их подготовки, просмотра и редактирования можно использовать любой подходящий текстовый редактор, например входящий в состав пакета Norton Communder. К командным файлам, в частности, относится конфигурационный файл msad.msc. Он, как отмечалось, загружается автоматически при загрузке системы.

Приведем пример задания командного файла. Допустим, что мы хотим создать командный файл для документа с именем demo. Командный файл должен делать следующее: перевести систему в режим ручных вычислений вводом команды manual, загрузить документ с помощью команды load demo, задать вычисления до конца документа командой process и перевести курсор на интересующую нас строку командой goto 50. Командный файл в этом случае представляет собой следующий текст:

```
manual
load demo
process
goto 50; Graphics 3 functions
```

После точки с запятой в командном файле можно записывать комментарии. Составленный по этим правилам командный файл нужно записать на диск под именем DG.MSC. Теперь можно использовать такую программу, задав следующую команду:

```
[Esc] execute DG ENTER
```

Исполнение файла DG задаст запланированную последовательность действий системы MathCAD.

## 2.3. Работа с файлами (File)

Основным видом файлов в системе MathCAD являются файлы с текстами документов, имеющие расширение .mcd. Для работы с этими файлами существует ряд команд, относящихся к позиции File основного меню. Они перечислены ниже.

Load [F5] — загрузка. Эта команда служит для загрузки файлов с диска и вывода их на экран дисплея. При задании команды в командной строке системы появляется запрос имени файла File to load. В ответ указывается имя файла в соответствии с правилами, принятыми при работе с файлами в операционной системе MS-DOS. Если файлы расположены в текущей директории системы, то достаточно указать их имя даже без расширения .mcd. В противном случае нужно указать путь к файлу, например B:/MAT/SLE, если файл с именем SLE находится в директории MAT диска B.

Помнить имя загружаемого файла не обязательно. Если вместо ввода имени нажать клавишу ENTER, то на экране появится подменю с именами файлов — до 10 имен. Одна из позиций меню выделена, и выделение можно перемещать по именам файлов с помощью клавиш перемещения курсора вверх и вниз. Можно также перелистывать странички меню с помощью клавиш PgUp и PgDn. Найденный таким образом нужный файл вводится нажатием клавиши ENTER.

Save [F6] — запись. Данная команда обеспечивает запись текущего документа на магнитный диск с расширением .mcd. При вводе этой команды система дает запрос об имени файла Save as: Имя файла. При необходимости указанное в запросе имя файла может быть изменено без указания расширения. Команда завершается нажатием клавиши ввода ENTER.

Append — наложение. С помощью этой команды можно включать в текущий редактируемый документ другой документ и получать в результате новый документ. Таким образом можно готовить документы раздельно и затем объединять их с помощью данной команды. Тем самым реализуются возможности структурного программирования.

Filename — имя файлов. Эта команда используется для переименования файлов данных. Они будут рассмотрены в дальнейшем.

Clear — стирание. Данная команда стирает текущий файл и обеспечивает перезагрузку файла конфигурации. Тем самым восстанавливается исходное состояние системы. Текущий документ и его имя исчезают с экрана дисплея.

Reset — переустановка. Действие этой команды аналогично действию предыдущей команды, но при этом не происходит перезагрузки конфигурационного файла.

Как отмечалось, ряд операций с файлами возможен и с помощью команд дисковой операционной системы, доступной при использовании команды DOS.

## 2.4. Режимы работы системы (Compute)

После того как документ введен в систему MathCAD, она может работать в различных режимах. Они указаны в подменю, относящемся к позиции Compute основного меню. Следует отметить, что при обычной загрузке системы устанавливается режим auto, что индицируется соответствующей надписью в конце командной строки. Рассмотрим режимы работы системы.

Calculate [F9] — вычисление. Эта команда используется в режиме Manual (см. далее) для запуска вычислений блоков, которые расположены от начала документа до конца страницы, в которой находится курсор. Действие команды сопровождается мигающей надписью WAIT. (ждите) и появлением на экране результатов вычислений после знаков =, в шаблонах графиков, матриц и векторов и т. д. Можно заметить также мигание формул, по которым происходят вычисления, а также точки после надписи WAIT.

Process — процесс. Эта команда запускает процесс вычислений по всему документу независимо от текущего положения курсора. Индикация вычислений сопровождается ранее отмеченными признаками.

Automatic — автоматические вычисления. Напоминаем, что эта команда обыч-

но задается по умолчанию. В данном режиме вычисления автоматически производятся с начала документа и до конца текущей страницы. Они происходят по мере скроллинга (перелистывания) страниц и перемещения курсора по тексту документа.

Режим auto создает впечатление, что система работает медленно, поскольку по мере перелистывания документа происходят вычисления, подчас очень сложные и трудоемкие. Поэтому редактировать документы со сложными расчетами в этом режиме неудобно. Этот недостаток, если его можно считать таковым, существенно ослаблен при использовании системы на персональных компьютерах класса IBM PC AT, имеющих повышенную частоту запуска микропроцессора и математический сопроцессор, повышающий скорость счета.

Еще одна причина медлительности при скроллинге — работа системы в графическом режиме. При этом даже малейшее обновление изображения на экране дисплея требует полной регенерации изображения, состоящего из многих десятков тысяч точек.

Manual [F9] — ручной. Этот режим наиболее удобен для редактирования документов. При нем вычисления не производятся. Вместо графиков, матриц и других элементов, относящихся к результатам вычислений, выводятся лишь их шаблоны, что облегчает редактирование и перемещение курсора по всему тексту документов. Чтобы загустить ручную режим вычислений, достаточно исполнить команду Calculate, вводимую нажатием клавиши F9. Напоминаем, что для загрузки системы с исходным режимом работы Manual достаточно провести загрузку основного файла системы в виде mcad /M.

Format — формат. Данная команда задает глобальный, т. е. используемый по всему документу, формат представления чисел. При ее вводе в командной строке появляется надпись

Format:rd=d ct=10 im=i et=15 pr=3

С помощью клавиш управления курсора и стандартных средств редактирования эти данные о формате чисел можно изменить на нужные.

Параметры задания формата имеют следующее назначение:

rd — задает тип используемых чисел (d — десятичные, o — восьмеричные, h — шестнадцатеричные, c — комплексные),

ct — устанавливает представление машинного нуля для комплексных чисел (если  $\text{Re}(Z)/\text{Im}(Z) > 10^{\text{ct}}$ , то z выводится как действительное число, а если  $\text{Im}(Z)/\text{Re}(Z) > 10^{\text{ct}}$  — как мнимое).

im — устанавливает обозначение (i или j) для мнимой единицы (квадратного корня из -1),

et — задает границу для представления чисел в экспоненциальной форме (такую форму числа приобретают, если их значения больше  $10^{\text{et}}$  и меньше  $10^{-\text{et}}$ ).

zt — задает значение машинного нуля при вычислениях (числа, меньшие  $10^{-\text{zt}}$ , воспринимаются как нулевые, значения zt могут лежать в пределах от 3 до 307),

pr — задает количество знаков после десятичной точки (оно может быть от 1 до 15).

Продолжим знакомство с командами, относящимися к позиции Compute.

Randomize — рандомизация. Эта команда настраивает генератор случайных чисел. При ее задании запрашивается значение числа seed (по умолчанию 1). Изменяя это число, можно создавать различные последовательности случайных чисел с равномерным распределением.

Dimension — размерность. Данная команда задает размерность физических величин. При ее вводе в командной строке появляется надпись

M=mass	L=length	T=time	Q=charge
(масса)	(длина)	(время)	(заряд)

При необходимости можно отредактировать эту строку и задать другие обозначения для размерностей физических величин.

Equation (on/off) — равенство (вкл/выкл). Эта команда обеспечивает включение или выключение итерационного процесса решения уравнений. Если команда включена (on), соответствующее математическое выражение помечается знаком

□ и не вычисляется.

Generate matrix [Alt—M] — задание матриц. Данная команда служит для задания векторов и матриц. При ее вводе в командной строке появляется запрос

Array size (rows columns):

(Размер массива (строк столбцов):)

В ответ на этот запрос нужно ввести число строк и столбцов матрицы. Если одно из этих чисел равно 1, то задается вектор. Матрица является двухмерным массивом чисел, а вектор — одномерным. Векторы и матрицы занимают определенный объем памяти. Общее число их элементов при стандартной конфигурации компьютера (объем памяти 640 Кбайт) не должно превышать 8000.

## 2.5. Редактирование документов (Edit/move)

Редактирование документов — одна из важнейших частей работы в системе MathCAD. Неслучайно на нее приходится большая часть команд. Далее будут отдельно описаны приемы редактирования текстовых и графических блоков, так что здесь рассматриваются лишь редактирование вычислительных блоков и некоторые общие приемы редактирования документов.

Значительная часть команд редактирования связана с перемещением и позиционированием курсора. Эти команды отмечены ниже:

↑ ↓	Перемещение курсора вверх и вниз.
← →	Перемещение курсора влево и вправо.
Tab	Перемещение курсора в текстовом блоке на одно слово, в вычислительном и графическом блоках на следующую позицию, между блоками вправо на 10 позиций.
Shift Tab	Перемещение курсора в текстовом блоке назад на одно слово, в вычислительном и графическом блоках на предшествующую позицию, в промежутках между блоками на 10 позиций назад.
PgUp	Перемещение курсора на 5 позиций вверх.
PgDn	Перемещение курсора на 5 позиций вниз.
Ctrl PgUp	Перемещение вверх на 80 % длины страницы.
Ctrl PgDn	Перемещение вниз на 80 % длины страницы.
Ctrl →	Перемещение курсора вправо на 10 позиций.
Ctrl ←	Перемещение курсора влево на 10 позиций.
Home	Перемещение курсора в исходную позицию блока.
End	Перемещение курсора в конец блока.
Ctrl Home	Перемещение курсора в начало документа.
Ctrl End	Перемещение курсора в конец документа.
ENTER	Перевод курсора в текстовом или графическом блоке в нижестоящий блок, а в промежутке между блоками перевод курсора в начало следующей строки.

Одна из важнейших операций редактирования — ввод символов и цифр. Ввод всегда производится по месту текущего положения курсора. При этом следует учитывать особые правила ввода операторов — например, знак: = вводится набором знака =. При вводе формул следует скобками указывать порядок действий. Например:

$$Y:2/3+4 \quad \text{дает } Y:=\frac{2}{3}+4$$

$$Y:2/(3+4) \quad \text{дает } Y:=\frac{2}{3+4}$$

В этих примерах видна принципиальная роль скобок при вводе — даже если они в окончательном выражении отсутствуют. Ненужные после окончательной трансформации выражений скобки видны лишь тогда, когда курсор находится в вычислительном блоке. При выводе его из блока в выражениях остаются только принципиально необходимые скобки.

При редактировании часто приходится стирать неверно набранные знаки. Для этого используются две клавиши стирания — клавиша с надписью Del в блоке цифровых клавиш и клавиша, расположенная над клавишей ENTER в основном блоке клавиш. Стираемый символ отмечается курсором. При уничтожении одной из скобок автоматически уничтожается вторая скобка. Это ведет к принципиальным изменениям выражений, к которым применяются операции уничтожения символов.

В описании системы MathCAD приведено много примеров правильного применения операций уничтожения и вставки символов. Однако опыт показал, что большинство этих операций интуитивно предсказуемо, и после одного—двух часов работы с системой данные операции выполняются почти автоматически.

Редактирование сильно облегчается благодаря ряду команд по редактированию блоков. Они расположены в подменю позиции Edit/move основного меню. Отметим возможности этих команд:

Сору [F2] — копия. Эта команда копирует блок (любой) в специальный буфер памяти. Блок при этом остается на своем месте.

X(cut) [F3] — отклейка. Данная команда копирует блок в буфер и одновременно стирает его. Операция напоминает использование клейкой ленты для переноса на нее картинки, нарисованной легко отслаивающейся краской.

Paste [F4] — перенос или подклейка. Переносит копию блока из буфера на экран дисплея и помещает ее на место, указанное текущим положением курсора. Оно соответствует левому верхнему углу блока. Напоминает перевод картинки с липкой ленты опять на бумагу. Перенос возможен, если место не занято другим блоком, — о такой ситуации может появиться предупреждение в командной строке. Тогда нужно перевести курсор на свободное место и повторить операцию переноса.

Separate — разделение. Эта команда используется для разделения блоков и сообщений об ошибках. Нередко сообщений об ошибках так много, что надписи о них заслоняют блоки и прочитать их невозможно. Команда раздвигает блоки так, что их содержание и сообщения об ошибках становятся доступными для наблюдения.

Insertline [F9] — вставка линии. Данная команда вставляет пустую строку в место документа, указанное курсором, сохраняя при этом блоки в целости. Пустая строка может быть вставлена, если на линии расположения курсора нет текстового блока.

Deleteline [F10] — уничтожение линии. Команда уничтожает пустую строку документа, на которую установлен курсор.

Goto — переход. Команда по запросу Goto:N1N2 обеспечивает переход курсора к строке N1 и столбцу N2 в ней. Если это место находится за пределами экрана, происходит скроллинг изображения. Число N2 может не указываться — тогда оно полагается равным нулю.

Move — перемещение. Команда аналогична команде Goto, но с той разницей, что перемещение производится относительно координат текущего положения курсора.

Find [F5] — поиск. Эта команда ищет по всему документу текст, указанный по запросу Search for: Текст. Если текст найден, курсор помещается в конец фрагмента текста. С помощью клавиш + и — можно продолжить поиск таких документов после и до положения курсора.

Replace [F6] — перемещение. Данная команда служит для замены одних фрагментов текста другими. Вначале она, как и предшествующая команда, ищет заданный фрагмент текста. Затем по запросу в командной строке Replace with: Текст она вставляет введенный текст на место найденного. Однако перед этим в командной строке появляется предупреждение-запрос: Yes, No, Quit or All (y, n, q, a). Ответ y подтверждает выполнение команды, n ведет к отказу выполнения, q — к выходу из режима поиска и a — к замене введенным текстом найденных фрагментов по всему документу.

Применение этих команд позволяет проводить быстрые операции по редактированию документов, включая уничтожение и перекомпоновку блоков, замену как

отдельных символов, так и целых участков текста документов. Иногда полезно наблюдать границы блоков в документе. Для этого используется команда, вводимая одновременным нажатием клавиш Ctrl и V. При этом каждый блок обводится прямоугольной рамкой. Повторное нажатие этих клавиш ведет к уничтожению рамки.

## 2.6. Редактирование текстов (Text)

Текстовые описания и комментарии намного повышают наглядность и понятность документов в системе MathCAD. При необходимости в системе можно готовить научно-технические статьи, отчеты и главы книг. Для этого в нее включены средства редактирования текстовых блоков. Конечно, они уступают изысканным средствам, имеющимся в специализированных текстовых редакторах (например, таких, как ChiWriter, MultiEdit, Words и др.) или компьютерных настольных издательских системах. Однако для обычного пользователя это, скорее, достоинство, чем недостаток. Тем более если учесть то, что система MathCAD воспроизводит математические знаки в форме, общепринятой для научно-технической литературы, и позволяет сочетать их с превосходными графиками и таблицами.

Возможны два типа текстовых блоков. Первый задается режимом text region (текст в ограниченной области). Как отмечалось, он вводится знаком кавычек « $\langle$ ». При этом в месте расположения курсора (если оно не занято блоком) появляется пара кавычек « $\langle$ », между которыми можно вводить текст. Нажатие клавиши ENTER в конце ввода строки переводит курсор и завершающие кавычки на новую строку. Таким образом можно вводить несколько строк текста. Размер такого текстового блока по вертикали определяется числом строк, а по горизонтали — длиной наиболее длинной из введенных строк. С помощью команды Width, рассмотренной ниже, можно ограничить максимальную длину строки.

Второй тип блока задается в режиме text band (небуквальный перевод — текст в предельной области). Этот режим вводится в виде соответствующей команды. При нем на экране появляется также пара кавычек, но одни из них помещаются в начале строки, а другие в ее конце. Длина строки при этом максимальна.

Тексты в файлах .MCD, соответствующие режиму text region, записываются в особой форме — они содержат перед каждым блоком указания его координат и размеров. А тексты в режиме text band записываются в обычном текстовом формате. Их без помех можно прочитать с помощью любого текстового редактора. Более того, в такие тексты можно вставлять тексты, подготовленные с помощью дополнительных текстовых редакторов.

Рассмотрим основные команды, относящиеся к позиции основного меню.

Width — ширина. Эта команда задает максимальную длину строки текстового блока в режиме text region. Если при вводе какого-либо слова длина строки превышает установленное значение, то слово целиком переносится на следующую строку.

Mark [ $\wedge$  X] — маркировка. Данная команда помещает знак, на котором установлен курсор. Выделение знака происходит цветом или инвертированием изображения в монохромных дисплеях. Обычно команда используется дважды — для отметки начала выделенного фрагмента текста и отметки его конца. При этом признаки выделения проявляются по отношению ко всему фрагменту текста.

Text band [ $\wedge$  T] — установка режима ввода больших фрагментов текста. Эта команда устанавливает описанный выше режим ввода текста text band.

Center — центровка. Данная команда помещает текст в строке, на которой находится курсор, в середину строки.

Backward [ $\wedge$  B] — назад. Команда интерпретирует команды перемещения курсора как перемещение назад.

Forward [ $\wedge$  F] — вперед. Эта команда интерпретирует команды перемещения курсора как команды перемещения вперед.

Justify [ $\wedge$  N] — выравнивание. Данная команда обеспечивает выравнивание строк и размещение в них доступного числа целых слов.

Для удобства редактирования текстов имеется ряд дополнительных команд,

вводимых нажатием определенных комбинаций клавиш. Они отмечены ниже:

Ctrl M	Перемещение курсора на одно слово.
Ctrl L	Перемещение курсора на одну строку.
Ctrl S	Перемещение курсора на одно предложение.
Ctrl P	Перемещение курсора на один параграф.
Tab	Перемещение курсора вправо на одно слово.
Shift Tab	Перемещение курсора влево на одно слово.

В первых четырех командах направление перемещения задается ранее описанными командами Forward и Backward.

Специальная позиция основного меню In region (входа в область) содержит еще три важные команды для работы с маркированными фрагментами текста. Incopy [Ctrl F2] — копирование. Команда помещает копию выделенного участка в буфер, не уничтожая текст.

Incute [Ctrl F3] — отклейка. Данная команда обеспечивает помещение маркированного участка текста в текстовый буфер и уничтожает этот текст на экране.

Inpaste [Ctrl F4] — подклейка. Эта команда помещает текст из текстового буфера на место, указанное курсором.

Данные команды позволяют легко манипулировать любыми кусками текстов — вплоть до целых параграфов и страниц. В сочетании с ранее описанными средствами редактирования они упрощают разработку материалов в среде MathCAD.

## 2.7. Управление окнами и форматирование страниц (Window/page)

Для редактирования сложных документов система MathCAD позволяет перейти к двухоконному режиму вывода информации. Окна — это прямоугольные области экрана. В системе MathCAD второе окно появляется вслед за первым и частично его заслоняет. В каждом окне может находиться свой документ. Активным называют то окно, в котором в данный момент находится курсор.

Для управления окнами в системе MathCAD имеется три команды, относящиеся к позиции Window/page основного меню.

Split [F7] — разделить. Эта команда разделяет экран на два окна. Линию раздела можно перемещать следующим образом: установить курсор на желаемую линию раздела и нажать клавишу F7.

Unsplit [<sup>^</sup>F7] — объединить. Команда по смыслу противоположна первой, т. е. уничтожает второе окно — остается лишь пассивное окно.

Jump [F8] — перескочить. Эта команда перебрасывает курсор из одного окна в другое, т. е. фактически задает активность одного из окон.

Двухоконный режим удобен, когда требуется значительная переработка документа, например если большие его части из конца нужно переместить в начало, а оно занято другими блоками. Он полезен и в том случае, когда надо быстро и наглядно сравнить два варианта решения задачи. Разумеется, с помощью текстового буфера можно копировать блоки из одного окна в другое. Можно независимо друг от друга считывать с диска в окно нужные файлы или записывать документы с любого окна на диск.

Вторая часть позиции Window/page относится к управлению форматом страниц при их распечатке принтером или плоттером. При этом имеются следующие команды:

Margin — отступ. Эта команда задает ширину отступа в левой части страницы. При ее исполнении появляется запрос left margin=0 (т. е. ширина отступа равна нулю). Ширина отступа задается в знакоместах. Если отступ нужен, подставьте вместо нуля нужное значение ширины отступа.

Page length — длина страницы. Данная команда задает длину страницы, т. е. число строк в ней. При ее задании появляется запрос Page length=0. Указание 0 задает страницу бесконечной длины (действует по умолчанию). Если нужна страница заданной длины, замените нуль на нужное число строк в странице.

Breakpage — обрыв страницы. Эта команда задает построение линии раздела страниц в виде точечной линии.

Insert pagebreak — включение линии обрыва страницы. Данная команда формирует сплошную линию раздела страниц с характерным загибом слева — напоминает вытянутый поперек всей страницы знак квадратного корня. По этой линии обычно производится обрезка страниц.

Система MathCAD может задавать печать документов с шириной страницы до 160 знаков (отсчет номеров столбцов с 0 до 159). Однако большинство принтеров при стандартной печати имеют в строке 80 или 132 знакоместа. Система MathCAD обеспечивает не только получение такой ширины страниц, но и печать лишь выделенной части документа.

Печать документов производится при вводе команды print. В ответ в командной строке появляется запрос

Print area N1, N2 to N3, N4

Здесь числа N1 и N2 задают положение левого верхнего угла арены печати, а числа N3 и N4 — нижнего правого угла. По умолчанию они характеризуют начало и конец документа в активном окне. Если необходима печать всего документа, нажмите клавишу ENTER. В противном случае введите параметры, задающие нужный размер арены документа, выводимой на печать, и также нажмите клавишу ENTER. Поступит еще один запрос о порте, через который идет печать, — по умолчанию это наиболее распространенный параллельный порт PRN. Теперь для начала печати надо нажать клавишу ENTER.

В некоторых случаях желательна реализация особых возможностей печати, например использование полужирного, курсивного и других видов шрифтов. Для этого на принтер следует послать специальные последовательности управляющих кодов — ESCP. Символ ESCP означает передачу десятичного кода 027, за которым следует один или несколько управляющих кодов и кодов знаков, выводимых на печать.

Система MathCAD в текстовых блоках позволяет вводить управляющие коды после знака /. При этом если нет кода ESCP 027, то любой код после знака / приведет к выводу символа с его кодом на экран дисплея и печать этого символа принтером. Например, если ввести в текстовый блок строку

«Это личико»/001»

то вслед за стрелкой появится изображение человеческого лица, так как именно такой символ закодирован кодом 1 (заметьте, что коды надо указывать всегда тремя цифрами, т. е. 1 следует вводить как 001). Характер действия управляющих кодов для каждого типа принтера указан в его описании. Наиболее распространены управляющие коды для принтеров фирм IBM и Epson.

Настройку принтера можно изменить с помощью специального *драйвера печати* — в системе MathCAD это файл mscad.msp. Данный файл записан также в текстовом формате, и его легко просмотреть и отредактировать с помощью внешнего редактора.

## 2.8. Различия версий системы MathCAD

Версии системы MathCAD 1.0 и 1.1 появились в 1986 г. и в настоящее время уже устарели. Версия 2.50, появившаяся в 1989 г., у нас в стране пока известна мало. Поэтому материал данной книги посвящен версиям 2.01 и 2.04 (1987, 1988 гг.). Новые возможности версии 2.50 будут отмечены в данном разделе.

В версиях 2.01 и 2.04 в сравнении с версией 1.1 включены ряд новых векторных и матричных операторов и функций, функции linterp, arg и ii (см. гл. 4), данные файлового типа, автоматическое масштабирование графиков, графики столбчатого типа, улучшенные средства редактирования (в частности, команды стирания и переноса блоков).

В версии 2.50 расширено использование ниспадающих меню. В частности, с их помощью задаются форматы графиков. Например, формат 2D-графиков (двух-



мерных) задается вводом курсора в блок графики и нажатием клавиши F. Появляется следующее подменю:

Size=15,30	Размер (строк, столбцов)
Trace Types=1	Тип линий, меток.
Log Cicles=0,3	Число циклов логарифмического графика.
Subdivisions=4,1	Число интервалов сетки.
Global Default	Глобальная установка данных.
Fever	Возврат к исходным данным.
Done	Продолжение работы.

Это отличие не является принципиальным. Более того, оно не очень удобно, особенно для пользователей компьютеров (IBM PC, XT и jr) с невысоким быстродействием. Связано это с тем, что вывод подменю в рабочее поле экрана нарушает изображение, и требуется значительное время для его восстановления. Для компьютеров такого класса предпочтительно применение версий 2.01 и 2.04. Они имеют и меньшее число обращений к диску в ходе работы, и меньшее время загрузки системы.

К принципиальным отличиям версии 2.50 следует отнести введение операторов сортировки элементов векторов и матриц, реализацию трехмерной 3D-графики и импорт графических изображений из других графических систем, например TurboCAD и AutoCAD. Это заметно увеличивает возможности системы, в частности позволяет использовать в документах сложные иллюстрации.

Действие операторов сортировки достаточно очевидно и полно описано во встроенной системе помощи, вводимой в действие нажатием функциональной клавиши F1. Кстати, система помощи в версии 2.50 несколько улучшена и дает более полную информацию чем в ранних версиях.

Трехмерная графика требует задания функции двух переменных  $f(x, y)$ . Бычисленные по ряду значений  $x$  и  $y$  значения этой функции используются для задания двухмерной матрицы с условным именем M. Для построения графика достаточно в нужном месте, указанном курсором, ввести шаблон графика. Для этого нажимаются клавиши Alt и 2. В левом нижнем углу шаблона вместе светлого прямоугольника нужно указать имя матрицы M.

Документ 2.1 иллюстрирует построение трехмерной поверхности. Можно заметить, что примененный алгоритм не обеспечивает исключения невидимых линий изображения.

Формат 3D-графиков также задается вводом в его шаблон курсора и нажатием клавиши F. Подменю форматирования имеет следующий вид:

Size=15,30	Размеры (строк, столбцов).
Rotation=10	Угол вращения вокруг оси z.
Tilt=35	Угол вращения вокруг оси x.
Vertical Scale=3	Вертикальный масштаб.
Global Default	Глобальная установка данных.
Revent	Возврат к исходным данным.
Done	Продолжение работы

Изменив формат, можно представить трехмерную фигуру в наиболее наглядном виде. В частности, можно изменять углы, под которыми видна фигура.

Для импорта графических изображений используются файлы, создаваемые другими графическими системами. Они должны быть преобразованы в файлы с расширением .mcs. Для этого в версии 2.50 имеется специальный файл трансляции mostrans. В DOS вводится команда

mostrans Имя файла. Расширение

В результате преобразования получается файл с исходным именем и расширением .mcs, который загружается в систему.

Для построения рисунка, импортируемого из другой системы, задается шаблон нажатием клавиш Alt и 5. В левом нижнем углу на месте светлого прямоугольника нужно ввести имя преобразованного файла без расширения. Документ 2.2 иллюстрирует построение стилизованного изображения космического челнока «Колумбия», созданного с помощью графической системы AutoCAD.

A 3D wireframe plot of a function, likely representing a surface of a function of two variables. The surface is depicted as a grid of lines forming a saddle-like shape, curving upwards in one direction and downwards in another. The plot is rendered in black lines on a white background, with a perspective view showing the surface's curvature.

## Док. 2.1. Построение трехмерной фигуры

## Док. 2.2. Построение рисунка, импортируемого из другой системы

Весьма полезной является возможность версии 2.50 изменить положение и размеры импортируемого рисунка. Для этого используется команда форматирования, создающая следующие подменю:

Size = 15,30	Размер (строк, столбцов).
Magnification = 1,1	Увеличение размеров.
Center = .5,.25	Положение центра.
Box = y	Рамка (у — есть, n — нет).
Global Default	Глобальная установка данных.
Revert	Возврат к исходным данным.
Done	Продолжение работы.

От одной версии к другой увеличивается число типов принтеров, с которыми работает система. Если первые версии поддерживали работу с несколькими типами принтеров, то версия 2.50 — с 38 типами принтеров, в том числе с лазерными и многоцветными принтерами, плоттерами и графопостроителями.

Версия 2.50 ориентирована на персональные компьютеры класса IBM PC AT и PS/2, снабженные математическими сопроцессорами. Однако, как отмечалось, она может использоваться и на более ранних моделях компьютеров, хотя для них удобнее применять версии 2.01 и 2.04.

Новейшая версия системы MathCAD 3.0 ориентирована на персональные компьютеры с микропроцессором 80386 и сопроцессором 80386 [15]. В нее введены элементы символьных вычислений, при которых результаты вычислений могут быть представлены в аналитическом виде. Символьный процессор системы базируется на пакете Maple фирмы Waterloo Maple Software.

В набор встроенных операций введены две новые функции — для нахождения собственных значений и собственных чисел вещественных матриц. В эту версию включен также электронный справочник, содержащий подсказки по многим математическим формулам. Для облегчения ввода спецзнаков используются цветные криптограммы (отчетливое наблюдение их, однако, возможно только при использовании дисплея с высоким разрешением и большим размером экрана).

Фирма MathSoft Inc наряду с самой системой выпускает 10 пакетов применения (MathCAD Applications Packs) по цене 95 дол. за каждый (сама система стоит около 500 дол.). В каждом пакете содержится полтора—три десятка примеров применения системы: в математике, физике, механике, электротехнике и т. д. Наличие таких пакетов заметно расширяет возможности системы и позволяет ориентировать ее на конкретные задачи пользователя. Два крупных пакета применений (в математике и электрорадиотехнике), содержащих около 150 примеров, подготовлено автором в учебно-научном центре по компьютерным и информационным технологиям «КИТ» (часть из них приведена в книге).

В последнее время при разработке документов используют следующий прием — все вспомогательные вычисления располагают в невидимой правой части экрана после слова equation. При этом обычно используются глобальные переменные со знаком присваивания  $\equiv$ . Все вычисления ниже слова equation выполняются как вычисления в одном блоке, и их результаты доступны в любом месте документа. Этот прием заметно облегчает разработку учебных программ и избавляет пользователя от созерцания сложных, но подчас ненужных подробностей вычислений. Применение этого приема возможно, начиная с версии 2.01, хотя оно специально не оговорено.

### 3. АЛФАВИТ И ТИПЫ ДАННЫХ СИСТЕМЫ MathCAD

#### 3.1. Идентификаторы и алфавит

Алфавит системы MathCAD содержит строчные и прописные латинские буквы, арабские цифры, ряд греческих букв и специальных знаков. С их помощью задаются имена встроенных функций и операторов, а также идентификаторы — имена

вводимых пользователем переменных, констант и функций.

Идентификаторы должны начинаться с буквы, и их имена должны быть уникальными, т. е. единственными в своем роде. В состав идентификаторов могут входить цифры и имена встроенных функций, но при наличии отличительных дополнений. Следующие идентификаторы являются допустимыми:

x, X, Alfa, Start\_Value, U1, FSin и т. д.

А вот эти идентификаторы недопустимы:

1U — начинается с цифры,

sin — совпадает с именем встроенной функции,

альфа — содержит нелатинские буквы.

Греческие буквы вводятся нажатием клавиши Alt и одной из клавиш, указанных ниже:

A→α B→β D→δ E→ε F→Φ G→Γ H→Φ I→∞ L→λ  
N→η O→Ω P→π Q→θ R→ρ S→σ T→τ U→μ W→ω

В текстовых блоках могут использоваться знаки кириллицы — алфавита русского языка. Для этого знаки должны быть загружены с помощью специального драйвера или находиться в ПЗУ компьютера. Однако с их применением связаны определенные проблемы.

Дело в том, что каждый знак алфавита кодируется соответствующим кодом. Основные знаки алфавита и некоторые управляющие символы имеют коды от 0 до 127. Вспомогательные или дополнительные знаки имеют коды от 128 до 255. Например, упомянутые буквы греческого алфавита имеют коды от 224 до 239. Знаки кириллицы при наиболее распространенной альтернативной кодировке имеют коды от 128 до 175 и от 128 до 240.

Таким образом, коды знаков кириллицы перекрываются кодами греческих букв. Поэтому их нельзя применять одновременно. В некоторых драйверах имеется возможность переключения алфавитов, что частично решает эту проблему. Однако она всплывает при попытке распечатки документов принтером, так как кодировка знаков у знакогенераторов принтеров также не допускает одновременного применения знаков кириллицы с кодами от 224 до 239 и греческих букв.

Для дисплеев и принтеров с альтернативной кодировкой знакогенераторов ряд символов (в основном греческие буквы) воспроизводятся и печатаются как русские буквы (кроме знака ≈):

α→ρ β→с δ→ы ε→ю Φ→э Γ→т π→у  
θ→щ σ→х τ→ч μ→ц ∞→ь ≡→Е ≈→←

К сожалению, этот недостаток при альтернативной кодировке непреодолим, и надо жертвовать выводом либо греческих, либо строчных русских букв. Греческие буквы можно вводить их именами, например alpha, beta, pi и т. д. При обнаружении на распечатках документов (или на экране дисплея) вместо греческих букв русские следует интерпретировать их как знаки, отмеченные выше.

В состав алфавита системы MathCAD входит и множество специальных математических символов, например знак суммы Σ, интеграла ∫ и т. д. Ввод этих знаков производится при наборе определенных комбинаций клавиш. Иногда знаки на клавишах не совпадают с теми символами, которые вводятся ими. Так, знак двоеточия : вводит символ присваивания :=. Полный набор этих знаков и правила их ввода будут описаны в гл. 4.

### 3.2. Десятичные, восьмеричные и шестнадцатеричные числа

Числа являются важными типами данных, которыми оперирует система MathCAD. Как и в математике, основными являются десятичные, восьмеричные и шестнадцатеричные числа. Система MathCAD не оперирует такими, больше программистскими, чем математическими, понятиями, как целое короткое и длинное число, короткое или длинное число с плавающей точкой и т. д. Говоря точнее, система не выделяет эти числа в отдельные типы данных, а рассматривает их как частные типы, относящиеся к указанным основным типам.

Десятичные числа — это числа с основанием 10. Один разряд таких чисел мо-

жет иметь одно из десяти возможных значений: 0, 1, 2, 3, 4, 5, 6, 7, 8 или 9. Значения чисел лежат в пределах от  $10^{-307}$  до  $10^{307}$ . По умолчанию указываются три знака числа после разделительной точки при экспоненциальном представлении чисел. Способы задания формата десятичных чисел были описаны в § 2.4. Ввод числовых данных может выполняться в любой форме, столь же ненавязчиво, как это делается в обычных математических расчетах. Например, число 123.456 можно ввести как  $0.123 \cdot 10^3$  или  $123456 \cdot 10^{-3}$  и т. д.

Восьмеричные числа имеют основание 8, и один разряд их может иметь значения 0, 1, 2, 3, 4, 5, 6 и 7. Эти числа в конце помечаются буквой O (от слова octal — восемь).

Шестнадцатеричные числа имеют основание 16. Их один разряд с эквивалентными десятичными значениями (DEC) соответствует следующим значениям:

HEX	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
DEC	0	1	2	3	4	5	6	7	8	9	11	12	13	14	15	16

Здесь десятичные значения поясняют вес разряда. Чтобы отличить шестнадцатеричные числа от десятичных и восьмеричных первые помечаются в конце символом H (от слова hexagonal — шестнадцать).

Если шестнадцатеричное число начинается с буквы, например ABCH, то система MathCAD путает его с идентификатором и сообщает об ошибке — неопознанный идентификатор. Чтобы избежать этого, шестнадцатеричные числа нужно начинать с цифры 0.

Восьмеричные и шестнадцатеричные числа используются обычно для компактного указания адресов и целых чисел в вычислительной технике. Поэтому диапазон их значений ограничен и отвечает потребностям 32-битового представления информации.

В физических и технических расчетах величины часто указываются со своей размерностью. Система MathCAD позволяет не только учитывать размерности величины, но и производить их преобразования. По умолчанию заданы следующие размерные величины: время (time), длина (length), масса (mass) и заряд (charge). Для величин с размерностью к их значениям приписывается буква, с которой начинается латинское обозначение размерности.

### 3.3. Комплексные числа

Комплексные числа представляются в алгебраической форме:

$$Z = \text{Re}(Z) + \text{Im}(Z)i$$

или

$$Z = \text{Re}(Z) + \text{Im}(Z)j,$$

где символ  $i$  или  $j$  означает квадратный корень из  $-1$ ;  $\text{Re}(Z)$  — действительная и  $\text{Im}(Z)$  — мнимая части числа. Символ мнимой единицы и форматы задания комплексных чисел были описаны в § 2.4.

Помимо алгебраической существует показательная форма представления комплексного числа:

$$Z = M \cdot e^{i\varphi} \text{ или } Z = M \cdot e^{j\varphi},$$

где  $M$  — модуль комплексного числа и  $\varphi$  — его фаза.

Число, содержащее только мнимую часть, называется мнимым. В системе MathCAD предусмотрено, что если действительная или мнимая часть мала, то комплексное число  $Z$  представляется соответственно как мнимое или как действительное число. Границы такого представления устанавливаются значением параметра  $ct$  в команде format (см. ее описание в § 2.4).

Перед использованием комплексных чисел должна быть задана мнимая единица, т. е. корень квадратный из  $-1$ . Это значение должно быть присвоено переменной  $i$  или  $j$ . С комплексными числами могут употребляться многие функции и операторы системы MathCAD, в том числе матричные и векторные. В этом случае их аргумен-

том указывается число  $Z$ . Функции  $\text{Re}(Z)$  и  $\text{Im}(Z)$  обеспечивают выделение действительной и мнимой частей аргумента  $Z$ .

Комплексные числа широко применяются в различных математических расчетах. Они играют важную роль в электротехнических и радиотехнических расчетах. Например, на их применении основан символический метод расчета линейных электрических и электронных цепей на переменном токе.

### 3.4. Константы, переменные и равенства

Константы — это типы данных, имеющие неизменное значение во всей программе. Строго говоря, в системе MathCAD имеется единственный тип констант — числовые. Это просто числа, например 123,  $12 \cdot 10^{-5}$ , 0.345,  $12.34 \cdot 10^{123}$  и т. д.

Условно к константам можно отнести и предварительно определенные переменные:

$\pi = 3.14159...$	Число пи.
$e = 2.71823...$	Основание натурального логарифма.
$\infty = 10^{307}$	«Бесконечность».
$\% = 0.01$	Процент.
$\text{TOI} = 0.001$	Погрешность численных методов.
$\text{ORIGIN} = 0$	Нижняя граница индексации массивов.
$\text{PRNCOLWIDTH} = 8$	Число столбцов — параметр оператора WRITEPRN.
$\text{PRNPRECISION} = 4$	Число десятичных знаков, используемых оператором WRITEPRN.

Эти переменные имеют указанные значения после загрузки системы. Однако их значения могут переопределяться по ходу выполнения программы. Не рекомендуется их использование не по прямому назначению.

Переменная — имеющий имя (идентификатор) элемент языка системы MathCAD, который может нести определенное и неоднократно изменяемое по ходу выполнения документа числовое значение.

Присваивание переменной определенного значения производится с помощью оператора присваивания  $:=$ , например

$1 := 1$  X:  $= 12.345$  EPS  $= 1 \cdot 10^{-6}$  и т. д.

Знак присваивания вводится набором знака двоеточия.

До присваивания переменной какого-либо значения ее применять нельзя. В противном случае появляется сигнал ошибки в виде надписи undefined (не определена). Документ 3.1 поясняет правильное и неправильное применение переменных. Присваивание можно также осуществлять с помощью знака  $\equiv$ . Переменные в этом случае будут глобальными, и операцию присваивания можно применять в любом месте документа.

Для вывода значений переменных после их имени ставится знак равенства  $=$ . Например

X:  $= 123$   
X  $\equiv 123$

Форма вывода зависит от установленного формата вывода. Он может быть глобальным, т. е. относиться ко всей программе, и локальным, т. е. имеющим ограниченную одним числом область действия.

Глобальный формат устанавливается с помощью ранее описанной команды format, вводимой в командном режиме или с помощью меню. Локальный формат вывода задается следующим образом: в блок вывода значения переменной, таблицы или матрицы помещается курсор и нажимается клавиша F. Дальнейшие операции аналогичны для описанной команды format, но область их действия ограничена выбранным блоком (см. док. 3.2).

В системе MathCAD возможно определение единиц измерения физических величин. Для этой цели используется знак  $\equiv$ . Например, если нужно определить  $\text{cm}$  как единицу длины, то следует записать

$\text{cm} \equiv 1\text{L}$

Переменная в системе MathCAD является имеющим уникальное имя элементом языка, которому можно присваивать любое допустимое значение по всему тексту документа.

Для присвоения переменной значения используйте

конструкцию **ИМЯ\_ПЕРЕМЕННОЙ := Значение**

Для вывода значения переменной используйте

конструкцию **ИМЯ\_ПЕРЕМЕННОЙ =**

**Примеры:**

<b>ONE := 1</b>	Переменной ONE присвоено значение 1
<b>ONE = 1</b>	Выведено значение переменной ONE
<b>Lotus123 := 2</b>	Переменной Lotus123 присвоено значение 2
<b>Lotus123 = 2</b>	Выведено значение переменной Lotus123
<b>DVar := e</b>	Переменной DVar присвоено значение константы e
<b>DVar = 2.718</b>	Выведено значение переменной DVar
<b>true := 1</b>	Введена переменная ИСТИННО true
<b>false := 0</b>	Введена переменная НЕ_ИСТИННО false
<b>X := 5</b>	Задаю значение переменной X
<b>Y := DVar · <math>\frac{4}{X}</math></b>	Переменные в арифметическом выражении
<b>Y = 2.175</b>	Вывод значения арифметического выражения
<b>Z = *</b>	Вывод необъявленной ранее переменной
<b>undefined</b>	ведет к появлению сообщения об ошибке
<b>e := 10</b>	Системной переменной e присвоено значение 10
<b>e = 10</b>	Выведено новое значение системной переменной e
<b><math>e^2 = 100</math></b>	Переменная e утратила начальное значение $e=2.71\dots$ , так что $e^2$ уже не равно $\exp(2)$

### Док. 3.1. Применение переменных

Примеры операций с величинами, имеющими размерность, приведены в док. 3.3.

Часто используются циклические вычисления, т. е. повторяющиеся заданное число раз. Например, необходимо вычислить некоторое число значений какой-либо функции  $f(x)$ . Для этого нужно получить ряд значений аргумента  $x$ . Циклические вычисления характерны и для итерационных и рекуррентных методов.

Для задания циклических вычислений с целочисленной управляющей переменной цикла используется следующая конструкция:

**Имя переменной := Начал..Нкон**

Здесь знак .. вводится набором знака ;. Начал — начальное значение переменной

Система MathCAD выводит данные, например значения переменных, с точностью 3 знака после запятой с округлением последнего знака ( по умолчанию ). Глобальный формат устанавливается командой `format`. Для установки локального формата введите курсор в вычислительный блок и нажмите клавишу F.

Примеры вывода в глобальном и локальном форматах:

$e = 2.718$	формат по умолчанию <code>pr=3</code>
$e = 2.71828183$	Локальный формат <code>pr=8</code>
$e = 2.718281828459045$	Локальный формат <code>pr=15</code>
$\frac{1}{3} = 0.333$	формат по умолчанию <code>pr=3</code>
$\frac{1}{3} = 0.333$	Локальный формат <code>pr=3</code>
$\frac{2}{3} = 0.667$	Иллюстрация округления
$X := 0.00000001$	Иллюстрация перехода к
$X = 1 \cdot 10^{-8}$	экспоненциальному представлению

$M := \begin{bmatrix} 1 & 1 \\ 3 & 4 \\ 1 & 1 \\ 5 & 6 \end{bmatrix}$	Задание матрицы M
---	-------------------

$M = \begin{bmatrix} 0.333 & 0.25 \\ 0.2 & 0.167 \end{bmatrix}$	Вывод матрицы в глобальном формате
---	------------------------------------

$M = \begin{bmatrix} 0.3333333333 & 0.25 \\ 0.2 & 0.166666667 \end{bmatrix}$	Вывод матрицы в локальном формате
--	-----------------------------------

### Док. 3.2. Вывод данных в глобальном и локальном форматах

и  $N_{\text{кон}}$  — конечное значение переменной. Если  $N_{\text{нач}} < N_{\text{кон}}$ , то шаг изменения переменной равен  $+1$ , а если  $N_{\text{нач}} > N_{\text{кон}}$  — то  $-1$ .

Переменные такого типа в системе MathCAD называются переменными с заданными пределами изменения. Шаг изменения можно задать любым, используя другую конструкцию задания таких переменных:

Имя переменной  $:= N_{\text{нач}}, N_{\text{след}}..N_{\text{кон}}$

где  $N_{\text{след}}$  — следующее за  $N_{\text{нач}}$  значение переменной. Шаг в этом случае равен  $N_{\text{след}} - N_{\text{нач}}$ .



$\text{cm} \equiv 1\text{L}$  Сантиметр есть единица длины (length)  
 $\text{g} \equiv 1\text{m}$  Грамм есть единица массы (mass)  
 $\text{sec} \equiv 1\text{t}$  Секунда есть единица времени  
 $\text{ns} \equiv 1 \cdot 10^{-9} \text{ sec}$  Наносекунда есть  $10^{-9}$  секунды  
 $W := 100 \text{ cm}$  Задана длина W, равная 100 см  
 $W = 100 \text{ length}$  Выведена длина W  
 $P := 100 \cdot \text{g}$  Задана масса P, равная 100 г  
 $P = 100 \cdot \text{mass}$  Выведена масса P  
 $T := 20 \cdot \text{sec}$  Задано время T, равное 100 с  
 $T = 20 \cdot \text{time}$  Выведено время T  
 $Tf := 1 \cdot \text{ns}$  Задано время T, равное 1 нс  
 $Tf = 1 \cdot 10^{-9} \cdot \text{time}$  Выведено время T (в секундах)  
 $\text{dyne} \equiv \text{g} \cdot \frac{\text{cm}^2}{\text{sec}^2}$  Определена новая величина - дина  
 $\text{dyne} = 1 \cdot \text{mass} \cdot \text{length} \cdot \text{time}^{-2}$  Выведена дина  
 $\text{kg} \equiv 1000 \cdot \text{g}$  Задана единица массы kg  
 $\text{m} \equiv 100 \cdot \text{cm}$  Задана единица длины m  
 $P := 100 \cdot \frac{\text{kg}}{\text{m}^2}$  Определено давление P  
 $P = 10 \cdot \text{mass} \cdot \text{length}^{-2}$  Выведено давление P с размерностью  
 $S := 10 \cdot \text{m}^2$  Задана площадь S  
 $F := 1000 \cdot \text{kg}$  Задана сила F  
 $P := \frac{F}{S}$  Задана формула для давления P  
 $P = 10 \cdot \text{mass} \cdot \text{length}^{-2}$  Выведено давление P с размерностью  
 $P(F) := \frac{F}{S}$  Задана функция пользователя  
 $F := 0 \cdot \text{kg}, 10 \cdot \text{kg} \dots 1000 \cdot \text{kg}$  Задано изменение F

10

P(F)

0

График  
P(F)

10000000

0 F

<- F в граммах

Док. 3.3. Операции над величинами с размерностью

Циклы, реализованные с помощью переменных с заданными пределами изменения

k := 0 ..5    l := 5 ..0    t := -2,-1.5 ..2    f := 2,1.5 ..-2

k
0
1
2
3
4
5

l
5
4
3
2
1
0

t
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2

f
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2

Циклы с целочисленным изменением переменной

Циклы с нецелочисленным изменением переменных

Пример реализации двойного цикла с вложением

i := 1 ..4                      Внешний цикл  
k := 1 ..3                      Внутренний цикл  
A<sub>i,k</sub> := i \* k                  Задание элементов матрицы

A =  $\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 \\ 0 & 2 & 4 & 6 \\ 0 & 3 & 6 & 9 \\ 0 & 4 & 8 & 12 \end{bmatrix}$                   Вывод матрицы

После объявления цикла используемые переменные должны быть индексированными ( скалярными ) переменными

i := 0 ..8

x := 2 \* i  
non-scalar value

Переменная x задана неверно- она не скаляр и будет не определена

y<sub>i</sub> := i<sup>2</sup>

Переменная y задана верно как скаляр

x = \*  
undefined

Переменная x не определена

y =  $\begin{bmatrix} 0 \\ 1 \\ 4 \\ 9 \\ 16 \\ 25 \\ 36 \\ 49 \\ 64 \end{bmatrix}$

Вывод скаляра y

Док. 3.4. Примеры использования переменных с заданными пределами изменения

Документ 3.4 поясняет использование переменных с заданными пределами изменения. Следует отметить, что описанные конструкции можно рассматривать как заголовки циклов. Все следующие за ними вычислительные блоки выполняются циклически, и, следовательно, входящие в них переменные становятся элементами массивов. На это расходуется память компьютера.

### 3.5. Массивы, матрицы, векторы и таблицы

Массив — это последовательность чисел. Система MathCAD оперирует двумя типами массивов. Первый — это одномерные массивы или векторы, второй — это двумерные массивы или матрицы. Матрицы можно рассматривать как  $n$  одномерных массивов, каждый из которых имеет  $m$  элементов. Векторы и матрицы также имеют имена, но в отличие от переменных характеризуют совокупность значений ряда их элементов.

Элементы векторов характеризуются порядковым номером или индексом. Обычно нумерация их идет с нуля. В общем случае она задается значением системной переменной ORIGIN. Элементы вектора имеют только один индекс. Например, если задан вектор  $V$ , то его элементами будут  $V_0, V_1, V_2$  и т. д., в общем виде  $V_i$ , где  $i$  — индекс.

Матрицы, в свою очередь, имеют элементы с двумя индексами, один из которых указывает на номер строки, а другой — на номер столбца. Например, если задана матрица  $M$ , то ее элементами будут  $M_{0,0}, M_{0,1}, M_{2,3}$  и т. д., в общем  $M_{i,j}$ .

Для задания вектора или матрицы следует установить курсор на место, где планируется задать вектор или матрицу, и нажать одновременно клавиши Alt и M. В ответ на запрос в командной строке о числе строк и столбцов матрицы следует ввести их нужные значения и нажать клавишу ENTER. В результате на экране дисплея появится шаблон вектора или матрицы в виде квадратных скобок с прямоугольниками — местами ввода значений элементов.

Присвоить тому или иному элементу заданное значение можно двумя способами. По первому нужно подвести курсор к прямоугольнику — шаблону этого элемента — и затем ввести на его место необходимое значение элемента. Для перемещения по шаблону вектора можно использовать наряду с клавишами перемещения курсора и клавишу табуляции Tab. Второй путь — применение операции присваивания. Например, если элементу вектора  $V_2$  нужно присвоить значение 123, то следует записать

$$V_2 := 123$$

При вводе такой записи следует учитывать, что подстрочный индекс вводится набором знака [ (открывающая квадратная скобка). Таким образом, для ввода указанной записи нужно нажимать следующие клавиши:

$$V[2:1\ 2\ 3 \text{ что даст на экране } V_2 := 123$$

Аналогично для матрицы нужно ввести значения двух индексов, причем их следует заключить в круглые скобки и разделить запятой:

$$M[(1, 3):4\ 5\ 6 \text{ что даст на экране } M_{1,3} := 456$$

В дальнейшем мы рассмотрим ряд операций с векторами и матрицами. В док. 3.5 показаны приемы работы с векторами. Вначале задан вектор  $x$ , содержащий 10 элементов. Далее выражение  $y := 2 \cdot x$  создает новый вектор  $y$ , у которого каждый элемент равен удвоенному элементу вектора  $x$  с тем же индексом. Затем задан вывод вектора  $y$ . В конце документа формируется еще один вектор  $y$ , причем

$$y_k := \exp(x_k),$$

и выводится таблица значений  $y_k$ . После этого описания будут понятны примеры простых действий с матрицами, приведенные в док. 3.6.

Из этих примеров видно также, что система MathCAD выводит значения элементов векторов и матриц в виде таблиц. В стандартной конфигурации персональных компьютеров IBM PC XT с емкостью ОЗУ 640 Кбайт суммарное число элементов векторов и матриц может достигать 8000. Такое число элементов, к примеру, имеет квадратная матрица с 88 строками и 88 столбцами. Предельное число столбцов при вводе матриц в шаблон на экране дисплея равно 100.

Для машин с меньшей емкостью ОЗУ общее число элементов будет меньше. Однако многие компьютеры сейчас имеют расширенную по крайней мере до 1 Мбайта память ОЗУ. MathCAD использует расширение памяти под размещение векто-

Вектор в системе MathCAD – одномерный массив  
числовых данных, объединенных именем и  
представляемых одной строкой или столбцом

Для задания вектора нажмите клавиши Alt и M

В командной строке появится надпись

Array size ( rows columns ) Nстрок Nстолбцов  
Размер массива ( строк столбцов )

Задайте нужное число либо строк, либо столбцов

Если Nстрок=1, вектор будет одной строкой

$V := (1 \ 2 \ 3 \ 4)$        $V = (1 \ 2 \ 3 \ 4)$

Если Nстолбцов=1 или опущено, вектор будет одним столбцом

$x := \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$       Задание вектора x

$y := 2 \ x$       Задание вектора  $y=2*x$

$y = \begin{bmatrix} 0 \\ 2 \\ 4 \\ 6 \\ 8 \end{bmatrix}$       Вывод вектора y

$k := 0 \dots 4$       Поэлементное создание вектора y  
(k – индекс элемента)

$y_k := \exp[x_k]$

$y = \begin{bmatrix} 1 \\ 2.718 \\ 7.389 \\ 20.086 \\ 54.598 \end{bmatrix}$       Вывод вектора y

$y_3 = 20.086$       Вывод третьего элемента вектора  
(индексация начинается с нуля)

ORIGIN = 0      Эта переменная указывает на число,  
с которого начинается индексация

$V0 := \begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix}$       Если элементы вектора не определены,  
появится сообщение об ошибке

missing operand

$V0 = \cdot$       Вывод неопределенного вектора  
также приводит к сообщению об ошибке

undefined

### Док. 3.5. Примеры работы с векторами

ров, массивов и данных для построения графиков. Это позволяет существенно увеличить размер документов, готовящихся и исполняющихся в среде системы MathCAD.

Матрица в системе MathCAD-это двумерный числовой массив с определенным именем. Задание матрицы аналогично заданию вектора, но надо указывать и Nстрок, и Nстолбцов. Нумерация как строк, так и столбцов матрицы начинается со значения, указанного системной переменной ORIGIN. При задании матрицы достаточно задать один ее элемент

$M_{2,2} := 1$       Задание впервые  $M_{2,2} = 1$   
задает матрицу, у которой  
остальные элементы равны 0

$$M = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Можно задать матрицу заполнением шаблонов всех элементов

$M1 := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$       Задание матрицы M1

$MT := M1^T$       Транспонирование матрицы M1

$MT = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$       Вывод транспонированной матрицы

$M3 := \begin{bmatrix} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix}$       Если элементы матрицы не заданы,  
это ведет к сообщению об ошибке

**missing operand**

$M3 = \cdot \cdot$       Вывод неопределенной матрицы также '  
**undefined**      приводит к сообщению об ошибке

Решение системы линейных уравнений  $A \cdot X = B$   
и инвертирование матрицы коэффициентов A:

$A := \begin{bmatrix} 4 & .24 & -.08 \\ .09 & 3 & -.15 \\ .04 & -.08 & 4 \end{bmatrix}$        $B := \begin{bmatrix} 8 \\ 9 \\ 20 \end{bmatrix}$       Задание матрицы A  
и вектора свободных  
членов системы B

$A^{-1} = \begin{bmatrix} -0.25 & -0.02 & 0.004 \\ -0.008 & 0.334 & 0.012 \\ -0.003 & 0.007 & 0.25 \end{bmatrix}$       Инвертирование  
матрицы A

$X := A^{-1} \cdot B$       Решение системы линейных уравнений

$X = \begin{bmatrix} 1.909 \\ 3.195 \\ 5.045 \end{bmatrix}$       Результат решения

Док. 3.6. Примеры работы с матрицами

Запись и считывание файловой переменной

$X := 0 \dots 5$

Задание переменной-вектора Y

$Y_X := X^2$

$WRITE(Vfile) := Y_X$

Запись вектора Y в виде файла Vfile

$Z_X := READ(Vfile)$

Считывание файла Vfile с диска и присвоение значений вектору Z

$Z = \begin{bmatrix} 0 \\ 1 \\ 4 \\ 9 \\ 16 \\ 25 \end{bmatrix}$

Вывод вектора Z, аналогичного вектору Y

Запись и считывание матричных файлов

$M := \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

Задана матрица M

$WRITEPRN(DMAT) := M$

Матрица M записана в файл DMAT

$M := \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

Матрица M обнулена

$M = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

Проверено обнуление матрицы M

$M := READPRN(DMAT)$

Матрица M считана из файла DMAT

$M = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

Вывод значений матрицы M подтверждает ее восстановление

$i := \sqrt{-1}$

Задание мнимой единицы

$N := \begin{bmatrix} 1 + 2i & 3 + 4i \\ 5 + 6i & 7 + 8i \end{bmatrix}$

Задание матрицы N с комплексными элементами

$WRITEPRN(DCMAT) := N$

Запись матрицы N в файл DCMAT

$P := READPRN(DCMAT)$

Считывание матрицы из файла и создание новой матрицы P

$P = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}$

Вывод матрицы P

Док. 3.7. Действия с файлами данных

### 3.6. Файлы данных

Одним из достоинств системы MathCAD является возможность ее взаимодействия с другими программами и системами. Например, в MathCAD можно вводить данные эксперимента, получаемые с контрольно-измерительных систем, которые затем подвергать необходимой математической обработке.

Для общения с «внешним миром» система MathCAD использует файлы — совокупность данных, объединенных под названием — именем файла. MathCAD использует последовательные файлы, в которых данные в виде чисел записываются на диск или считываются с диска строго в последовательном порядке. Файлы хранят также информацию о типе записанных данных — значениях переменных, векторов или матриц.

Имеется шесть команд для работы с файлами данных.

READ (Имя) — считывает данные из файла с указанным именем и как функция возвращает скаляр.

WRITE (Имя) — создает на диске файл данных с указанным именем. Может использоваться в виде

WRITE (Имя):= Арифметическое выражение.

APPEND (Имя) — записывает значение в уже существующий файл данных. Может использоваться в виде

APPEND (Имя файла):= Арифметическое выражение.

READPRN (Имя) — считывает данные матрицы и как функция возвращает матрицу.

WRITEPRN (Имя) — записывает значения элементов матрицы в файл с указанным именем. Каждая строка в файле соответствует строке в матрице. Может использоваться в виде

WRITEPRN (Имя):= Матричное выражение.

APPENDPRN (Имя) — записывает матрицу в уже существующий файл с указанным именем. Может применяться в виде

APPENDPRN (Имя переменной):= Матричное выражение.

Из этих определений вытекает, что система MathCAD имеет файловые переменные, которые можно записывать в виде файлов на диск и считывать их с диска. При этом имя файла может быть своим или совпадать с именем переменной, а файл имеет расширение .DAT (от слова data — данные). Файлы для матричных переменных имеют расширение.

Файлы данных имеют очень простую структуру и представляют собой запись в текстовом формате (формате ASCII) наборов числовых данных. Это позволяет легко использовать эти файлы в программах, написанных на обычных языках программирования, например Бейсик, либо готовить их в таких программах для передачи в систему MathCAD.

Для присвоения файлам данных имени служит команда filename. В командном режиме она вводится следующим образом:

[ Esc ] filename Имя ENTER

Документ 3.7 демонстрирует различные приемы работы с файлами. В этих примерах используются запись файлов на диск и их считывание с диска. Последний пример иллюстрирует запись матрицы с комплексными коэффициентами. Считанная матрица имеет все действительные коэффициенты, но с помощью матричных операций несложно восстановить исходную матрицу. Необходимую для математической обработки матрицу можно подготовить с помощью внешних программ, записать в виде файла и затем уже передать в систему MathCAD.

### 3.7. Графические данные

Результаты расчетов система MathCAD может представлять в виде графических данных, включаемых в текст документов. Как отмечалось, для задания шаблона графика использует команда, вводимая набором символа @. На рис. 1.4, док. 1.5 и 1.6 были показаны простейшие приемы вывода графических данных. Здесь мы

обсудим все возможности графического процессора системы MathCAD.

Прежде всего отметим, что вид графика определяется его форматом, т. е. совокупностью указаний на те или иные параметры графика. Формат графика запоминается, и при записи документа на диск он предшествует записи команд построения графика. Поэтому при исполнении любого документа графики строятся в том виде, который был получен при последнем исполнении документа.

Для задания формата графиков используется команда `plot format`, вводимая в командном режиме, т. е. после нажатия клавиши `Esc`. В этом случае формат задается глобально, т. е. по всему документу. Однако можно задать формат и локально, только для данного графика. Для этого курсор следует ввести внутрь шаблона графика и нажать клавишу `F`.

И в том и в другом случае в командной строке появится приглашение к редактированию формата графиков в виде

```
logs=0,0 subdivs=1,1 size=5,15 type=1
```

Здесь указаны те параметры формата, которые система задает по умолчанию или которые были введены глобально и записаны в конфигурационный файл. С помощью обычных приемов редактирования можно изменить параметры формата на нужные.

Рассмотрим, что дает возможность изменения формата.

`logs=Y, X` — задает параметры графиков с логарифмическими осями  $X$  и  $Y$  координатной сетки. Если  $X$  и  $Y$  заданы нулевыми, то используется линейный масштаб графика по соответствующей оси. При  $X=1$  или  $Y=1$  используется логарифмический масштаб без промежуточных делений. Если  $X$  или  $Y$  заданы как целые числа  $N$ , то задается  $N$  циклов логарифмического изменения величин по этим осям и строится соответствующее число делений по осям.

`subdivs=Y, X` — задает число делений масштабной сетки. Для графиков с линейным масштабом значения  $X$  или  $Y$ , равные 1, означают, что промежуточных делений блок графики не имеет, т. е. масштабная сетка вырождается в прямоугольник. Если  $X$  или  $Y$  — целое число  $N$ , то диапазон величин по осям делится на  $N$  поддиапазонов и выводятся соответствующие дополнительные линии масштабной сетки. Для логарифмического масштаба число делений сетки должно согласовываться с числом циклов. Допустимо указывать значения  $N=1, 2$  и  $9$ .

`size=Y, X` — задает размеры графика в знаках по осям  $X$  и  $Y$ . Полный экран имеет размеры  $Y=25$  и  $X=80$ . Однако можно задавать графики как меньших, так и больших размеров (до  $X$  или  $Y$ , равных 127).

`type=...` — задает тип линий, которыми строится график. После слова `type=` может использоваться один из следующих параметров:

`I` или `L` — задает непрерывную линию с линейной интерполяцией между узловыми точками графика,

`d` — строит точки в узлах, не соединяя их линиями,

`s` — соединяет узловые точки ступенчатой линией,

`e` — указывает диапазон возможных разбросов величин в соответствующей точке,

`b` — строит прямоугольники, вершины которых расположены на узловых точках,

`x` — строит знаки вида  $x$  в каждой узловой точке, сами точки друг с другом не соединяются,

`X` — строит знаки вида  $x$  в каждой узловой точке и соединяет их отрезками прямых,

`p` — помечает каждую узловую точку знаком  $+$ , сами точки не соединяются,

`P` — помечает каждую узловую точку знаком  $+$  и соединяет их друг с другом отрезками прямых,

`o` — помечает каждую узловую точку прямоугольничком, сами точки друг с другом не соединяются,

`O` — помечает каждую узловую точку прямоугольничком, соединяет точки отрезками прямых,

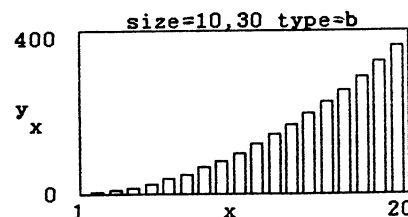
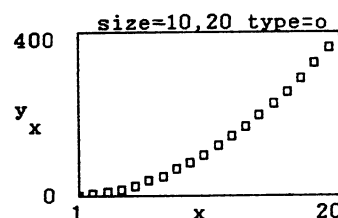
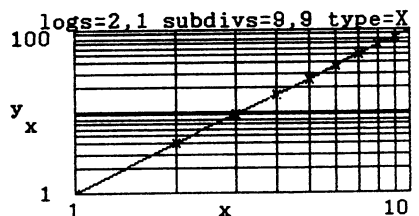
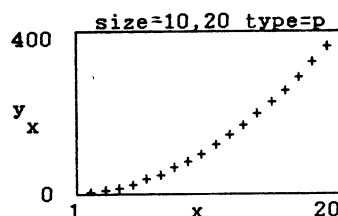
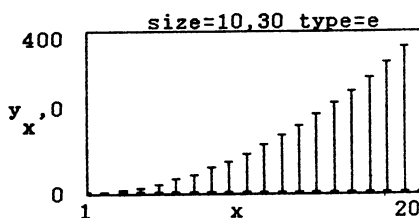
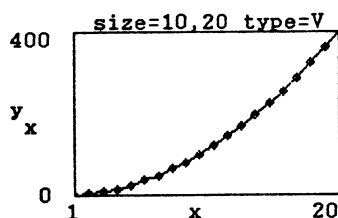
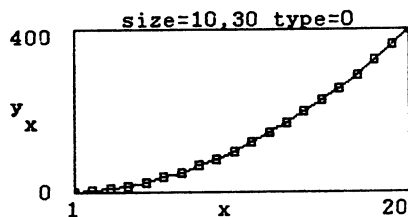
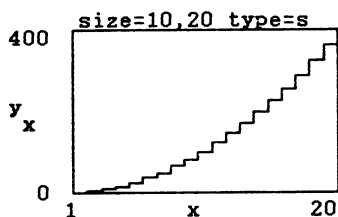
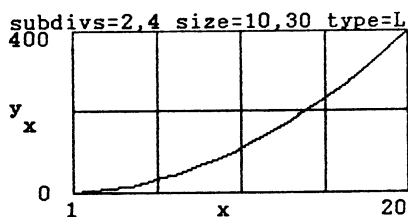
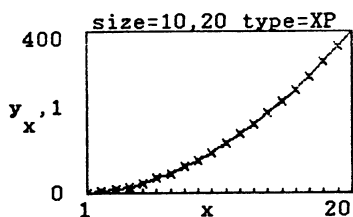
`v` — помечает каждую узловую точку ромбиком, сами точки друг с другом не соединяются,



`x := 1 ..20`

`y_x := x2`

`<-- Парабола`



Док. 3.8. Построение графиков с различными параметрами формата

V — помечает каждую узловую точку ромбиком, соединяет точки друг с другом отрезками прямых.

С помощью параметров команды `format` можно строить разнообразные графики, удовлетворяющие самым изысканным вкусам. Более того, эти команды могут комбинироваться. Например, команда `On` будет строить одну кривую непрерывной линией с прямоугольниками в узловых точках, а другую кривую — с ромбиками в узловых точках.

Документ 3.8 демонстрирует построение графиков различного типа с указанием параметров команды `format` для них. Он наглядно показывает большие выразительные возможности графики и легкость средств их задания.

Многие графические программы не справляются с построением графиков функций с особенностями. Например, они не строят график функции

$$f(x) = \sin(x)/x,$$

поскольку при  $x=0$  деление  $\sin(x)$  на  $x$  воспринимается как ошибка. Между тем неопределенность  $0/0$  в данном случае дает конечное значение 1. Как видно из док. 1.1, графический процессор системы MathCAD легко справляется с построением графика такой функции. Другая функция с особенностями —  $\tan(x)$  периодически устремляется в бесконечность. В итоге график может иметь масштаб, при котором вид кривых будет неприемлемым. В таком случае график функции может быть построен при ручном задании масштаба по оси Y.

Множество различных графиков описывается в последующих разделах книги. Отметим, что данные о формате графиков можно взять из просмотра файлов с расширением .MCD. Для этого следует воспользоваться текстовым редактором. Кроме того, данные о формате графиков, включенных в загружаемые с диска примеры, всегда можно установить, подведя курсор в блок графики и нажав клавишу F. В командной строке появится надпись о формате графика.

## 4. ОПЕРАТОРЫ И ФУНКЦИИ СИСТЕМЫ MathCAD

### 4.1. Арифметические операторы

Операторы в системе MathCAD являются выраженными в виде специальных знаков командами для выполнения той или иной математической операции. Операторы входят в состав математических выражений. Характерной особенностью системы MathCAD является близость вида операторов к виду специальных математических знаков, описывающих соответствующие операции.

Поскольку в наборе знаков персональных компьютеров нет многих математических знаков, в системе предусмотрено формирование таких знаков любых размеров в графическом режиме работы. Это и придает документам системы MathCAD поразительную схожесть с текстами обычной научно-технической литературы.

Документ 4.1 содержит данные об основных арифметических операторах, используемых для выполнения типовых арифметических операций, а также о схемах их ввода. Следует заметить, что такие операторы могут вводиться и без параметров. В этом случае на месте каждого параметра появляется его шаблон в виде маленького прямоугольника. На его место затем подставляется значение параметра.

Помимо общепринятых операторов математических операций, имеющихся в большинстве языков программирования, система MathCAD располагает несколькими новыми мощными операторами суммирования членов ряда  $\Sigma$ , перемножения членов ряда  $\Pi$ , вычисления производной  $d/dt$  и численного интегрирования  $\int$ . С их помощью можно решать сложные задачи с элементами расчетов, относящимися к курсу высшей математики. В то же время применение этих операторов столь же просто, как и операторов общеизвестных арифметических операций.

В § 4.10 описываются численные методы, применяемые при реализации операторов вычисления производной и определенного интеграла. Здесь лишь отметим, что эти операторы выполняют соответствующие операции с заданной погрешностью.

Название	Ввод	Пример
Скобки (X)	или (X)	$2 \cdot (1 + 3) = 8$
Степень	$X^Y$	$3^2 = 9$
Факториал	$X!$	$4! = 24$
Смена знака	$-X$	$-5 = -5$
Квадратный корень	$\sqrt{X}$	$\sqrt{9} = 3$
Абсолютное значение	$ X $	$ -5  = 5$
Сложение	$X + Y$	$2 + 3 = 5$
Вычитание	$X - Y$	$8 - 5 = 3$
Умножение	$X * Y$	$2 \cdot 3 = 6$
Деление	$X / Y$	$\frac{12}{3} = 4$
Вывод	$X =$	$15 = 15$
Присваивание	$X : Y$	$X := 8 \quad X = 8$
Изменение в отрезке	$Ns ; Nf$ $Xs , Xl ; Xf$	$N := 0, 0.5 \dots 4$ $X := 0, 0.5 \dots 4$
<p>Здесь Ns-стартовое значение N  Nf-конечное значение N, шаг 1  Xs-стартовое значение X  Xl-следующее значение X  Xf-конечное значение X  Шаг dX=Xl-Xs</p>		
Суммирование членов ряда	$i \notin X$	$i := 1 \dots 5 \quad X_i := 2 \cdot i$ $\sum_i X_i = 30$
Перемножение членов ряда	$i \# X$	$\prod_i X_i = 3.84 \cdot 10^3$
Определенный интеграл $x \& f(x)$		$\int_0^1 \sqrt{2x} dx = 0.943$
Производная	$x ? f(x)$	$x := 1$ $\frac{d}{dx} \sin(x) = 0.54$ $\cos(1) = 0.54$
Неравенство	$X > Y$	$\frac{5}{4} > \frac{4}{5} = \frac{1}{0}$

Док. 4.1. Арифметические операции и примеры их применения

Вид функции	Имя	Пример
Экспоненциальная функция	<code>exp(Z)</code>	<code>exp(2) = 7.389</code>
Натуральный логарифм	<code>ln(Z)</code>	<code>ln(2) = 0.693</code>
Десятичный логарифм	<code>log(Z)</code>	<code>log(2) = 0.301</code>
Синус ( угол в радианах )	<code>sin(Z)</code>	<code>sin(1) = 0.841</code>
Косинус	<code>cos(Z)</code>	<code>cos(1) = 0.54</code>
Тангенс	<code>tan(Z)</code>	<code>tan(1) = 1.557</code>
Угол радиуса-вектора	<code>angle(x,y)</code>	<code>angle(1,1) = 0.78</code>
Арксинус	<code>asin(Z)</code>	<code>asin(1) = 1.571</code>
Арккосинус	<code>acos(Z)</code>	<code>acos(1) = 0</code>
Арктангенс	<code>atan(Z)</code>	<code>atan(1) = 0.785</code>
Синус гиперболический	<code>sinh(Z)</code>	<code>sinh(1) = 1.175</code>
Косинус гиперболический	<code>cosh(Z)</code>	<code>cosh(1) = 1.543</code>
Тангенс гиперболический	<code>tanh(Z)</code>	<code>tanh(1) = 0.762</code>
Арксинус гиперболический	<code>asinh(Z)</code>	<code>asinh(1) = 0.881</code>
Арккосинус гиперболический	<code>acosh(Z)</code>	<code>acosh(2) = 1.317</code>
Арктангенс гиперболический	<code>atanh(Z)</code>	<code>atanh(.2) = 0.203</code>
функция Бесселя	<code>J0(x)</code>	<code>J0(1) = 0.765</code>
функция Бесселя	<code>J1(x)</code>	<code>J1(1) = 0.44</code>
функция Бесселя	<code>Jn(n,x)</code>	<code>Jn(1,1) = 0.44</code>
функция Бесселя	<code>Y0(x)</code>	<code>Y0(1) = 0.088</code>
функция Бесселя	<code>Y1(x)</code>	<code>Y1(1) = -0.781</code>
функция Бесселя	<code>Yn(n,x)</code>	<code>Yn(1,1) = -0.781</code>
Гамма-функция	<code>Γ(x)</code>	<code>Γ(.5) = 1.772</code>
функция вероятности ошибок	<code>erf(x)</code>	<code>erf(1) = 0.843</code>
Задание комплексного числа	<code>Z := 2 + 3i</code>	
Действительная часть Z	<code>Re(Z)</code>	<code>Re(Z) = 2</code>
Мнимая часть Z	<code>Im(Z)</code>	<code>Im(Z) = 3</code>
Аргумент Z	<code>arg(Z)</code>	<code>arg(Z) = 0.983</code>
<code>X := 0    Y := 0</code>		
Задание функции пользователя	<code>mod(X,Y) := <math>\sqrt{X^2 + Y^2}</math></code>	
и ее применение	<code>mod(2,3) = 3.606</code>	
<code>X = 0    Y = 0</code>		

Док. 4.2. Встроенные функции общего назначения и пример задания функции пользователя

Ее значение задается системной переменной TOL и по умолчанию равно 0.001. Если требуется иная погрешность, то перед применением оператора нужно присвоить системной переменной это значение погрешности (например,  $TOL:=0.00001$ ).

Из док. 4.1 видно также, что MathCAD имеет операторы сравнения в виде неравенств: больше  $>$ , меньше  $<$ , больше или равно  $\geq$ , меньше или равно  $\leq$  и не равно  $\neq$ . Фактически эти операторы являются функциями, вырабатывающими значение 1, если условие выполняется, и 0, если оно не выполняется. Пример этот дан в конце док. 4.1.

## 4.2. Встроенные функции и функции пользователя

Система MathCAD содержит множество встроенных функций. Как отмечалось, особенностью функции является возврат ее значения в ответ на обращение к функции по ее имени с указанием аргументов функции в круглых скобках.

Элементарные функции в системе MathCAD можно подразделить на тригонометрические, гиперболические и алгебраические. Все они, кроме функции  $\text{angle}(x, z)$ , вычисляющей угол (в радианах) радиуса-вектора точки  $(x, y)$ , могут иметь аргумент в виде как действительных чисел  $x$ , так и комплексных  $Z$ . Назначение таких функций, вид обращения к ним и примеры применения даны в док. 4.2. Там же приведены и данные о вычислении ряда специальных математических функций — Бесселя, гамма-функции и функции ошибок.

К широко распространенным функциям относятся также функции выделения действительной  $\text{Re}(Z)$  и мнимой  $\text{Im}(Z)$  частей комплексного числа, а также функция вычисления его аргумента  $\text{arg}(Z)$ . Аргументом является угол  $\theta$  в выражении  $r \cdot e^{i \cdot \theta}$ , лежащий в пределах  $[-\pi; +\pi]$ .

Несмотря на обилие встроенные в систему MathCAD функций, многие из которых будут описаны далее, всегда может возникнуть необходимость в новых функциях. Такая возможность реализована с помощью следующей конструкции:

Имя (Список аргументов) := Выражение

Функции пользователя могут иметь один аргумент или несколько аргументов, образующих список. Аргументы в списке разделяются запятыми. В выражение могут входить как переменные из списка аргументов, так и переменные, используемые в других блоках документов. Переменные, входящие в список аргументов, являются локальными, т. е. их действие проявляется только в пределах блока функции. Функция пользователя воспринимает значения других переменных, которые были им присвоены до ее определения.

Функция пользователя, будучи раз определенной в программе, может использоваться точно так же, как и встроенная функция, т. е. путем обращения к ней по имени с указанием в скобках списка параметров. Разумеется, использованию такой функции должно предшествовать ее задание. Пример задания функции пользователя дан в док. 4.2. Множество таких примеров будет приведено и в дальнейшем. Применение функций пользователя придает системе MathCAD свойство расширяемости и облегчает ее подстройку под конкретные задачи пользователя.

## 4.3. Векторные и матричные операторы и функции

Благодаря наличию в системе MathCAD операторов задания векторов и матриц операции над ними выполняются столь же просто, как и над обычными переменными. Это резко облегчает решение задач линейной алгебры. Практически все операции могут проводиться над векторами и матрицами как с действительными, так и комплексными элементами.

Ниже описаны основные векторные и матричные операции. Векторы обозначены буквой  $V$ , матрицы — буквой  $M$ , скалярные величины — буквой  $Z$ . Операции указаны вместе с операндами.

Операция	Ввод	Назначение операции
$Z \cdot V \quad V \cdot Z$	$Z * V \quad V * Z$	Умножение вектора на скаляр
$Z \cdot M \quad M \cdot Z$	$Z * M \quad M * Z$	Умножение матрицы на скаляр
$V1 \cdot V2$	$V1 * V2$	Умножение двух векторов
$M \cdot V$	$M * V$	Умножение матрицы на вектор
$M1 \cdot M2$	$M1 * M2$	Умножение двух матриц
$V/Z$	$V/Z$	Деление вектора на скаляр
$M/Z$	$M/Z$	Деление матрицы на скаляр
$V1 + V2$	$V1 + V2$	Сложение векторов
$M1 + M2$	$M1 + M2$	Сложение матриц
$V + Z \quad Z + V$	$V + Z \quad Z + V$	Сложение вектора со скаляром
$V1 - V2$	$V1 - V2$	Вычитание векторов
$M1 - M2$	$M1 - M2$	Вычитание матриц
$V - Z$	$V - Z$	Вычитание из вектора скаляра
$-V$	$-V$	Смена знака у всех элементов вектора
$-M$	$-M$	Смена знака у всех элементов матрицы
$M^n$	$M^n$	Возведение матрицы в степень $n$ путем $n$ -кратного перемножения (при $n = -1$ — инвертирование)
$ V $	$ V $	Вычисление квадратного корня из $\mu V$
$ M $	$ M $	Вычисление определителя матрицы
$V^T$	$V [\text{Alt}] !$	Транспонирование вектора
$M^T$	$M [\text{Alt}] !$	Транспонирование матрицы
$V1 \times V2$	$V1 [\text{Alt}] * V2$	Кроссумножение двух трехэлементных векторов
$\bar{V}$	$V''$	Получение комплексно-сопряженного вектора
$\bar{M}$	$M''$	Получение комплексно-сопряженной матрицы
$\Sigma V$	$[\text{Alt}] \$ V$	Сумма всех элементов вектора
$\vec{V}$	$V [\text{Alt}]$	Векторизация вектора
$\vec{M}$	$M [\text{Alt}]$	Векторизация матрицы
$M^{<n>}$	$M [\text{Alt}] ^n$	Выделение $n$ -го столбца матрицы
$V_n$	$V [n]$	Выделение $n$ -го элемента вектора
$M_{n1, n2}$	$M [ (n1, n2) ]$	Выделение элемента $M_{n1, n2}$ матрицы

Векторизацией вектора или матрицы называется выполнение каких-либо операций (например, возведение в степень) одновременно над всеми элементами их массива. Это упрощает векторные и матричные операции. Указание под знаком векторизации самого вектора или матрицы никаких действий над ними не производит. Так, если  $V$  — вектор, то  $V \vec{\times} 2$  создаст вектор, все элементы которого удвоены, а  $\bar{V}$  оставит вектор без изменения.

Документ 4.3 иллюстрирует основные операции с векторами. Перечисленные выше и приведенные в док. 4.3 операции являются общими. Множество других специальных операций с векторами, таких как статистика вектора, регрессия, интерполяция, быстрое преобразование Фурье и др., будет описано далее. Документ 4.4 демонстрирует проведение матричных операций.

Система MathCAD имеет также ряд векторных и матричных функций. Из векторных функций общего назначения следует отметить следующие:

$\text{length}(V)$  — возвращает длину вектора, т. е. число его элементов,  
 $\text{last}(V)$  — возвращает индекс последнего элемента вектора,  
 $\text{max}(V)$  — возвращает значение элемента, имеющего максимальное значение,  
 $\text{min}(V)$  — возвращает значение элемента, имеющего минимальное значение,  
 $\text{Re}(V)$  — возвращает вектор действительных частей элементов комплексного вектора  $V$ ,  
 $\text{Im}(V)$  — возвращает вектор мнимых частей элементов комплексного вектора  $V$ .

Умножение вектора на скаляр

$$V := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad U := V \cdot 3 \quad U = \begin{bmatrix} 3 \\ 6 \\ 9 \end{bmatrix}$$

Сложение и вычитание векторов

$$V1 := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad V2 := \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix} \quad V3 := \begin{bmatrix} 8 \\ 7 \\ 8 \end{bmatrix}$$

$$V1 + V2 - V3 = \begin{bmatrix} -2 \\ -1 \\ 0 \end{bmatrix}$$

Умножение двух векторов

$$V3 := V1 \cdot V2 \quad V3 = 26$$

Кросс - умножение двух трехэлементных векторов

$$V3 := V1 \times V2 \quad V3 = \begin{bmatrix} -2 \\ 4 \\ -2 \end{bmatrix}$$

Суммирование всех элементов вектора

$$V := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad VSUM := \phi V \quad VSUM = 6$$

Транспонирование вектора

$$V := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad U := V^T \quad U = (1 \quad 2 \quad 3)$$

Вычисление нормы вектора

$$V := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad |V| = 3.742$$

Смена знака у всех элементов вектора

$$V := -V \quad V = \begin{bmatrix} -1 \\ -2 \\ -3 \end{bmatrix}$$

Векторизация

$$V := \overrightarrow{V1} \quad V = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

Выделение элемента вектора

$$X := V_1 \quad X = 2$$

Векторные функции

$$\text{length}(V) = 3 \quad \text{last}(V) = 2$$

$$\text{max}(V) = 3 \quad \text{min}(V) = 1$$

#### Док. 4.3. Основные операции с векторами

Следующие матричные функции также относятся к функциям общего применения:

rows (M) — возвращает число строк матрицы M,  
cols (M) — возвращает число столбцов матрицы M,  
tr (M) — возвращает след квадратной матрицы M, т. е. сумму ее диагональных элементов,

Транспонирование матрицы

$$A := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \quad B := A^T \quad B = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

Матричные функции

$$\text{rows}(A) = 2 \quad \text{cols}(A) = 3$$

$$M := \text{identity}(3)$$

$$\text{tr}(M) = 3 \quad M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Двойное инвертирование матрицы дает ту же матрицу

$$A := \begin{bmatrix} 7 & 8 \\ 4 & 5 \end{bmatrix} \quad B := A^{-1} \quad C := B^{-1}$$

$$B = \begin{bmatrix} -1.667 & -2.667 \\ -1.333 & 2.333 \end{bmatrix} \quad C = \begin{bmatrix} 7 & 8 \\ 4 & 5 \end{bmatrix}$$

Умножение матриц

$$A := \begin{bmatrix} 3 & 2 \\ 3 & 4 \end{bmatrix} \quad B := \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} \quad A \cdot B = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$$

Вычисление детерминанта матрицы

$$A := \begin{bmatrix} 2 & 5 & -3 \\ 1 & 4 & -1 \\ 1 & 3 & 2 \end{bmatrix} \quad D := |A| \quad D = 10$$

Выделение 2-го столбца матрицы

$$A := \begin{bmatrix} 2 & 5 & -3 \\ 1 & 4 & -1 \\ 1 & 3 & 2 \end{bmatrix} \quad V := A^{<1>} \quad V = \begin{bmatrix} 5 \\ 4 \\ 3 \end{bmatrix}$$

Выделение заданного элемента матрицы

$$A := \begin{bmatrix} 2 & 5 & -3 \\ 1 & 4 & -1 \\ 1 & 3 & 2 \end{bmatrix} \quad A_{1,1} = 2 \quad A_{0,0} = 2 \quad A_{2,1} = 3$$

Инвертирование знака у всех элементов матрицы

$$A := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \quad -A = \begin{bmatrix} -1 & -2 & -3 \\ -4 & -5 & -6 \end{bmatrix}$$

Умножение, деление всех элементов матрицы на скаляр

$$2A = \begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \end{bmatrix} \quad \frac{A}{2} = \begin{bmatrix} 0.5 & 2.5 & 1.5 \\ 2 & 2.5 & 3 \end{bmatrix}$$

Сложение матриц

$$A := \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad B := \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} \quad A + B = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

Векторизация матрицы

$$M := \vec{A} \quad M = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

Создание комплексно-сопряженной матрицы

$$i := \sqrt{-1} \quad A := \begin{bmatrix} 1 + 2i & 4 - 5i \\ 0 & 0 \end{bmatrix}$$

$$B := \bar{A} \quad B = \begin{bmatrix} 1 - 2i & 4 + 5i \\ 0 & 0 \end{bmatrix}$$

Док. 4.4. Основные операции с матрицами



$M := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$	Задание матрицы M 2*3
$rows(M) = 2$	Число строк матрицы
$cols(M) = 3$	Число столбцов матрицы
$ORIGIN := 1$	Нумерация индексов задана с 1
$M1 := identity(3)$	Задание единичной квадратной матрицы
$M1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	Вывод единичной матрицы
$V := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$	Задание вектора
$augment(M1, V) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{bmatrix}$	Комбинация матрицы M1 с вектором V
$M2 := \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$	Задание матрицы M2
$M := augment(M1, M2)$	Комбинация матриц M1 и M2
$M = \begin{bmatrix} 1 & 0 & 0 & 1 & 4 \\ 0 & 1 & 0 & 2 & 5 \\ 0 & 0 & 1 & 3 & 6 \end{bmatrix}$	Вывод матрицы, полученной в результате комбинирования
$tr \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} = 15$	Вычисление следа квадратной матрицы -суммы ее диагональных элементов
$i := \sqrt{-1}$	Задание мнимой единицы -переход к комплексным операциям
$COMPLM := \begin{bmatrix} 1 + 2i & 2 + 3i & 4 + 4i \\ 5 + 6i & 6 + 7i & 7 + 8i \\ 8 + 9i & 9 + 10i & 11 \end{bmatrix}$	Задание матрицы с комплексными элементами
$R := Re(COMPLM)$	Создание матрицы R действительных частей
$J := Im(COMPLM)$	Создание матрицы J мнимых частей
$R = \begin{bmatrix} 1 & 2 & 4 \\ 5 & 6 & 7 \\ 8 & 9 & 11 \end{bmatrix}$	Вывод матрицы R
$J = \begin{bmatrix} 2 & 3 & 4 \\ 6 & 7 & 8 \\ 9 & 10 & 0 \end{bmatrix}$	Вывод матрицы J
$K := 1$ $L := i \dots 3$	Синтез комплексной матрицы MC=COMPLM по матрицам R и J
$MC_{K,L} := R_{K,L} + J_{K,L} \cdot i$	
$MC = \begin{bmatrix} 1 + 2i & 2 + 3i & 4 + 4i \\ 5 + 6i & 6 + 7i & 7 + 8i \\ 8 + 9i & 9 + 10i & 11 \end{bmatrix}$	Вывод матрицы MC

Док. 4.5. Примеры применения основных матричных функций

identity — возвращает единичную матрицу, у которой диагональные элементы равны 1, а остальные 0,

Re (M) — возвращает матрицу действительных частей элементов комплексной матрицы M,

Im (M) — возвращает матрицу мнимых частей элементов комплексной матрицы M,

augment (MV1, MV2) — комбинирует в одну матрицу или один вектор два компонента (в виде матриц или векторов),

$\delta$  (n1, n2) — возвращает дельта-функцию Кронекера (n1 и n2 — целые числа, результат равен 1, если n1 = n2, иначе он равен 0),

$\epsilon$  (n1, n2, n3) — возвращает значения  $-1$ ,  $0$  или  $+1$ , характеризующие полный антисимметричный тензор ранга 3 (n1, n2 и n3 — целые числа от 0 до 2 или от ORIGIN до ORIGIN+2, если ORIGIN не равна нулю).

Две последние функции в [9] указаны в разделе матричных функций. Однако, строго говоря, они имеют более широкое применение. Так, дельта-функцию легко использовать для моделирования импульсных сигналов.

Примеры применения ряда из этих функций даны в док. 4.5. В целом можно отметить, что набор векторных и матричных операторов и функций достаточен для решения практически любых задач линейной алгебры, векторного и матричного анализа. Важно отметить, что почти все операции могут выполняться над векторами и матрицами как с действительными, так и комплексными коэффициентами.

#### 4.4. Функции линейной и сплайн-интерполяции

В прикладных применениях математики широко используются зависимости вида  $y(x)$ , обычно заданные рядом значений  $x$  и  $y$  их узловых точек. Однако, когда этих точек мало, такая зависимость оказывается не информативной и не наглядной.

Система MathCAD имеет средства для получения промежуточных точек зависимостей  $y(x)$  путем их интерполяции. Возможны два основных вида такой интерполяции — линейная и сплайн-интерполяция. Исходные данные для интерполяции образуют два вектора  $X$  и  $Y$ , элементы которых являются координатами узловых точек.

При линейной интерполяции узловые точки соединяются отрезками прямых. Если  $x$  выходит за пределы конечных точек, то осуществляется линейная экстраполяция по отрезкам прямых, примыкающим к конечным точкам. При линейной интерполяции интерполирующая функция имеет разрывы первой и второй производных. Поэтому линейная интерполяция удовлетворительна лишь при достаточно гладких функциях и большом числе узлов интерполяции.

При сплайн-интерполяции зависимость  $y(x)$  заменяется кусками полиномов третьей степени. Каждый полином проходит точно через три ближайшие узловые точки. Коэффициенты полинома подбираются так, чтобы обеспечить не только непрерывность функции в узловых точках, но и непрерывность ее двух производных. В переводе с английского слово *spline* означает «гибкая линейка». График функции при сплайн-интерполяции действительно напоминает гибкую линейку, закрепленную в узловых точках интерполируемой функции. Эти свойства сплайн-интерполяции позволяют эффективно применять ее даже при малом числе узловых точек — до 5—7 для простых функций.

Интерполяция реализуется с помощью следующих функций:

linterp (VX, VY, x) — возвращает значение  $y(x)$  для заданного  $x$  при линейной интерполяции,

cspline (VX, VY) — возвращает вектор VS вторых производных при сплайн-интерполяции и кубической экстраполяции,

pspline (VX, VY) — возвращает вектор VS вторых производных при сплайн-интерполяции и параболической экстраполяции,

lspline (VX, VY) — возвращает вектор VS вторых производных при сплайн-интерполяции и линейной экстраполяции.

interp (VS, VX, VY, x) — возвращает значение  $y(x)$  для заданного  $x$  при сплайн-интерполяции.

### Линейная интерполяция

$U := \begin{bmatrix} 0 \\ .2 \\ .4 \\ .6 \\ .8 \\ 1.0 \\ 1.2 \end{bmatrix}$ 
 $I := \begin{bmatrix} 0 \\ 50 \\ 20 \\ 3 \\ 4 \\ 14 \\ 55 \end{bmatrix}$

Массивы исходных  
данных ВАХ  $I(U)$

$\text{linterp}(U, I, 0.15) = 37.5$

Примеры интерполяции ВАХ

$\text{linterp}(U, I, 0.5) = 11.5$

$J(V) := \text{linterp}(U, I, V)$

Задание функции  $J(V)$  ВАХ

$V := -0.05, -0.025 \dots 1.2$

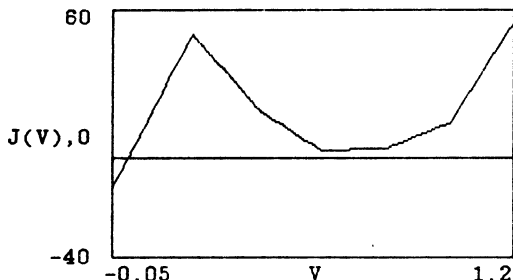


График ВАХ при  
ее линейной  
интерполяции

### Интерполяция кубическими сплайнами

$IS := \text{cspline}(U, I)$

Получение вектора  
вторых производных  $IS$

$\text{interp}(IS, U, I, 0.15) = 48.483$

Примеры сплайн-интерполяции

$\text{interp}(IS, U, I, 0.5) = 8.181$

$J(V) := \text{interp}(IS, U, I, V)$

Задание функции  $J(V)$  ВАХ

$V := -.05, -0.025 \dots 1.2$

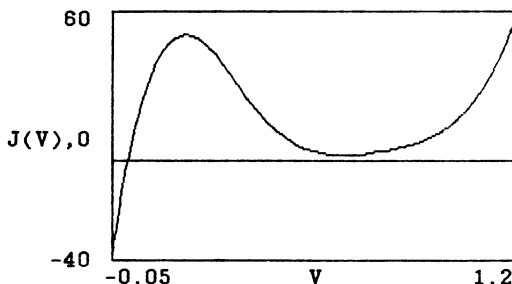


График ВАХ при  
ее интерполяции  
кубическими сплайнами

Док. 4.6. Линейная и сплайн-интерполяция N-образной ВАХ туннельного диода и построение ее графиков

Таким образом, линейная интерполяция требует применения лишь одной функции `interp`, параметрами которой являются векторы координат узловых точек  $X$  и  $Y$  и значение аргумента  $x$ . Сплайн-интерполяция имеет три варианта, различающихся способом экстраполяции функции при выходе аргумента  $x$  за пределы, ограниченные конечными точками. Экстраполяция может быть линейной, квадратичной или кубической, причем значение первой производной берется таким же, как и в крайних узловых точках.

Сплайн-интерполяция производится в две стадии. На первой с помощью функций `lspline`, `pspline` и `cspline` вырабатывается вектор вторых производных в узловых точках —  $VS$ . Затем с помощью функции `interp` находится значение  $y(x)$  для заданного  $x$ . Таким образом, сплайн-интерполяция реализуется несколько сложнее, чем линейная интерполяция.

Документ 4.6 иллюстрирует применение этих двух видов интерполяции для построения графиков  $N$ -образной вольт-амперной характеристики (ВАХ) туннельного диода, заданной всего семью узловыми точками. Первая часть документа показывает, что линейная интерполяция в данном случае дает слишком грубый график. Сплайн-интерполяция (вторая часть документа), напротив, дает прекрасное графическое отображение ВАХ, несмотря на ограниченное число узловых точек. Полезно обратить внимание на задание ВАХ в виде функции пользователя  $J(V)$ , где  $V$  — напряжение и  $J$  — ток, определяемые с помощью функций интерполяции.

#### 4.5. Функции статистики и линейной регрессии

Система MathCAD позволяет проводить наиболее распространенные статистические расчеты над данными, заданными в виде векторов. Так, для вычисления основных статистических характеристик одномерного массива из  $N$  чисел, образующих компоненты вектора  $V$ , используются следующие функции:

`mean(V)` — вычисляет среднее значение

$\text{mean} = (1/N) \sum V_i$ , где  $i = 1..N$ ,

`var(V)` — вычисляет вариацию (дисперсию) элементов вектора  $V$

$\text{var} = (1/N) \sum [V_i - \text{mean}]$ ,

`side(V)` — вычисляет среднеквадратическую погрешность, т. е. квадратный корень из вариации.

К статистическим относятся также три специальные функции:

`snorm(x)` — кумулятивная нормальная функция,

`erf(x)` — функция ошибок,

$\Gamma(x)$  — гамма-функция.

Для получения случайных чисел с равномерным распределением в отрезке  $[0..x]$  используется функция

`rnd(x)`,

создаваемая с помощью программного генератора случайных чисел. Описанная ранее команда `randomize` меняет специальный параметр генератора и позволяет получать различные последовательности случайных чисел.

Еще одна функция

`hist(intervals, data)`

служит для построения гистограмм. Она вырабатывает вектор частот попадания данных `data` в заданные интервалы, определяемые значениями элементов вектора `intervals`.

Документ 4.7 иллюстрирует применение некоторых из описанных статистических функций. В нем вначале создается вектор из 200 случайных чисел, дается их графическое представление, вычисляются основные статистические параметры и строится гистограмма распределения.

Для проведения линейной регрессии используются функции:

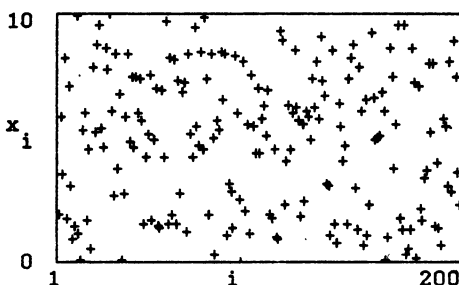
`corr(VX, VY)` — возвращает скаляр — коэффициент корреляции Пирсона,

Задание 200 случайных чисел с равномерным распределением

```
i := 1 .. 200
```

```
xi := rnd(10)
```

Графическое представление случайных чисел



При задании  
формата графика  
установлено  
type=p

Вычисление типовых статистических параметров массива x

```
mean(x) = 4.897
```

```
var(x) = 7.866
```

```
max(x) = 9.988
```

```
stdev(x) = 2.805
```

```
min(x) = 0
```

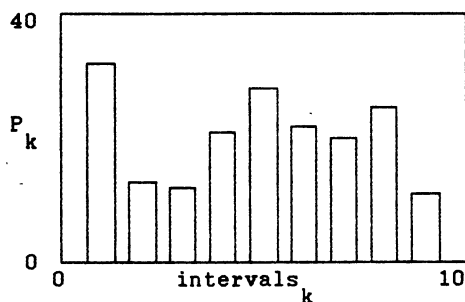
Подготовка данных к построению гистограмм

```
N := 10      j := 0 .. N      k := 0 .. N - 1
```

```
intervalsj := 1 + j ·  $\frac{10}{N}$ 
```

```
P := hist(intervals, x)
```

Построение гистограмм распределения чисел массива x



При задании  
формата графика  
установлено  
type=b

Док. 4.7. Генерация и графическое представление случайных чисел

slope(VX, VX) — возвращает значение коэффициента a для линейной регрессии  $y(x) = ax + b$ ,  
intercept(VX, VY) — возвращает значение параметра b линейной регрессии.

Ввод данных- векторов x и y с дисковых файлов

x := READPRN(data1)      N := length(x)

y := READPRN(data2)      N = 100

Вычисление основных статистических параметров

mean(x) = 495      mean(y) =  $2.125 \cdot 10^5$

var(x) =  $8.333 \cdot 10^4$       var(y) =  $2.246 \cdot 10^{10}$

stdev(x) = 288.661      stdev(y) =  $1.499 \cdot 10^5$

Проведение линейной регрессии

corr(x,y) = 0.957

m := slope(x,y)      b := intercept(x,y)

linear(x) := m · x + b

$$SSE_L := \frac{\sum (y - \text{linear}(x))^2}{N}$$

$$MSE_L := \frac{SSE_L}{N - 2} \quad MSE_L = 1.814 \cdot 10^8$$

Проведение параболической регрессии

x2 :=  $\begin{bmatrix} x^2 \end{bmatrix}$       i := 0 .. N - 1

X<sub>i,0</sub> := 1      X<sup><1></sup> := x      X<sup><2></sup> := x2

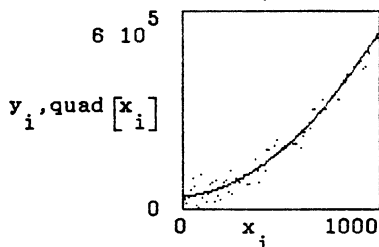
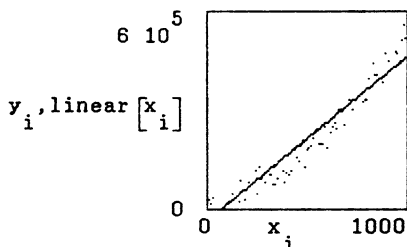
b := (X<sup>T</sup> · X)<sup>-1</sup> · (X<sup>T</sup> · y)

quad(x) := b<sub>0</sub> + b<sub>1</sub> · x + b<sub>2</sub> · x<sup>2</sup>      b =  $\begin{bmatrix} 3.753 \cdot 10^4 \\ 62.153 \\ 0.439 \end{bmatrix}$

$$SSE_Q := \frac{\sum (y - \text{quad}(x))^2}{N}$$

$$MSE_Q := \frac{SSE_Q}{N - 3} \quad MSE_Q = 8.209 \cdot 10^8$$

Графики для линейной и параболической регрессии



Док. 4.8. Линейная и параболическая регрессия

Возможно проведение с помощью этих функций и двухпараметрической нелинейной регрессии с применением известных преобразований на этапе ввода данных и вывода параметров регрессии [ 1, 2 ].

Документ 4.8 демонстрирует проведение линейной и параболической регрессии для данных, вводимых из внешних файлов, включенных в комплект поставки системы MathCAD. Данные соответствуют сильно «зашумленной» квадратичной зависимости  $y(x)$ , поэтому параболическая регрессия дает явно лучшие результаты, чем линейная.

## 4.6. Функции для решения уравнений и минимизации

Многие нелинейные уравнения, например трансцендентные, и системы из них не имеют аналитических решений и решаются графическими или численными методами. Численные методы решения с заданной (не более TOL) погрешностью вычислений можно легко реализовать в системе MathCAD.

**Функция**

**root** (Выражение, Имя переменной)

ищет значение переменной, при котором выражение становится равным нулю. Таким образом отыскивается корень уравнения вида  $F(x) = 0$ . Поиск корня осуществляется итерационным методом, причем перед этим можно задать начальное значение  $x$ . Это полезно делать и для обнаружения корня вблизи определенных, известных лишь приближенно значений  $x$ . Если уравнение  $y(x)$  имеет несколько корней, то их можно найти последовательно, используя функцию **root** при разных начальных значениях  $x$ .

Если необходимо найти решение уравнения с несколькими неизвестными или системы уравнений, задается блок уравнений. Он имеет следующую структуру: Given Служебное слово, отмечающее начало блока.

Уравнения.

Ограничительные условия.

Выражения с функциями **Find** и **Minerrg**.

Проверка решения (если необходимо).

В данном случае используются следующие функции:

**Find** ( $v_1, v_2, \dots, v_n$ ) — возвращает значение одной переменной или значения вектора переменных  $V$  (если их несколько), отвечающие точному решению;

**Minerrg** ( $v_1, v_2, \dots, v_n$ ) — возвращает значение одной переменной или значения элементов вектора переменных  $V$ , отвечающие приближенному решению с минимальной среднеквадратической погрешностью.

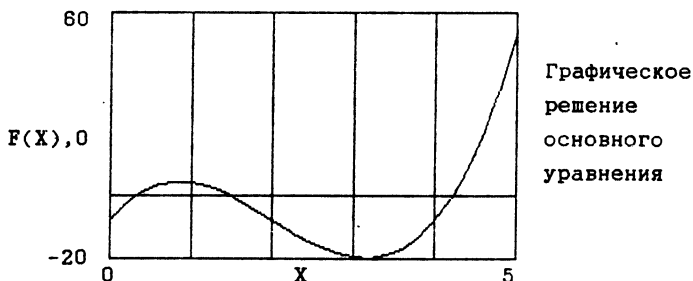
Ограничительные условия служат для ограничения области решения с помощью функции **Find** или минимизации среднеквадратической погрешности с помощью функции **Minerrg**. Они задаются следующими конструкциями ( $e$  — сокращение от слова expression):

Условие	Ввод	Пояснение
$e_1 \approx e_2$	$e_1 [Alt] = e_2$	Равно
$e_1 > e_2$	$e_1 > e_2$	Больше
$e_1 < e_2$	$e_1 < e_2$	Меньше
$e_1 \geq e_2$	$e_1 [Alt] \geq e_2$	Больше или равно
$e_1 \leq e_2$	$e_1 [Alt] \leq e_2$	Меньше или равно

Особо следует отметить еще одно применение знака  $\approx$ . Он используется для указания равенства в уравнениях при итерационных методах их решения. Документ 4.9 отражает решение актуальной для садоводов задачи — раскрой железного листа под ящик заданного объема. Ящик делается из тонкого листа с шириной  $W$  и длиной  $L$  отгибом под прямым углом полосок с шириной  $X$  со всех четырех сторон. Решение ищется вначале в виде

$$F(X) = V(X) - V_0 = 0,$$

Раскрой железного листа под ящик заданного объема  
 $V0 := 7.5$       Заданный объем ящика  
 $W := 4$       Ширина листа  
 $L := 8$       Длина листа  
 $X := 0, 0.2 \dots 5$       Расстояние отгиба листа  
 $V(X) := (L - 2 \cdot X) \cdot (W - 2 \cdot X) \cdot X$       Формула для объема листа  
 $F(X) := V(X) - V0$       Основное уравнение



#### Решение задачи

Инициализация	Решение	Комментарий
$X := 0$	$\text{root}(V(X) - V0, X) = 0.297$	Плоский ящик
$X := 1$	$\text{root}(V(X) - V0, X) = 1.5$	Глубокий ящик
$X := 4$	$\text{root}(V(X) - V0, X) = 4.203$	Корень не реален физически $X < 0.5 \cdot W$

#### Нахождение $X=XM$ для раскрой ящика максимального объема

$X := 1$       Инициализация  $X$   
 Given      Начало блока решения  
 $V(X) := (L - 2 \cdot X) \cdot (W - 2 \cdot X) \cdot X$       Основное уравнение  
 $V(X) \approx 100$       Объем, заведомо превышающий требуемый  
 $XM := \text{minerr}(X)$       Поиск оптимального  $X$   
 $XM = 0.848$       Найденное значение  $XM$   
 $V(XM) = 12.317$       Максимальный объем ящика

Док. 4.9. Раскрой железного листа под ящик заданного и максимального объемов

где  $V(X)$  — объем ящика как функция ширины полоски отгиба, а  $V0$  — требуемый объем.

Из графика  $F(X)$  можно сделать вывод, что имеется три решения, из которых только два физически реальны. Одно решение соответствует большому, но неглубокому ящику, а другое — меньшему, но глубокому. Все три решения найдены с помощью функции  $\text{root}$ . Во второй части документа решена иная задача — нахождение  $X$ , при котором из заданного листа можно скроить ящик максимально воз-

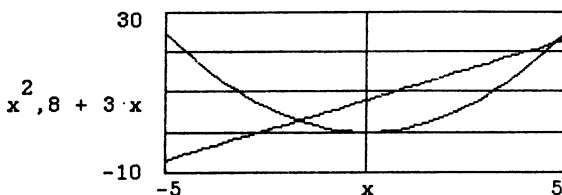


можного объема. В данном случае используется функция Minerr в составе блока уравнений.

Документ 4.10 иллюстрирует применение функции Find для решения системы уравнений. Ищутся точки пересечения параболы отрезком прямой. Из графиков этих функций видно наличие двух решений — одно для  $x < 0$  и другое для  $x > 0$ . С помощью ограничительных условий выделены оба решения. Этот пример отра-

#### Графическое решение

$x := -5, -4.75 \dots 5$



#### Блок первого решения

$x := 0$      $y := 0$     Начальные значения  $x$  и  $y$

Given

$y \leftarrow x^2$     Система уравнений

$y \leftarrow 8 + 3 \cdot x$

$x < 0$     Указание на поиск отрицательного корня

$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} := \text{Find}(x, y)$     Первое решение

$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} -1.762 \\ 2.895 \end{bmatrix}$

$x_0^2 = 2.895$     Проверка первого решения

$8 + 3 \cdot x_0 = 2.895$

#### Блок второго решения

$x := 3$      $y := 0$     Начальные значения  $x$  и  $y$

Given

$y \leftarrow x^2$     Система уравнений

$y \leftarrow 8 + 3 \cdot x$

$x > 0$     Указание на поиск положительного корня

$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} := \text{Find}(x, y)$     Второе решение

$\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} 4.702 \\ 22.105 \end{bmatrix}$

$x_0^2 = 22.105$     Проверка решения

$8 + 3 \cdot x_0 = 22.105$

Док. 4.10. Нахождение точек пересечения прямой с параболой

жает возможность нахождения нескольких корней нелинейного уравнения, что часто требуется в практических расчетах.

Для решения нелинейного уравнения оно должно быть корректно сформулировано. Некорректная формулировка решения, естественно, не даст ожидаемых результатов или приведет к отказу от решения.

Документ 4.11 показывает решение уравнения  $x^2=3$  при начальном значении  $x=10$ . Если задать  $x=10$  как точное равенство, то решение станет невозможным (см. начальную часть документа). При этом выдаются сигналы об ошибке. Однако если записать  $x^2 \approx 3$  и  $x:=10$ , то последнее равенство будет рассматриваться лишь как начальное приближение и решение будет найдено итерационным методом с приближением по минимуму среднеквадратической погрешности.

Документ 4.12 показывает еще одну интересную возможность.— применение

```
x := 10
```

```
Given
```

```
  2
```

```
x := 3
```

При начальном значении  $x=10$   
точное равенство невыполнимо  
и решение невозможно

```
x0 := Find(x),
```

```
too few constraints.
```

```
x0 = . . .
```

```
undefined
```

```
Given
```

```
  2
```

```
x ≈ 3
```

Начальное значение  $x=10$  используется  
как первое приближение, поэтому решение  
возможно

```
x0 := Find(x)
```

```
x0 = 1.732
```

Док. 4.11. Решение некорректно заданной задачи

```
G(a,x) := root [e^x - a·x^2, x]
```

```
a := 1 .. 10
```

```
x0 := 0
```

```
xa := G[a, xa-1]
```

Док. 4.12. Использование функции  
root в функции пользователя

a	x	a·x <sup>2</sup>
1	-0.704	0.495
2	-0.54	0.583
3	-0.459	0.633
4	-0.408	0.665
5	-0.371	0.68
6	-0.344	0.709
7	-0.322	0.725
8	-0.304	0.738
9	-0.289	0.749
10	-0.276	0.758

функции `root` в составе функции пользователя. Это возможно и для функций `Find` и `Minerr`.

Функции `Find` и `Minerr` предназначены для поиска действительных корней нелинейных уравнений. С их помощью возможен также поиск комплексных корней. Для этого нужно перед применением этих функций задать квадратный корень из  $-1$ . Мнимые корни нередко имеют полиномы.

Для решения систем нелинейных уравнений могут использоваться и итерационные методы. Однако в этом случае систему уравнений необходимо записывать в векторной форме. В следующих главах будет приведен ряд примеров с такой формой записи уравнений.

В заключение необходимо отметить, что не существует единого метода решения уравнений. Поэтому не исключены случаи отказа системы от решения заданного уравнения.

## 4.7. Функции с условными выражениями

Рассмотрим ряд функций, при использовании которых результат зависит от выполнения или невыполнения какого-либо условия. Вначале приведем несколько простых функций:

`floor(x)` — возвращает наибольшее целое, меньшее или равное действительному значению  $x$ ,

`ceil(x)` — возвращает наименьшее целое, большее или равное действительному значению  $x$ ,

`Phi(x)` — функция Хевисайда, возвращает 1, если  $x$  равно нулю или положительно, и 0 в противном случае,

`mod(x1, x2)` — возвращает остаток от деления  $x1$  на  $x2$  со знаком, соответствующим знаку  $x1$ .

Для реализации итерационных алгоритмов большую ценность представляют следующие две функции:

`until` (выражение 1, выражение 2) — выполняет итерации и возвращает значение выражения 2, если выражение 1 больше или равно нулю, иначе прекращает итерации,

`if` (условие, выражение 1, выражение 2) — возвращает значение выражения 1, если условие выполняется, и выражения 2, если условие не выполняется.

Документ 4.13 иллюстрирует вычисление квадратного корня из числа  $a$  итерационным методом Ньютона. Как известно, если задать  $q_0$  как начальное значение корня, то  $q_i$  будет непрерывно приближаться к точному решению при следующем итерационном процессе:

$$q_{i+1} = (q_i + a/q_i) / 2.$$

Если нужно найти решение с заданной погрешностью `egg`, то следует непрерывно следить за выполнением условия

$$|q_i^2 - a| - \text{egg} > 0.$$

Если это условие выполняется, итерации должны быть продолжены, иначе — прекращены (см. функцию `until`).

Документ 4.14 иллюстрирует применение функции `if` для математического моделирования процессов выпрямления синусоидального сигнала. Вначале задана синусоидальная функция  $i(x)$ . Затем с помощью функции `if` создана функция  $y(x)$ , дающая значения синусоидальной функции, только если синусоидальная функция положительна, и 0, если она отрицательна. Еще одна функция  $z(x)$  дает значения синусоидальной функции, если она положительна, и ее значения со сменой знака, если она отрицательна. В документе приведены временные зависимости для этих трех функций.

<code>a := 100</code>	Исходное число
<code>err := .01</code>	Погрешность вычислений
<code>M := 12</code>	Предельное число итераций
<code>i := 0 .. M</code>	Задание итерационного цикла
<code>q<sub>0</sub> := 1</code>	Задание начального приближения

i
0
1
2
3
4
5
6
7
8
9
10
11
12

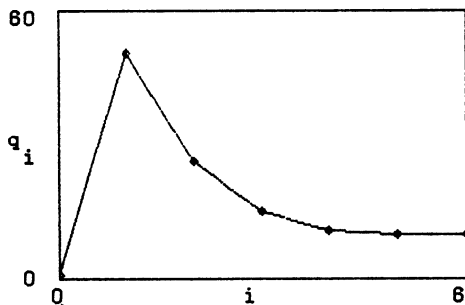
Контроль индекса i итераций

$q_{i+1} := \text{until} \left[ \left  q_i^2 - a \right  - \text{err}, \frac{q_i + \frac{a}{q_i}}{2} \right]$	Итерационная формула Ньютона
---	---------------------------------

$q = \begin{bmatrix} 50.5 \\ 26.24 \\ 15.026 \\ 10.84 \\ 10.033 \\ 10 \end{bmatrix}$	Результаты итерационных вычислений
--	--

Графическое представление приближения к корню

`i := 0 .. 6`



Док. 4.13. Вычисление квадратного корня из числа с остановкой итераций с помощью функции intel

Задание трех функций

$f(x) := \sin(x)$

$y(x) := \text{if}(f(x) > 0, f(x), 0)$

$z(x) := \text{if}(f(x) > 0, f(x), -f(x))$

Построение графиков функций

$x := 0, .1 \dots 20$

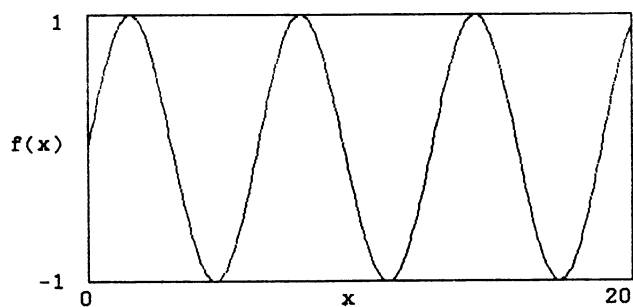


График  
 $f(x)$

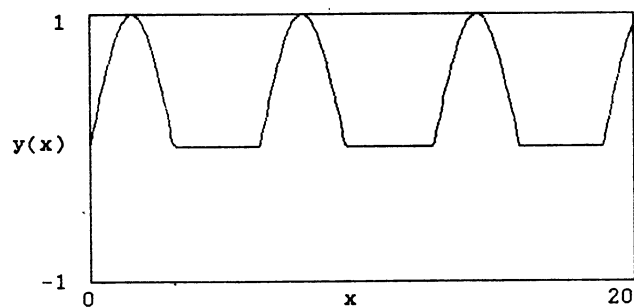


График  
 $y(x)$

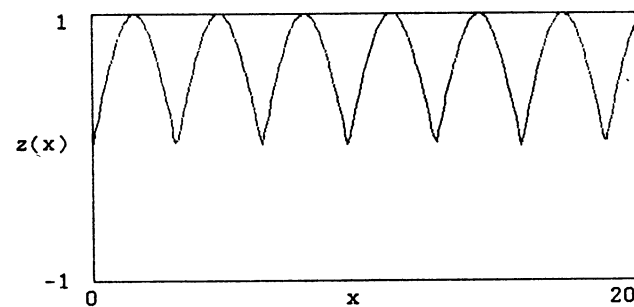


График  
 $z(x)$

Док. 4.14. Моделирование сигналов с помощью функции if

## 4.8. Функции прямого и обратного быстрого преобразования Фурье

Для решения задач спектрального анализа и синтеза система MathCAD содержит функции прямого и обратного быстрого преобразования Фурье (БПФ). Аргументами этих функций служат векторы с действительными или комплексными элементами, число которых определяется как  $2^m$ , где  $m$  — целое число. Быстрое преобразование Фурье является реализацией по специальным быстрым численным алгоритмам обычного преобразования Фурье.

Рассмотрим функции БПФ:

$\text{fft}(V)$  — обеспечивает прямое БПФ для вектора  $V$  с действительными членами, используя формулу

$$C_j = (1/\sqrt{n}) \cdot \sum V_k \cdot \exp(2\pi \cdot i \cdot (j/n) \cdot k).$$

где  $n$  — число элементов вектора  $V$  ( $n=2^m$ );  $i$  — мнимая единица;  $k$  — номер гармоники. Результирующий вектор имеет  $2^m+1$  элементов с комплексными коэффициентами. Данное преобразование означает преобразование отсчетов из временной области в частотную.

$\text{cfft}(V)$  — обеспечивает прямое БПФ, но для вектора  $V$  с комплексными коэффициентами (если коэффициенты действительные, то ввод их при использовании функции  $\text{fft}$  более прост).

$\text{ifft}(V)$  — обеспечивает обратное БПФ по формуле

$$D_j = (1/\sqrt{n}) \cdot \sum V_k \cdot \exp(2\pi \cdot i \cdot (j/n) \cdot k)$$

для вектора  $V$  с комплексными элементами, число которых равно  $2^m+1$ . Результирующий вектор имеет  $n=2^m+1$  элементов. Обратное БПФ означает преобразование из частотной области во временную.

$\text{icfft}(V)$  — обеспечивает обратное преобразование Фурье в комплексной форме (для исходного и результирующего векторов).

Документ 4.15 иллюстрирует применение прямого БПФ для получения спектра 10 гармоник прямоугольного импульса. Импульс задан 32 отсчетами, из которых восемь единичные, а остальные нулевые. Прямое БПФ дает спектр гармоник, из которого выделены нужные 10 гармоник, для них рассчитаны амплитуды и фазы. В заключение построен график огибающей амплитудного спектра и график для фазового сдвига гармоник.

В литературе встречаются альтернативные формы записи БПФ:

$$F(v) = (1/n) \cdot \sum_v f(\tau) \cdot \exp(-2\pi \cdot i \cdot (v/n) \cdot \tau),$$

$$f(\tau) = \sum_v F(v) \cdot \exp(2\pi \cdot i \cdot (v/n) \cdot \tau).$$

Для реализации БПФ в альтернативной форме можно ввести следующие две функции:

$$\text{xfft}(V) := (1/\sqrt{n}) \cdot \text{icfft}(V),$$

$$\text{ixfft}(V) := (\sqrt{n}) \cdot \text{cfft}(V).$$

Быстрое преобразование Фурье является эффективным алгоритмом быстрой обработки сигналов. Он используется, в частности, для реализации фильтрации сигналов на фоне шумов.



## 4.9. Сообщения об ошибках

Как упоминалось, система MathCAD содержит диагностические средства контроля правильности составления документа. Возникающие при этом ошибки весьма разнообразны — система предусматривает сообщения о более чем 60 типах

3DC := 12 not a name	Такое имя некорректно
DC3 = ■ undefined	Переменная не определена
$\frac{1}{0} = \cdot$ singularity	Исключительная ситуация — деление на 0
log(0) = ■ must be nonzero	Аргумент должен быть не 0
i := 0,1 ..-5 illegal range	Ошибка в пределе
$V := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$	
$V_5 = \cdot$ index out of bounds	Индекс за пределами границ индексации
P := 1 P_5 = ■ must be vector	Элемент должен быть вектором
Yn(.12,1) = ■ must be integer	Должно быть целым
interp(VX,VY) = ■ too few arguments	Слишком мало аргументов
Yn(2,3,4) = ■ too many arguments	Слишком много аргументов
Y0(-1) = ■ domain error	Ошибка в области определения
ORIGIN := 100000	Некорректно значение ORIGIN
A_5 := 0 illegal ORIGIN	
READ(Error) File not found	Файл не найден

Док. 4.16. Наиболее характерные ошибки



ошибок. К счастью, с большинством из них Вы, возможно, и не столкнетесь.

Документ 4.16 содержит перечень примеров с наиболее характерными ошибками. На нем дан и перевод сообщений об ошибках с английского языка на русский. Напоминаем, что сигнал ошибки появляется в прямоугольном шаблоне в виде надписи на английском языке. Сам шаблон располагается снизу от выражения, в котором найдена ошибка. Однако исходящая от прямоугольника черта указывает непосредственное место ошибки.

Нередко одна ошибка, например неопределенная вначале переменная, может привести к каскаду ошибок. При этом сообщения об ошибках накладываются друг на друга и нередко заслоняют друг друга. В этом случае надписи об ошибках прочесть невозможно. В такой ситуации очень полезна команда *Separate* (разделение), имеющаяся в позиции *Edit/move* основного меню. Ввод этой команды автоматически раздвигает блоки документа, так что все сообщения об ошибках становятся доступными для их прочтения.

Следует отметить, что многие ошибки отличаются от приведенных лишь уточнением признака ошибки, например *must be 3-vector* (должен быть трехэлементный вектор), *must be array* (должен быть массив), *must be scalar* (должен быть скаляр) и т. д.

Тщательная продуманность языка диалога системы с пользователем и неоднократно упоминаемая близость языка к типовому языку математической литературы резко сокращают возможность возникновения синтаксических ошибок — весьма частых при реализации математических расчетов с помощью программ, составленных на обычных языках программирования. Это наряду с обилием встроенных и наиболее распространенных математических возможностей делает решение сложных задач надежным и достоверным.

## 4.10. Численные методы, реализованные в системе

При проведении серьезных расчетов пользователя не могут не интересовать численные методы, реализованные в системе MathCAD. Дело в том, что нет абсолютно надежных численных методов, способных решать любые задачи [5—8]. Есть, к счастью, немного задач, при которых те или иные методы могут не дать достоверных результатов, а порой привести к катастрофическому росту погрешности вычислений или даже к аварийному останову ЭВМ (к технической аварии это отношения не имеет).

Для вычисления производных система MathCAD использует следующий алгоритм:

1. Если значение  $|x|$  в выражении  $\frac{d}{dx}f(x)$  больше  $10^5$ , то задается шаг  $h = \sqrt{\text{TOL} \cdot x}$ ; иначе  $h = \sqrt{\text{TOL}}$ .

2. Для  $h=1$  вычисляется производная по формуле для двух точек:

$$\text{estimate}_0 = [f(x+h) - f(x-h)] / (2h).$$

3. Вычисляется производная по формуле для пяти точек (или, точнее, четырех точек, расположенных симметрично относительно центральной точки):

$$\text{estimate}_n = [f(x-h) - 8 \cdot f(x-h/2) + 8 \cdot f(x+h/2) - f(x+h)] / (6 \cdot h).$$

4. Вычисляется величина  $|\text{estimate}_n - \text{estimate}_{n-1}| / 15$ . Если она меньше величины  $\text{TOL} \cdot |\text{estimate}_n|$ , то вычисления прекращаются; иначе  $n$  увеличивается на 1 и происходит переход на шаг 3 алгоритма.

Данный метод позволяет для гладких функций  $f(x)$  получать значения производных с погрешностью, не превышающей значения системной переменной *TOL* (по умолчанию  $\text{TOL} = 0.001$ ). Документ 4.17 иллюстрирует численное дифференцирование для различных значений погрешности *TOL*.

Для численного интегрирования с заданной погрешностью применяется метод Симпсона, реализованный следующим алгоритмом. Вначале при восьми интервалах отрезка интегрирования  $[a..b]$  обычным методом Симпсона вычисляется

$$I_k = \int_a^b f(x) dx.$$

Затем число интервалов увеличивается вдвое, вычисляется  $I_{k+1}$  ( $k=1, 2, 3, \dots$ ) и т. д. Производится сравнение разности  $I_{k+1} - I_k$  с величиной TOL, и, если первая больше, вычисления продолжаются, иначе они заканчиваются. Документ 4.18 иллюстрирует численное интегрирование с различной погрешностью TOL.

Для монотонных функций данный метод дает хорошие результаты. Однако он может привести к аварийному останову, если при  $x=a$  или  $x=b$  функция  $f(x) \rightarrow \infty$ . Трудности возникают и в том случае, когда один из пределов равен бесконечности. Документ 4.19 иллюстрирует один из приемов вычисления несобственного интеграла, у которого бесконечный верхний предел интегрирования заменен конечным, таким, при котором  $f(b) \rightarrow 0$ .

Могут вычисляться и кратные интегралы (см. рис. 1.2, на котором было показано вычисление двойного интеграла). Однако вычисление интегралов с большей кратностью ведет к резкому увеличению времени вычислений и может вызвать заикливание.

Поиск действительных корней у нелинейных уравнений выполняется хорошо известным методом секущих [2]. Итерации производятся до тех пор, пока разность найденных приближенно значений корня не станет меньше значения TOL.

<b>X := 1</b>	<b>Значение аргумента</b>
<b>TOL := 0.01</b>	<b>Задание погрешности</b>
<b>f(x) := sin(x)</b>	<b>Задание функции</b>
$\frac{d}{dx} f(X) = 0.540302193338657$	<b>Вычисление производной</b>
<b>TOL := 0.000001</b>	<b>Задание меньшей погрешности</b>
$\frac{d}{dx} f(X) = 0.540302305867982$	<b>Новое вычисление производной</b>
<b>cos(X) = 0.54030230586814</b>	<b>Точное значение <math>\sin'(x) = \cos(x)</math></b>

Док. 4.17. Вычисление производной с различной погрешностью

<b>TOL := 0.1</b>	<b>Задание погрешности</b>
$\int_{-39}^1 e^x dx = 2.749262872839197$	<b>Вычисление интеграла</b>
<b>TOL := 0.000001</b>	<b>Задание меньшей погрешности</b>
$\int_{-39}^1 e^x dx = 2.71828186361373$	<b>Вычисление интеграла</b>
<b>e = 2.718281828459045</b>	<b>Точное значение</b>

Док. 4.18. Вычисление определенного интеграла с различной погрешностью TOL

TOL := 0.01	Задание погрешности
$f(x) := \frac{\sin(x)}{x}$	Задание функции с особенностью при $x \rightarrow 0$ (неопределенность $0/0 \rightarrow 1$ )
$I := \int_0^1 f(x) dx$	Этот интеграл вычисляется, так как MathCAD трактует деление на 0 как большое, но конечное число
I = 0.940874737092495	
$I := \int_{.00001}^1 f(x) dx$	Здесь особый нижний предел 0 заменен малым, но конечным
I = 0.946073071305514	

TOL := 0.001	Задание погрешности
$I := \int_0^{100} x \cdot e^{-x} dx$	В этом интеграле бесконечный верхний предел заменен конечным
I = 0.99997592935233	Точное значение 1

Графическая иллюстрация допустимости замены  
бесконечного верхнего предела конечным

$x := 0, 0.2 \dots 10$

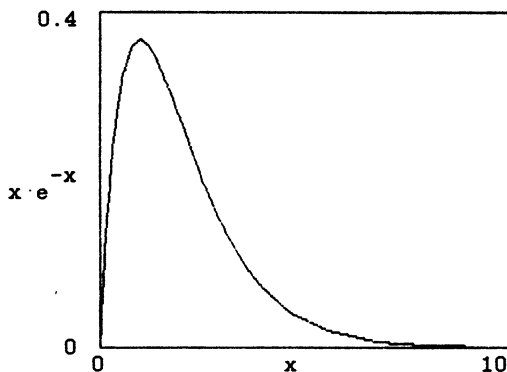


График функции  
 $x \cdot e^{-x} \rightarrow 0$   
при больших  $x$

Док. 4.19. Вычисление несобственных интегралов

Перед началом итераций задается начальное значение  $x_0$ . Его выбором можно локализовать область, в которой находится искомый корень.

При численном решении системы из нескольких нелинейных уравнений их следует записывать в векторной форме. Примеры этого даны в приложении. Аналогичным образом могут решаться системы нелинейных дифференциальных уравнений. Для этого они заменяются конечно-разностными уравнениями и для решения используются известные численные методы, например Эйлера или Рунге—Кутты.

Следует отметить, что по скорости решения сложных задач система MathCAD уступает специализированным программам, даже если они написаны на языке Бейсик. Однако по времени подготовки задач к решению, наглядности и сервисным, в том числе графическим, возможностям система MathCAD имеет неоспоримые преимущества. При использовании современных персональных компьютеров класса IBM PC AT и PS/2, особенно если они оснащены математическим сопроцессором, скорость решения задач перестает быть важным фактором и отмеченные достоинства становятся первостепенными.

## 5. СПЕЦИАЛЬНЫЕ ВИДЫ МАТЕМАТИЧЕСКИХ И ФИЗИЧЕСКИХ РАСЧЕТОВ

### 5.1. Логические функции и их реализация

Система MathCAD в отличие от большинства языков программирования персональных компьютеров не имеет встроенных логических функций. Такие функции оперируют аргументами типа ДА, НЕТ или TRUE (верно), FALSE (фальшиво или неверно) и т. д. Мы будем пользоваться понятиями логической единицы 1 и логического нуля 0.

Простейшей логической функцией является функция отрицания  $\text{not}(a)$ , где аргумент  $a$  может иметь значение 0 или 1. О характере действия логической функции дает представление ее таблица состояний, которая показывает логическое состояние результата  $r$  в зависимости от логических состояний аргументов. Таблица состояний для функции  $\text{not}$  имеет вид

$a$	$r$	Выражение для $r$
0	1	
1	0	$r = \bar{a}$

Черта над аргументом здесь означает операцию инвертирования. Функция  $\text{and}(a, b)$  обеспечивает проведение операции логического умножения (И) в соответствии со следующей таблицей состояний:

$a$	$b$	$r$	Выражение для $r$
0	0	0	
1	0	0	
0	1	0	$r = a \cdot b$
1	1	1	

Функция логического сложения (ИЛИ)  $\text{or}(a, b)$  характеризуется следующей таблицей состояний:

$a$	$b$	$r$	Выражение для $r$
0	0	0	
1	0	1	
0	1	1	$r = a + b$
1	1	1	

Еще одна функция (ИСКЛЮЧАЮЩАЯ ИЛИ)  $\text{xor}(a, b)$  выполняет следующие действия:

$a$	$b$	$r$	Выражение для $r$
0	0	0	
1	0	1	
0	1	1	
1	1	0	$r = a + b - a \cdot b$

Для реализации логических функций в системе MathCAD можно использовать функцию Хевисайда  $\Phi(x)$ , задав в качестве аргумента  $x$  одно из приведенных

Задание логических функций

$\text{not}(a) := \Phi(.5 - a)$

$\text{not}(0) = 1 \quad \text{not}(1) = 0$

$\text{and}(a,b) := \Phi(a \cdot b - .5)$

$\text{and}(0,0) = 0 \quad \text{and}(0,1) = 0 \quad \text{and}(1,0) = 0 \quad \text{and}(1,1) = 1$

$\text{or}(a,b) := \Phi(a + b - .5)$

$\text{or}(0,0) = 0 \quad \text{or}(0,1) = 1 \quad \text{or}(1,0) = 1 \quad \text{or}(1,1) = 1$

$\text{xor}(a,b) := \Phi(a + b - 2 \cdot a \cdot b - .5)$

$\text{xor}(0,0) = 0 \quad \text{xor}(0,1) = 1 \quad \text{xor}(1,0) = 1 \quad \text{xor}(1,1) = 0$

$\text{max2}(a,b) := \text{if}(a > b, a, b)$

$\text{max2}(5,7) = 7 \quad \text{max2}(7,5) = 7$

$\text{min2}(a,b) := \text{if}(a < b, a, b)$

$\text{min2}(5,7) = 5 \quad \text{min2}(7,5) = 5$

Примеры, иллюстрирующие действие логической функции **and**

$N := 20$

$i := 1 \dots N$

$S1_i := \Phi(\text{rnd}(1) - 0.5)$

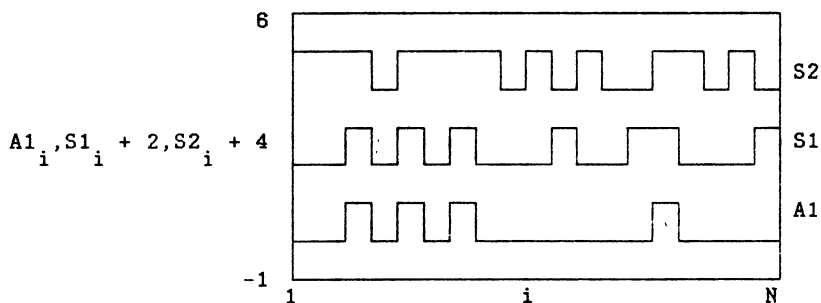
Задание сигнала 1

$S2_i := \Phi(\text{rnd}(1) - 0.5)$

Задание сигнала 2

$A1_i := \text{and}[S1_i, S2_i]$

Сигнал, который является  
логическим **and**( $S1, S2$ )



Док. 5.1. Задание логических функций и операции с ними

выше выражений. Однако при этом надо учитывать погрешности вычислений. Чтобы они не сказывались, можно определить указанные логические функции следующим образом:

$\text{not}(a) := \Phi(0.5 - a)$ ,

$\text{and}(a, b) := \Phi(a \cdot b - 0.5)$ ,

$$\begin{aligned}\text{ог}(a, b) &:= \Phi(a + b - 0.5), \\ \text{хог}(a, b) &:= \Phi(a + b - 2 \cdot a \cdot b - 0.5).\end{aligned}$$

Число 0.5 в этих выражениях устраняет возможность ошибок из-за погрешности операций при аргументе функции Хевисайда, близком к нулю.

К логическим можно условно отнести еще две полезные функции двух аргументов:

$$\begin{aligned}\text{max2}(a, b) &:= \text{if}(a > b, a, b), \\ \text{min2}(a, b) &:= \text{if}(a < b, a, b).\end{aligned}$$

Первая из них возвращает значение большего из двух аргументов, вторая — меньшего.

Документ 5.1 иллюстрирует задание и применение указанных выше логических функций. Интересный пример действия логической функции and приведен в конце документа. Здесь с помощью генератора случайных чисел и функции Хевисайда сформированы два случайных сигнала, имеющих в любой момент времени значение 0 или 1. Эти сигналы обозначены S1 и S2. Сигнал A является функцией логического умножения этих сигналов. Далее строится график всех этих сигналов. По существу этот пример моделирует работу анализатора логических состояний — прибора, который широко используется в электронике для отладки и тестирования логических устройств на интегральных микросхемах.

## 5.2. Итерации в векторной форме

Иногда в итерационном процессе решения должны участвовать несколько взаимозависимых переменных. Такие случаи встречаются при решении систем нелинейных, рекуррентных или дифференциальных уравнений. Решение систем нелинейных и дифференциальных уравнений рассматривается далее, а здесь мы остановимся на решении систем рекуррентных уравнений. Во всех этих случаях решение должно быть задано в векторной форме.

Примером решения системы рекуррентных уравнений может служить числовая модель развития эпидемии. В ней участвуют четыре переменные, отражающие следующие факторы:  $i$  — инфекция,  $s$  — восприимчивость к инфекции,  $d$  — смертность и  $r$  — выздоравливаемость. Изменение этих факторов во времени  $t$  описывается системой из четырех рекуррентных уравнений:

$$i_{t+1} = K_i \cdot s_t \cdot i_t,$$

$$s_{t+1} = s_t - k_s \cdot s_t \cdot i_t,$$

$$d_{t+1} = d_t + k_d \cdot i_t,$$

$$r_{t+1} = r_t + K_r \cdot i_t.$$

Здесь коэффициенты вида  $K_i$  имеют числовые значения, учитывающие эффективность действия указанных факторов. Видно, что характер итераций оказывается перекрестным, например  $i_{t+1}$  зависит от  $s_t$ , а  $s_{t+1}$  — от  $i_t$  и т. д. Система MathCAD позволяет провести такие итерации в векторной форме.

Документ 5.2 показывает реализацию решения описанной выше системы уравнений. Результаты решения представлены как в табличной, так и графической форме, которая позволяет проследить за характером развития эпидемии.

t := 0 ..20

Время

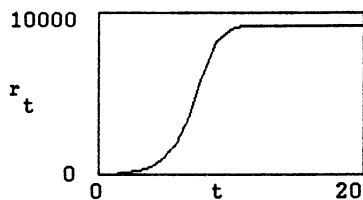
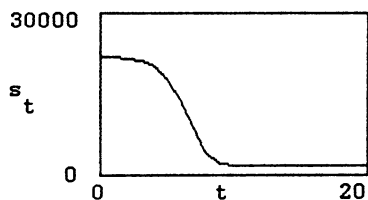
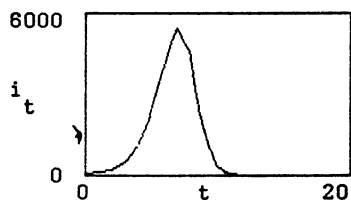
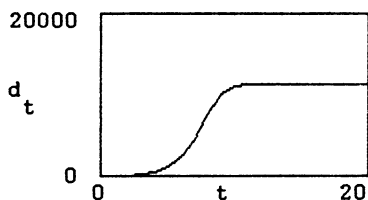
$$\begin{bmatrix} i \\ s \\ d \\ r \end{bmatrix}_0 := \begin{bmatrix} 50 \\ 22000 \\ 0 \\ 0 \end{bmatrix}$$

Начальные условия

Перекрестные итерации ( в векторной форме )

$$\begin{bmatrix} i \\ s \\ d \\ r \end{bmatrix}_{t+1} := \begin{bmatrix} 0.0001 \cdot s_t \cdot i_t \\ s_t - 0.0001 \cdot s_t \cdot i_t \\ d_t + 0.55 \cdot i_t \\ r_t + 0.45 \cdot i_t \end{bmatrix} \quad \begin{array}{l} \dots \text{инфекция} \\ \dots \text{восприимчивость} \\ \dots \text{смертность} \\ \dots \text{выздоровляемость} \end{array}$$

t	d <sub>t</sub>	i <sub>t</sub>	s <sub>t</sub>	r <sub>t</sub>
0	0	50	22000	0
1	27.5	110	21890	22.5
2	88	240.79	21649.21	72
3	220.435	521.281	21127.919	180.356
4	507.145	1101.38	20026.539	414.937
5	1112.904	2205.853	17820.836	910.558
6	2326.029	3930.716	13690.14	1803.115
7	4487.823	5459.819	8430.32	3671.837
8	7490.824	4602.803	3827.518	6128.856
9	10022.365	1761.731	2065.787	8200.117
10	10991.317	363.936	1701.851	8992.896
11	11191.482	61.936	1639.914	9156.667
12	11225.57	10.157	1629.757	9184.539
13	11231.133	1.855	1628.102	9189.109
14	11232.044	0.27	1627.832	9189.854
15	11232.192	0.044	1627.789	9189.875
16	11232.216	0.007	1627.78	9189.895
17	11232.22	0.001	1627.78	9189.898
18	11232.221	0	1627.78	9189.899
19	11232.221	0	1627.78	9189.899
20	11232.221	0	1627.78	9189.899



Док. 5.2. Моделирование развития эпидемии

### 5.3. Решение дифференциального уравнения первого порядка

В связи с большим разнообразием типов дифференциальных уравнений в систему MathCAD не включены специальные средства их решения. Однако как отдельные дифференциальные уравнения, так и их системы можно решать обычными численными методами. Дифференциальным уравнениям принадлежит первостепенная роль в решении многих математических, физических и технических задач. Поэтому несколько параграфов этой главы посвящено решению таких уравнений.

Даже для решения одного дифференциального уравнения первого порядка, имеющего вид

$$dy/dx=f(x, y),$$

существует множество различных методов решения [1—8], отличающихся прежде всего погрешностью вычислений.

Обычно решение заключается в нахождении ряда значений  $x_i$  и  $y_i$  искомой зависимости  $y(x)$  при  $i$ , изменяющемся от 0 до  $N$  при шаге изменения  $x$ , равном  $h$ . Мы будем рассматривать три наиболее распространенных способа решения дифференциального уравнения, при которых  $h=\text{const}$ . При этом для  $i=0$  значения  $x_0$  и  $y_0$  должны быть известны как начальные условия, без чего невозможно единственное решение.

Простой метод Эйлера реализуется применением на каждом шаге вычислений следующих итерационных выражений:

$$x_{i+1} = x_i + h,$$

$$y_{i+1} = y_i + h \cdot f(x, y).$$

Документ 5.3 иллюстрирует решение этим методом дифференциального уравнения

$$dy/dx=x \cdot y,$$

известное аналитическое решение которого имеет вид

$$y(x)=A \cdot \exp(x^2/2),$$

причем  $A=1$ ,  $x_0=0$  и  $y_0=1$ . Наличие аналитического решения позволяет легко оценить погрешность метода, которая в общем пропорциональна  $h^2$ .

Обычно говорят о порядке метода  $n$ , если его погрешностью имеет порядок  $n-1$ . Таким образом, простой метод Эйлера имеет первый порядок. Будучи самым простым из известных, этот метод имеет и наибольшую погрешность. Последнее не должно служить поводом для паники, поскольку при малых  $h$  и достаточно гладких решениях этот метод вполне приемлем для решения многих практических задач.

Как видно из док. 5.3, в котором дано решение при довольно большом шаге  $h$ , решение численным методом количественно заметно отличается от точного аналитического решения. Это хорошо заметно и из рассмотрения графиков решения. В то же время качественный характер численного решения остается верным.

Для уменьшения погрешности решения следует применять методы более высокого порядка. Модифицированный метод Эйлера, например, является методом второго порядка, и его погрешность пропорциональна  $h^3$ . Данный метод реализуется следующими итерационными выражениями на каждом шаге вычислений:

$$x_{i+1} = x_i + h,$$

$$y_{i+1}^1 = y_i + h \cdot f(x_i, y_i)/2,$$

$$y_i = y_i + h \cdot f(x_i + h/2, y_{i+1}^1).$$



$h := 0.1$  Шаг изменения  $x$   
 $N := 10$  Число шагов  
 $f(x, y) := x \cdot y$  Функция  $y' = f(x, y)$

$i := 0 \dots N - 1$  Задание цикла

$\begin{bmatrix} x \\ y \end{bmatrix}_0 := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$  Задание начальных условий

$\begin{bmatrix} x \\ y \end{bmatrix}_{i+1} := \begin{bmatrix} x_i + h \\ y_i + h \cdot f[x_i, y_i] \end{bmatrix}$  Итерационные уравнения

Результаты решения

$x = \begin{bmatrix} 0 \\ 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \\ 0.5 \\ 0.6 \\ 0.7 \\ 0.8 \\ 0.9 \\ 1 \end{bmatrix}$

Значения  $x$

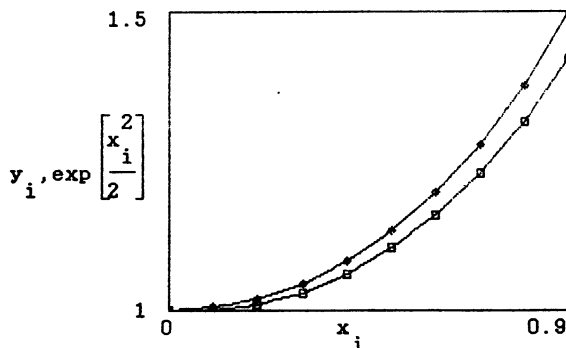
$y = \begin{bmatrix} 1 \\ 1.01 \\ 1.03 \\ 1.061 \\ 1.104 \\ 1.159 \\ 1.228 \\ 1.314 \\ 1.419 \\ 1.547 \end{bmatrix}$

Решение методом E1

exp  $\begin{bmatrix} 2 \\ x_i \\ 2 \end{bmatrix}$   

1.005
1.02
1.046
1.083
1.13
1.19
1.278
1.377
1.499

Точное решение



<- Точное решение

<- Приближенное решение

Графическое представление решения

Док. 5.3. Решение дифференциального уравнения простым методом Эйлера (E1)

Следует отметить, что здесь каждая формула порождает (при ее реализации по правилам системы MathCAD) массив значений соответствующей переменной. При большом числе точек решения это ведет к заметным затратам памяти. Поэтому целесообразно отказаться от вычисления промежуточного значения и реализовать вычисления по формулам

$$x_{i+1} = x_i + h,$$

$$y_{i+1} = y_i + h \cdot f(x_i + h/2, y_i + h \cdot f(x_i, y_i)/2).$$

В этом случае один из массивов удается исключить.

Документ 5.4 иллюстрирует реализацию данного метода с выводом результатов

$h := 0.1$

Шаг изменения  $x$

$N := 10$

Число шагов

$f(x, y) := x \cdot y$

Функция  $y' = f(x, y)$

$i := 0 \dots N - 1$

Задание цикла

$\begin{bmatrix} x \\ y \end{bmatrix}_0 := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Задание начальных условий

Итерационные уравнения в векторной форме

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} := \begin{bmatrix} x_i + h \\ y_i + h \cdot f\left[x_i + \frac{h}{2}, y_i + h \cdot \frac{f[x_i, y_i]}{2}\right] \end{bmatrix}$$

Результаты решения

$x = \begin{bmatrix} 0.0 \\ 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \\ 0.5 \\ 0.6 \\ 0.7 \\ 0.8 \\ 0.9 \\ 1.0 \end{bmatrix}$

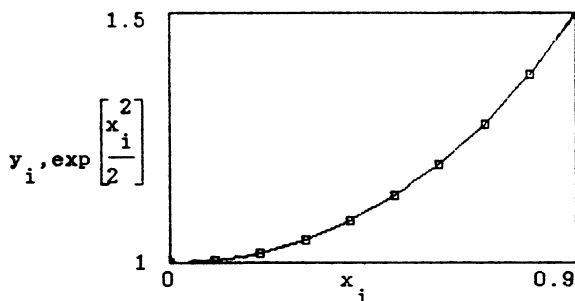
Значения  
 $x$

$y = \begin{bmatrix} 1.005 \\ 1.02 \\ 1.046 \\ 1.083 \\ 1.133 \\ 1.197 \\ 1.277 \\ 1.376 \\ 1.497 \\ 1.646 \end{bmatrix}$

Решение  
методом E1

$\begin{bmatrix} x_i \\ y_i \end{bmatrix}_{\text{exp}} = \begin{bmatrix} 1.005 \\ 1.02 \\ 1.046 \\ 1.083 \\ 1.133 \\ 1.197 \\ 1.277 \\ 1.376 \\ 1.497 \\ 1.646 \end{bmatrix}$

Точное  
решение



<- Точное и  
приближенное  
решения практи-  
чески совпадают

Графическое  
представление  
решения

Док. 5.4. Решение дифференциального уравнения модифицированным методом Эйлера (E1)

h := 0.1    N := 10    Шаг изменения x и число шагов  
f(x,y) := x · y    Функция y' = f(x,y)

k1(x,y) := h · f(x,y)

Задание

k2(x,y) := h f [x +  $\frac{h}{2}$ , y +  $\frac{k1(x,y)}{2}$ ]

коэффициентов

k3(x,y) := h f [x +  $\frac{h}{2}$ , y +  $\frac{k2(x,y)}{2}$ ]

k1, k2, k3, k4 и k

k4(x,y) := h f(x + h, y + k3(x,y))

как функций

пользователя

k(x,y) := k1(x,y) + 2 k2(x,y) + 2 k3(x,y) + k4(x,y)

i := 0 .. N - 1    Задание цикла

$\begin{bmatrix} x \\ y \end{bmatrix}_0 := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Задание начальных условий

$\begin{bmatrix} x \\ y \end{bmatrix}_{i+1} := \begin{bmatrix} x_i + h \\ y_i + \frac{k[x_i, y_i]}{6} \end{bmatrix}$

Итерационные уравнения

Результаты вычислений

x =  $\begin{bmatrix} 0 \\ 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \\ 0.5 \\ 0.6 \\ 0.7 \\ 0.8 \\ 0.9 \\ 1 \end{bmatrix}$

Значения x

y =  $\begin{bmatrix} 1 \\ 1.005 \\ 1.02 \\ 1.046 \\ 1.083 \\ 1.133 \\ 1.187 \\ 1.278 \\ 1.377 \\ 1.499 \\ 1.649 \end{bmatrix}$

Решение методом RK4

exp  $\begin{bmatrix} 2 \\ x_i \\ 2 \end{bmatrix}$   

1.005
1.02
1.046
1.083
1.133
1.187
1.278
1.377
1.499
1.649

Точное решение

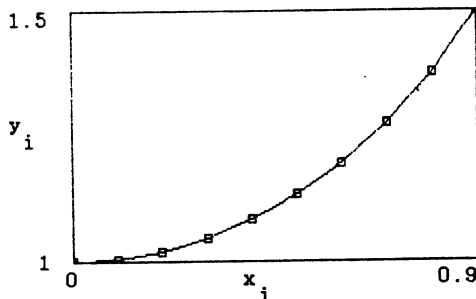


График решения

Док. 5.5. Решение дифференциального уравнения методом Рунге—Кутты (RK4)

в табличной и графической форме. Теперь уже на графике отличить численное решение от аналитического практически не удастся.

Тем не менее при высоких требованиях к точности решения можно воспользоваться методом Рунге—Кутты четвертого порядка. При нем погрешность пропорциональна  $h^5$ . Этот метод реализуется с помощью следующих используемых на каждом шаге вычислений выражений:

$$\begin{aligned}k_1 &= h \cdot f(x_i, y_i), \\k_2 &= h \cdot f(x_i + h/2, y_i + k_1/2), \\k_3 &= h \cdot f(x_i + h/2, y_i + k_2/2), \\k_4 &= h \cdot f(x_i + h, y_i + k_3), \\x_{i+1} &= x_i + h, \\y_{i+1} &= y_i + (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4) / 6.\end{aligned}$$

В таком общепринятом виде формулы в системе MathCAD задавать нельзя, поскольку переменные  $k_1$ ,  $k_2$ ,  $k_3$  и  $k_4$ , определяемые через индексированные переменные  $x_i$  и  $y_i$ , должны быть индексированными переменными, т. е. элементами массивов. Однако есть возможность избавиться от задания четырех массивов, если определить  $k_1$ ,  $k_2$ ,  $k_3$  и  $k_4$  как функции пользователя с аргументами  $x_i$  и  $y_i$ .

Документ 5.5 иллюстрирует применение этого приема для решения дифференциального уравнения методом Рунге-Кутты. Благодаря данному приему векторная запись итерационных процедур задана только для переменных  $x_i$  и  $y_i$ . Точность решения методом Рунге-Кутты контрольного примера столь высока, что на графиках погрешность обнаружить невозможно. Небольшие отличия можно выявить, лишь внимательно проанализировав табличные данные.

## 5.4. Решение дифференциального уравнения второго порядка

Дифференциальные уравнения второго порядка вида

$$y'' = d^2y/dx^2 = f(x, y, z),$$

где

$$z = y' = dy/dx,$$

также относятся к наиболее распространенным. Достаточно отметить, что они обычно описывают волновые процессы в механике, электротехнике и радиотехнике.

Мы ограничимся примером решения таких уравнений лишь одним методом — Рунге—Кутта четвертого порядка. Более простые методы читатель сможет реализовать сам. Полагая, что ему очевидна и аналогия между математической записью решения на каждом шаге и его векторной реализацией в системе MathCAD, приведем только док. 5.6. Как и ранее, коэффициенты Рунге—Кутта (они несколько иные, чем для реализации уравнения первого порядка) определены как функции пользователя для уменьшения числа используемых массивов.

Контрольный пример в док. 5.6 задает решение следующего дифференциального уравнения:

$$y'' = \frac{3}{4} \cdot (x \cdot z / \sqrt{y} + 1).$$

Погрешность решения этим методом пропорциональна  $h^5$ .

$$F(x, y, z) := 3 \frac{x \frac{z}{\sqrt{y}} + 1}{4} \quad \text{Функция } y'' = F(x, y, y'),$$

h := 0.05      N := 5      где z = y'

$$K1(x, y, z) := h \cdot F(x, y, z)$$

$$K2(x, y, z) := h \cdot F\left[x + \frac{h}{2}, y + h \cdot \frac{z}{2} + h \cdot \frac{K1(x, y, h)}{8}, z + \frac{K1(x, y, z)}{2}\right]$$

$$K3(x, y, z) := h \cdot F\left[x + \frac{h}{2}, y + h \cdot \frac{z}{2} + h \cdot \frac{K1(x, y, z)}{8}, z + \frac{K2(x, y, z)}{2}\right]$$

$$K4(x, y, z) := h \cdot F\left[x + h, y + h \cdot z + h \cdot \frac{K3(x, y, z)}{2}, z + K3(x, y, z)\right]$$

$$K(x, y, z) := \frac{K1(x, y, z) + K2(x, y, z) + K3(x, y, z)}{6}$$

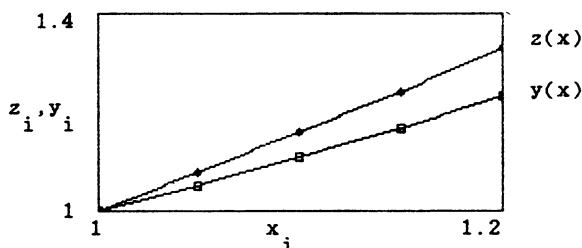
$$S(x, y, z) := \frac{K1(x, y, z) + 2 \cdot K2(x, y, z) + 2 \cdot K3(x, y, z) + K4(x, y, z)}{6}$$

$$i := 0 \dots N - 1$$

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \text{Начальные условия}$$

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ z_{i+1} \end{bmatrix} := \begin{bmatrix} x_i + h \\ y_i + h \left[ z_i + K[x_i, y_i, z_i] \right] \\ z_i + S[x_i, y_i, z_i] \end{bmatrix} \quad \text{Итерационные уравнения}$$

$$x = \begin{bmatrix} 1.05 \\ 1.1 \\ 1.15 \\ 1.2 \\ 1.25 \end{bmatrix}, \quad y = \begin{bmatrix} 1.052 \\ 1.108 \\ 1.168 \\ 1.232 \\ 1.301 \end{bmatrix}, \quad z = \begin{bmatrix} 1.077 \\ 1.158 \\ 1.243 \\ 1.332 \\ 1.426 \end{bmatrix}$$



Док. 5.6. Решение дифференциального уравнения второго порядка методом Рунге—Кутта

## 5.5. Решение систем дифференциальных уравнений

Системы обыкновенных дифференциальных уравнений имеют вид

$$dy/dx = f_1(x, y, z, \dots),$$

$$dz/dx = f_2(x, y, z, \dots),$$

.....

$F_1(x, y, z) := x \cdot y + z$       Задание функции F1

$F_2(x, y, z) := x + y \cdot z$       Задание функции F2

$h := 0.05$       Шаг изменения  $x$

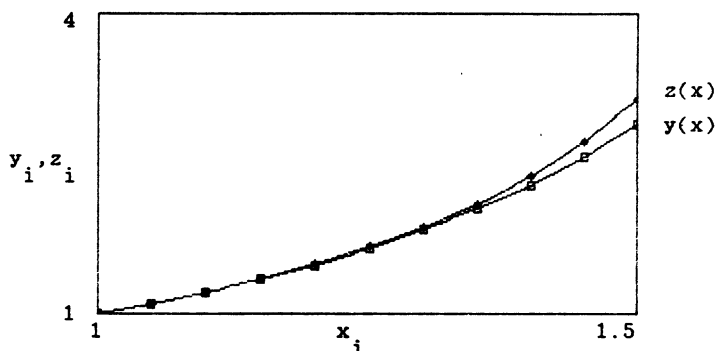
$N := 10$       Число шагов

$i := 0 \dots N$       Начало цикла

$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_0 := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$       Задание начальных условий

$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ z_{i+1} \end{bmatrix} := \begin{bmatrix} x_i + h \\ y_i + h \cdot F_1 \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \\ z_i + h \cdot F_2 \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \end{bmatrix}$       Итерационные уравнения в векторной форме

$$x = \begin{bmatrix} 1.05 \\ 1.105 \\ 1.16 \\ 1.215 \\ 1.27 \\ 1.325 \\ 1.38 \\ 1.435 \\ 1.49 \\ 1.545 \\ 1.55 \end{bmatrix} \quad y = \begin{bmatrix} 1.1 \\ 1.213 \\ 1.34 \\ 1.484 \\ 1.648 \\ 1.834 \\ 2.046 \\ 2.289 \\ 2.568 \\ 2.89 \\ 3.263 \end{bmatrix} \quad z = \begin{bmatrix} 1.1 \\ 1.213 \\ 1.342 \\ 1.489 \\ 1.659 \\ 1.859 \\ 2.084 \\ 2.376 \\ 2.713 \\ 3.139 \\ 3.668 \end{bmatrix}$$



Док. 5.7. Решение системы из двух дифференциальных уравнений простым методом Эйлера

```

h := 0.05 ;                Шаг изменения x
d := 0.5 * h                Половина шага изменения x
F1(x,y,z) := (x * y + z) * d  Задание уравнения F1*d
F2(x,y,z) := (x + y * z) * d  Задание уравнения F2*d
F11(x,y,z) := 2 * F1(x + d, y + F1(x,y,z), z + F2(x,y,z))
F22(x,y,z) := 2 * F2(x + d, y + F1(x,y,z), z + F2(x,y,z))

N := 10                     Число шагов
i := 0 .. N                 Задание цикла

```

```


$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

Задание начальных условий

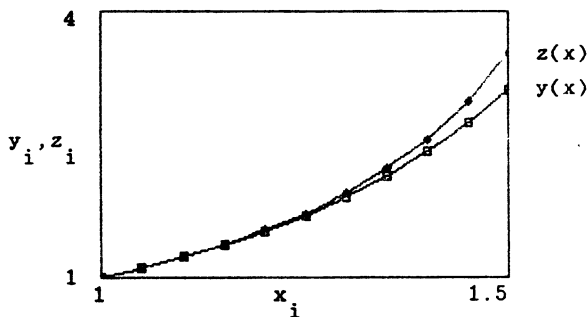
```

Итерационные уравнения в векторной форме

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ z_{i+1} \end{bmatrix} := \begin{bmatrix} x_i + h \\ y_i + F11[x_i, y_i, z_i] \\ z_i + F22[x_i, y_i, z_i] \end{bmatrix}$$

Результаты вычислений

$x = \begin{bmatrix} 1.051 \\ 1.111 \\ 1.152 \\ 1.227 \\ 1.255 \\ 1.333 \\ 1.355 \\ 1.414 \\ 1.455 \\ 1.511 \\ 1.555 \end{bmatrix}$	$y = \begin{bmatrix} 1.106 \\ 1.227 \\ 1.364 \\ 1.521 \\ 1.717 \\ 1.906 \\ 2.143 \\ 2.413 \\ 2.739 \\ 3.111 \\ 3.566 \end{bmatrix}$	$z = \begin{bmatrix} 1.106 \\ 1.228 \\ 1.368 \\ 1.531 \\ 1.721 \\ 1.949 \\ 2.224 \\ 2.553 \\ 2.937 \\ 3.331 \\ 3.746 \end{bmatrix}$
---	---	---



Док. 5.8. Решение системы из двух дифференциальных уравнений модифицированным методом Эйлера

В принципе они решаются теми же методами, что и одно дифференциальное уравнение. Естественно, при этом возрастает число итерационных формул и требуются большие объемы памяти под массивы.

Например, простой метод Эйлера для системы дифференциальных уравнений реализуется с помощью следующей системы итерационных уравнений:

$$x_{i+1} = x_i + h,$$

$$y_{i+1} = y_i + h \cdot f1(x_i, y_i, z_i),$$

$$z_{i+1} = z_i + h \cdot f2(x_i, y_i, z_i \dots),$$

.....

Реализация этого метода для системы из двух дифференциальных уравнений

$$dy/dx = x \cdot y + z,$$

$$dz/dx = x + y \cdot z$$

приведена в док. 5.7 для  $h=0.05$ ,  $x_0=y_0=z_0=1$ .

Как и при решении одного уравнения, значительно меньшую погрешность обеспечивает модифицированный метод Эйлера. Решение этим методом системы из двух дифференциальных уравнений иллюстрирует док. 5.8.

В принципе возможно решение систем дифференциальных уравнений в среде системы MathCAD и методом Рунге—Кутты четвертого порядка. Однако в этом случае решение становится громоздким и его лучше реализовать с помощью типовых программ на языках высокого уровня, например Бейсике или Паскале. Подобные программы описаны в [2, 3].

## 5.6. Вычисление специальных функций

Большинство специальных функций являются решениями определенных типов дифференциальных уравнений, которые нельзя представить в виде элементарных функций. Как отмечалось, некоторые из специальных функций (Бесселя, гамма-функция, интеграл вероятности) встроены в систему MathCAD. Другие могут быть вычислены численными методами, алгоритмы которых описаны в [1, 2].

Многие функции имеют представления в виде определенных интегралов. Наличие в системе MathCAD оператора вычисления определенного интеграла позволяет легко вычислить значения подобных функций.

Документ 5.9 демонстрирует вычисление ряда специальных функций: интегрального синуса  $Si(x)$ , интегрального косинуса  $Si(x)$ , интегралов Френеля  $C(x)$  и  $S(x)$  и др. Для всех этих функций характерно, что пределы интегрирования конечны. Рекомендуется задавать функцию таким образом, чтобы она не давала особенностей в пределах области интегрирования (см. пример вычисления интегрального синуса). В последнем случае функция  $\sin(x)/x$  имеет особенность при  $x=0$ , т. е. при нижнем пределе интегрирования. С помощью функции  $if$  эта особенность исключена. Следует отметить, что поскольку система MathCAD трактует бесконечность как конечное, но очень большое число, то вычисления возможны и без этого приема, но время вычислений окажется намного больше.

Если функция представима несобственным интегралом, например имеющим один из пределов, равный бесконечности, то ее вычисление по интегральному представлению становится либо ненадежным, либо вообще невозможным. Замена бесконечного предела большим конечным возможна далеко не всегда и требует довольно сложных оценок погрешности, возникающей от такой подмены. Поэтому некоторые функции целесообразно вычислять другими методами, например по разложениям в ряд, с применением рекуррентных и асимптотических выражений.

В качестве наглядного примера рассмотрим вычисление интегральной показательной функции  $E_n(x)$ , имеющей следующее интегральное представление:

$$E_n(x) = \int_1^{\infty} (e^{-x \cdot t} / t^n) \cdot dt \quad (n = 0, 1, 2, \dots).$$



Интегральный синус

$$\text{Si}(x) := \int_0^x \frac{\sin(t)}{t} dt$$

$$\text{Si}(0.2) = 0.199$$

$$\text{Si}(1) = 0.945432$$

Интегральный косинус

$$\text{Ci}(x) := 0.57721566 + \ln(x) + \int_0^x \frac{\cos(t) - 1}{t} dt$$

$$\text{Ci}(1) = 0.337$$

$$\text{Ci}(10) = -0.045$$

Интегралы Френеля

$$\text{C}(x) := \int_0^x \cos[1.571 t^2] dt$$

$$\text{C}(0.5) = 0.492 \quad \text{C}(2) = 0.488$$

$$\text{S}(x) := \int_0^x \sin[1.571 t^2] dt$$

$$\text{S}(0.5) = 0.065 \quad \text{S}(2) = 0.343$$

Полные эллиптические интегралы

$$\text{K}(m) := \int_0^{1.571} [1 - m \sin(t)^2]^{-\frac{1}{2}} dt$$

$$\text{K}(0.1) = 1.613 \quad \text{K}(0.5) = 1.854$$

$$\text{E}(m) := \int_0^{1.571} [1 - m \sin(t)^2]^{\frac{1}{2}} dt$$

$$\text{E}(0.1) = 1.531 \quad \text{E}(0.5) = 1.351$$

Неполная Гамма-функция

$$g(a, x) := \int_0^x e^{-t} t^{a-1} dt$$

$$g(1, 0.5) = 0.393$$

$$g(2, 8) = 0.997$$

Функция Дебая

$$\text{D}(n, x) := \int_0^x n \frac{t^n}{x^n [e^t - 1]} dt$$

$$\text{D}(4, 1) = 0.655$$

$$\text{D}(4, 10) = 0.009674$$

Док. 5.9. Интегрирование специальных функций

Вычисление функции  $E_0(x)$  по аналитическому выражению

$$E_0(x) := \frac{\exp(-x)}{x} \quad E_0(0.5) = 1.213061$$

Вычисление функции  $E_1(x)$  суммированием членов ряда

$$i := 1 \dots 10$$

$$E_1(x) := -0.5772116 - \ln(x) - \sum_{i=1}^n (-1)^i \cdot \frac{x^i}{i \cdot i!}$$

$$E_1(0.5) = 0.559778$$

Вычисление функции  $E_n(x)$  для ряда  $n$  по рекуррентной формуле

$$x := 0.5 \quad n := 1 \dots 10$$

$$E_0 := E_0(x) \quad E_1 := E_1(x)$$

$$E_{n+1} := \begin{bmatrix} 1 \\ -n \end{bmatrix} \cdot [\exp(-x) - x E_n]$$

n	E <sub>n</sub>
1	1.213061
2	0.559778
3	0.326642
4	0.221606
5	0.165243
6	0.130977
7	0.108208
8	0.092071
9	0.080071
10	0.070812
11	0.063458
12	0.05748

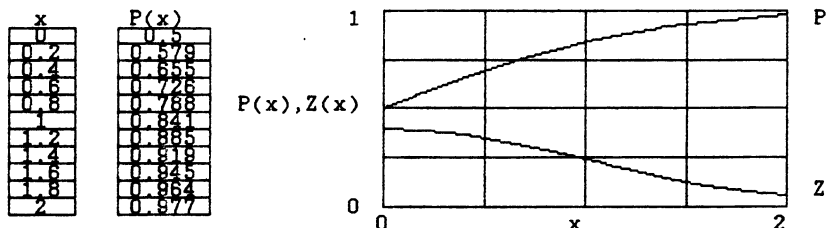
Вычисление функции  $P(x)$  по ее полиномиальной аппроксимации

$$Z(x) := 0.3989423 \cdot \exp\left[\frac{-x^2}{2}\right] \quad t(x) := \frac{1}{1 + 0.33267 \cdot x}$$

$$a_1 := .4361836 \quad a_2 := -.1201676 \quad a_3 := 0.93729$$

$$P(x) := 1 - Z(x) \cdot [a_1 \cdot t(x) + a_2 \cdot t(x)^2 + a_3 \cdot t(x)^3]$$

$$x := 0, 0.2 \dots 2$$



Док. 5.10. Вычисление специальных функций различными методами

Здесь для функции  $E_0(x)$  известно простое аналитическое выражение

$$E_0(x) = e^{-x} / x.$$

Функция  $E_1(x)$  вычисляется с применением следующего ряда:

$$E_1(x) = -\gamma - \ln(x) - \sum_{i=1}^{\infty} (-1)^i \cdot x^i / (i \cdot i!),$$

где  $\gamma = 0.577215665...$  — постоянная Эйлера.

Наконец, функции  $E_n(x)$  при  $n$  от 2 и выше могут вычисляться по рекуррентному соотношению

$$E_{n+1}(x) = [e^x - x \cdot E_n(x)] / n.$$

Все эти способы реализованы в примере, приведенном в док. 5.10. Естественно, что бесконечное число членов ряда при вычислении функции  $E_1(x)$  заменено конечным. Поскольку данный ряд сходится быстро, взято 10 членов ряда.

Во второй части документа рассмотрен еще один полезный способ вычисления специальных функций — с применением их аппроксимаций многочленами. Поскольку коэффициенты многочленов вычислены заранее, этот способ нередко позволяет получить наименьшее время вычислений при вполне приемлемой для практики погрешности. В док. 5.10 приведено вычисление функций  $Z(x)$  и  $P(x)$ . Результаты вычислений представлены в табличном и графическом виде.

## 5.7. Операции с комплексными числами

Как отмечалось в § 3.3, подавляющее большинство функций и операторов системы MathCAD может работать как с действительными, так и с комплексными аргументами и операндами. Это сильно облегчает многие виды расчетов, например электрических цепей символическим методом.

Документ 5.11 иллюстрирует различные операции с комплексными числами. Первая часть документа показывает проведение обычных математических операций с комплексными числами. Вторая часть демонстрирует простоту решения системы линейных уравнений с комплексными коэффициентами. Как известно, такими системами описываются линейные электрические цепи на переменном токе.

Приведенная система уравнений описывает цепь в виде параллельного LC-контура, подключенного к источнику переменной ЭДС  $E = 5 + 0i$  В с внутренним сопротивлением  $Z_E = 10 + 0i$  Ом. Если на некоторой частоте реактивное сопротивление индуктивности  $Z_L = 0 + 200i$ , а сопротивление емкости  $Z_C = 0 - 30i$ , то для токов  $I_1$  в цепи источника и  $I_2$  в цепи контура можно записать следующую систему уравнений:

$$\begin{bmatrix} 10 + 200i & -200i \\ -200i & (200 - 30)i \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 0 \end{bmatrix}.$$

Решение этой системы отражено в док. 5.11. Система задается в матричной форме  $AX=B$ , где  $A$  — матрица коэффициентов;  $X$  — вектор неизвестных (токов) и  $B$  — вектор свободных членов. Решение ищется с применением формулы  $X = A^{-1} \cdot B$  где  $A^{-1}$  — инвертированная матрица коэффициентов.

С комплексными операндами могут работать и наиболее сложные операторы системы MathCAD — численного дифференцирования и интегрирования. Эта возможность иллюстрируется в док. 5.11 на примере вычисления комплексного кругового интеграла

$$\int_0^{2\pi} f[z(t)] \cdot [dz(t)/dt] \cdot dt = 2\pi i.$$

# Элементарные операции с комплексными числами

$i := \sqrt{-1}$	Задание мнимой единицы
$Z1 := 7 + 5i$	Задание двух комплексных чисел $Z1$ и $Z2$
$Z2 := -4 - 3i$	
$Z := Z1 + Z2$	Сложение чисел $Z1$ и $Z2$
$Z = 3 + 2i$	
$\text{Re}(Z) = 3$	Выделение действительной и мнимой частей числа $Z$
$\text{Im}(Z) = 2$	
$ Z  = 3.606$	Вычисление модуля и аргумента числа $Z$
$\arg(Z) = 0.588$	
$Z1 :=  Z  \cdot e^{i \arg(Z)}$	Представление числа $Z$ в тригонометрической форме
$Z1 = 3 + 2i$	
$\frac{Z1}{Z2} = -0.72 + 0.04i$	Деление числа $Z1$ на $Z2$
$Z1 := \exp(Z)$	Вычисление $\exp(Z)$
$Z1 = -8.359 + 18.264i$	
$\bar{Z} = 3 - 2i$	Получение комплексно-сопряженного числа
$Z^2 = 5 + 12i$	Квадрат комплекс. числа
$x := 0 + 3i$	Вычисление комплексных корней квадратного уравнения
$\text{root}[x^2 + 2 \cdot x + 15, x] = -1 + 3.742i$	
$x := 0 - 3i$	
$\text{root}[x^2 + 2 \cdot x + 15, x] = -1 - 3.742i$	
Решение системы линейных уравнений с компл. коэффициентами	
$A := \begin{bmatrix} 10 + 200i & 0 - 200i \\ 0 - 200i & 0 + 170i \end{bmatrix}$	$B := \begin{bmatrix} 5 + 0i \\ 0 + 0i \end{bmatrix}$
$X := A^{-1} \cdot B$	Решение с помощью матричных операторов
$X = \begin{bmatrix} 0.037 + 0.131i \\ 0.044 + 0.154i \end{bmatrix}$	Вектор решений
Вычисление комплексного кругового интеграла	
$f(z) := \frac{1}{z}$	$z(t) := \cos(t) + i \cdot \sin(t)$
$t_{\text{start}} := 0$	$t_{\text{end}} := 6.2832$
	$\text{TOL} \approx 10^{-6}$
$\int_{t_{\text{start}}}^{t_{\text{end}}} f(z(t)) \left[ \frac{d}{dt} z(t) \right] dt = 6.283i$	

Док. 5.11. Примеры операций с комплексными числами, операторами и функциями

Значение этого интеграла — мнимое число. В ходе его вычисления с комплексными операндами используются операторы вычисления производной и интеграла.

## 5.8. Графики функций в полярной системе координат

Наряду с декартовой системой координат широко используется полярная система координат. В ней каждая точка задается длиной радиуса-вектора  $r$  и его углом  $\theta$ . Связь между декартовыми координатами точки и ее полярными координатами следующая:

$$x = r \cdot \cos(\theta), \quad y = r \cdot \sin(\theta).$$

Хорошо известным применением графиков в полярной системе координат являются фигуры Лиссажу. Обычно их наблюдают с помощью электронного осциллографа, подавая на входы  $X$  и  $Y$  синусоидальные сигналы различных частот. Если эти частоты строго кратны, то получаются неподвижные фигуры, по виду которых можно судить о соотношении частот и фаз. При нестрогой кратности фигуры вращаются.

Для получения фигур Лиссажу с помощью системы MathCAD достаточно вычислять координаты точек по формулам

$$x = \cos(n1 \cdot \omega), \quad y = \sin(n2 \cdot \omega).$$

Документ 5.12 иллюстрирует построение фигур Лиссажу.

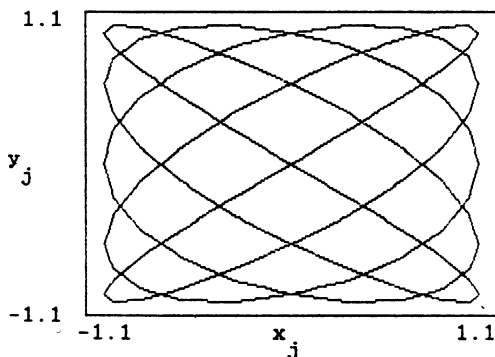
Еще одна форма задания кривых в полярной системе координат — параметрическая. В этом случае задается зависимость длины радиуса-вектора  $r(\omega)$  от круговой частоты  $\omega$  и координаты  $x$  и  $y$  определяются как

$$x = r(\omega) \cdot \cos(\omega), \quad y = r(\omega) \cdot \sin(\omega).$$

```

Число точек графика      N := 100
Кратность частот         n1 := 5      n2 := 4
p1 := 6.283 *  $\frac{n1}{N}$       p2 := 6.283 *  $\frac{n2}{N}$ 
j := 1 .. N + 1
xj := cos(p1 · j)        yj := sin(p2 · j)

```



Меняя кратность частот  $n1$  и  $n2$  и число точек  $N$ , можно получить множество других фигур Лиссажу

Док. 5.12. Построение фигур Лиссажу

```

x(r,w) := r cos(w)  y(r,w) := r sin(w)
w := 0, .1 .. 6.3    r(w) := 5 sin(w) cos(w)

```

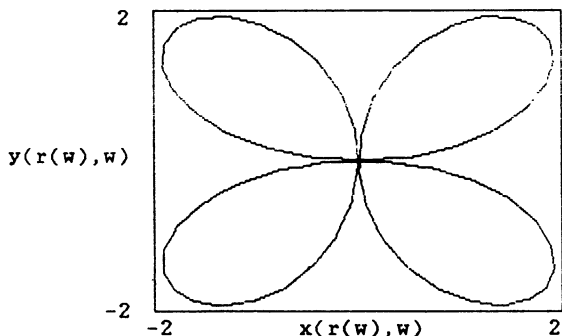


График  
параметрически  
заданной  
функции

Док. 5.13. Построение графика параметрической функции

Вид кривой при изменении  $\omega$  от 0 до  $2\pi$  существенно зависит от характера зависимости.

Документ 5.13 показывает построение кривой, заданной в параметрической форме с зависимостью

$$r(\omega) = 5 \cdot \sin(\omega) \cdot \cos(\omega).$$

Меняя вид функции  $r(\omega)$ , можно построить множество интересных кривых. Частота  $\omega$  обозначена буквой  $w$ .

## 5.9. Графики на комплексной плоскости

Комплексной плоскостью называют плоскость, на которой отображаются комплексные числа. Координатная система этой плоскости имеет ось действительных и ось мнимых чисел, по которым откладываются значения действительной и мнимой частей отображаемого на плоскости комплексного числа. В полярной системе координат точка с параметрами  $r$  и  $\theta$  отображает число

$$Z = r \cdot e^{i\theta}.$$

Можно также записать

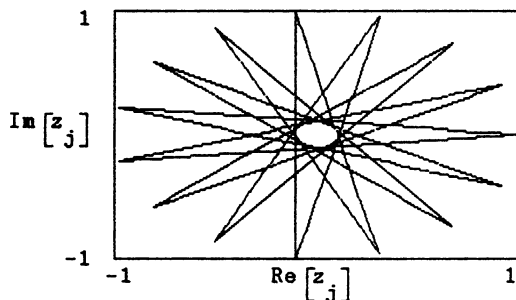
$$Z = r \cdot \cos(\theta) + ir \cdot \sin(\theta) = \operatorname{Re}(Z) + i \operatorname{Im}(Z).$$

Электротехнические приложения представления чисел на комплексной плоскости будут рассмотрены в следующей главе. Здесь мы приведем два примера из математики, достаточно поучительных и интересных.

Документ 5.14 показывает построение на комплексной плоскости  $N$ -лучевой звезды, а док. 5.15 — проведение комфортных преобразований фигуры на комплексной плоскости. В результате прямоугольник преобразуется в ромб.

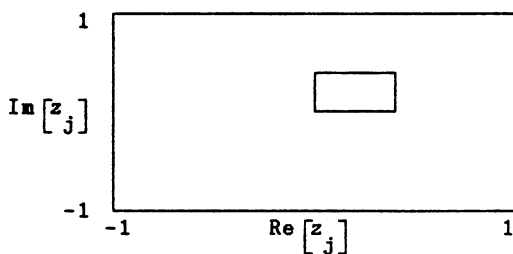
К новым разделам математики относится фрактальная геометрия. Не пытайтесь дать даже понятие о ней, ограничимся приведением док. 5.16, иллюстрирующего построение фрактальных поверхностей двух типов. Такие поверхности порой удивительно напоминают очертания галактик, туманностей, облаков, снежинок и других физических объектов, состоящих из множества частиц, обнаруживающих при своем случайном блуждании свойства к группированию.

Число лучей  $\rightarrow$   $N := 15$   $j := 0 \dots N$   
 $C := \text{floor}\left[\frac{N}{2}\right] + 1$   $z_j := \exp\left[i \cdot \left[\frac{C}{N}\right] \cdot 6.283 \cdot j\right]$

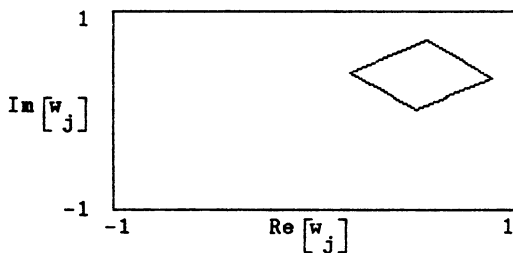


Док. 5.14. Построение N-лучевой звезды, вписанной в окружность

$z_0 := 0$   $z_1 := .4$   $z_2 := .4 + .4i$   $z_3 := .4i$   $z_4 := 0$   
 $j := 0 \dots 4$   
 $p := \frac{e^{.7i}}{.8}$   $c := .5$   $w_j := p \cdot z_j + c$



Фигура до  
преобразования



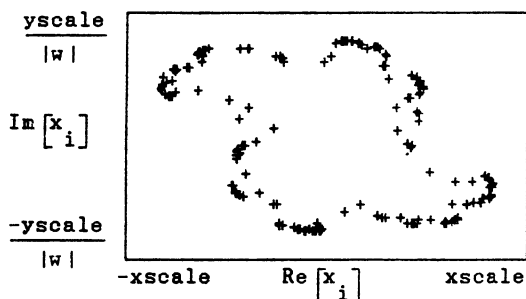
Фигура после  
преобразования

Док. 5.15. Преобразование фигуры на комплексной плоскости

```

pick(x) ≡ 2 floor(rnd(2)) - 1      returns +1 or -1.
f(x) := x2 + w ◻      w := 0.31 + 0.4 i
transient := 10      iterations := 250
i := 1 ..transient + iterations
x1 := 0
      xi+1 := pick(1) · √(xi - w)
i := transient ..iterations      xscale := 1.2
                                   yscale := 0.75

```



Новый пример для λ-формы      f(x) := λ x (1 - x) ◻

λ := 3

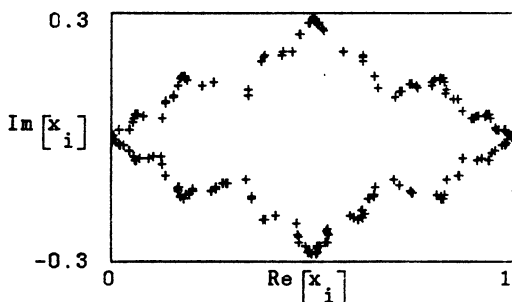
i := 1 ..(transient + iterations)

x<sub>i</sub> := 0

$$x_{i+1} := \frac{1 + \text{pick}(1) \cdot \sqrt{1 - \frac{4x_i}{\lambda}}}{2}$$

i := transient ..iterations

xscale := 1.1  
yscale := 1.0



Док. 5.16. Построение фрактальных кривых второго порядка



Элемент А распадается в элементе В

gm  $\equiv$  1M                      yr  $\equiv$  1T

Начальные условия

Масса элемента А                      A0 := 100 gm

Время жизни элемента А              thalfA := 10 yr

Время жизни элемента В              thalfB := 100 yr

Отношение атомарных масс          p := 1.1

Вычисление постоянных распада

$$kA := \frac{\ln(2)}{thalfA} \quad kB := \frac{\ln(2)}{thalfB}$$

Временные функции распада

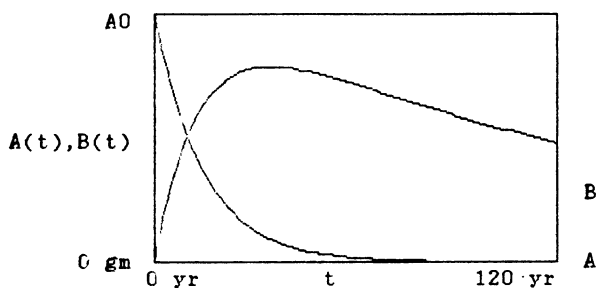
$$A(t) := A0 \cdot e^{-(kA \cdot t)}$$

$$B(t) := -p \cdot A0 \cdot e^{-(kB \cdot t)} \left[ e^{-(kA \cdot t)} - 1 \right]$$

Зависимость массы А и В от времени

t := 0 yr, 2 yr .. 120 yr

t2 := 0 yr, 10 yr .. 120 yr

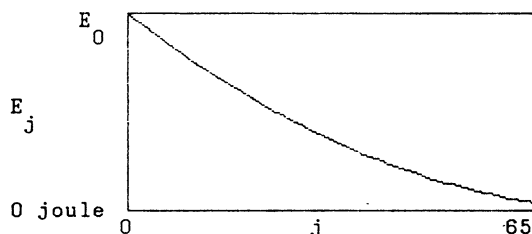


t2	A(t2)	B(t2)
yr	gm	gm
0	100	0
10	50	51.3
20	25	78.2
30	12.5	78.2
40	6.3	78.2
50	3.1	75.4
60	1.6	71.4
70	0.8	67.2
80	0.4	62.9
90	0.2	58.8
100	0.1	54.9
110	0	51.3
120	0	47.9

Док. 5.17. Моделирование радиоактивного распада

$M := 1500 \text{ lb}$       Масса автомобиля  
 $R_w := 14 \text{ in}$       Радиус колеса  
 $R_d := 10 \text{ in}$       Радиус тормозного диска  
 $F_b := 100 \text{ lbf}$       Тормозящая сила  
 $x := .8$       Коэффициент трения  
 $E_0 := \frac{1}{2} M (55 \text{ mph})^2$       Начальная энергия  
 $dt := 0.8 \text{ sec}$       Шаг времени  
 Расчет потери энергии при торможении  
 $j := 0 \dots 65$

$$E_{j+1} := E_j - \sqrt{2 \frac{j}{M} \frac{R_d}{R_w} F_b x dt}$$



$k := 0.5 \dots 65$

$k$	$E_k$ joule
0	206090
5	181668
10	158776
15	137415
20	117586
25	99289
30	82527
35	67299
40	53608
45	41456
50	30845
55	21779
60	14261
65	8299

Единицы измерения и их преобразования

$m \equiv 1L$        $sec \equiv 1T$        $cm \equiv .01 m$        $hr \equiv 3600 sec$   
 $in \equiv 2.54 cm$        $kg \equiv 1M$        $ft \equiv 12 in$        $mi \equiv 5280 ft$   
 $joule \equiv kg m^2 sec^{-2}$        $lb \equiv \frac{kg}{2.2}$        $watt \equiv \frac{joule}{sec}$   
 $mph \equiv \frac{mi}{hr}$        $kw \equiv 1000 watt$        $lbf \equiv 9.8 m sec^{-2}$        $lb$

Док. 5.18. Расчет потери энергии при торможении автомобиля

Основное уравнение диффузии

$$f(x,t) := \frac{d^2}{dx^2} f(x,t) \quad \Delta$$

Аппроксимация дифференциального уравнения диффузии  
конечно-разностным уравнением

t := 0 .. 9                      ... Время  
x := 15 .. 35                    ... Расстояние  
p := .25                         ... Коэффициент диффузии

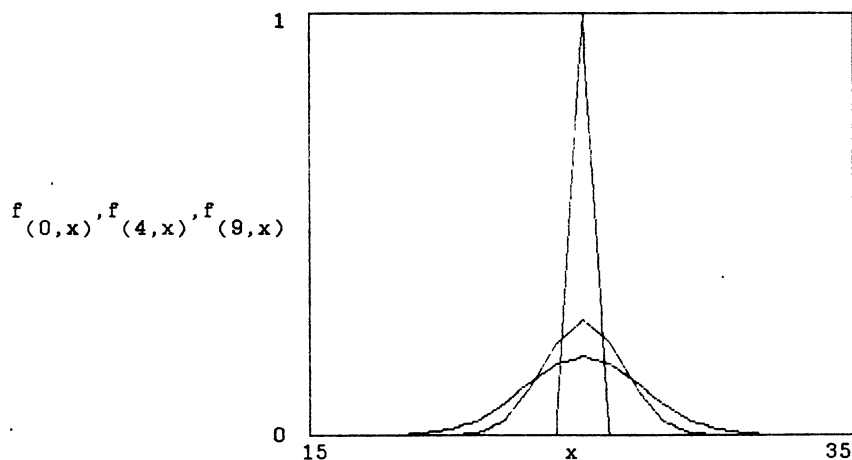
Начальные условия для момента времени t=0

f<sub>0,x</sub> := 0  
f<sub>0,0</sub> := 0  
f<sub>0,50</sub> := 0  
f<sub>0,25</sub> := 1

Конечно-разностное уравнение диффузии

$$f_{t+1,x} := f_{t,x} + p [f_{t,x-1} - 2 \cdot f_{t,x} + f_{t,x+1}]$$

Распределение концентрации для t=0, t=4 и t=9



Док. 5.19. Моделирование процесса диффузии

## 5.10. Примеры физических расчетов

Возможности системы MathCAD в проведении наглядных физических расчетов поистине неограниченны. Мы же ограничимся тремя типовыми примерами, вошедшими в демонстрационные файлы системы [9] и несколько переработанными.

Документ 5.17 иллюстрирует процесс распада радиоактивного элемента А внутри элемента В. Это один из многих случаев, когда процессы описываются аналитическими выражениями, приведенными в тексте документа. Результатом моделирования является графическая зависимость масс элементов от времени.

Документ 5.18 отражает расчет потери энергии при торможении автомобиля. Этот процесс, вообще говоря, описывается дифференциальным уравнением. Но в документе используется числовая модель процесса торможения, реализованная с помощью рекуррентных соотношений. График показывает уменьшение энергии во времени.

Интересный пример численного моделирования процессов диффузии дает док. 5.19. Здесь используется конечно-разностная модель диффузии, которая описывается дифференциальным уравнением второго порядка. Процесс диффузии является функцией двух параметров — расстояния  $x$  и времени диффузии  $t$ . Результатом моделирования являются графики распределения концентрации диффундирующих частиц в зависимости от расстояния  $x$  для трех моментов времени  $t$ .

## 6. РАСЧЕТ ЭЛЕКТРИЧЕСКИХ И ЭЛЕКТРОННЫХ ЦЕПЕЙ И УСТРОЙСТВ

### 6.1. Простейшие электротехнические расчеты

Система MathCAD удобна для составления пакетов программ простых электротехнических расчетов. Со временем из них можно создать библиотеку расчетов на «все случаи жизни». Надеемся, что приведенные в главе документы послужат основой этой библиотеки.

Документ 6.1 содержит ряд простых расчетов резистивных цепей: последовательного и параллельного соединения резисторов в общем виде, преобразования звезды в треугольник и треугольника в звезду, а также расчет передачи энергии постоянного тока от источника с конечным выходным сопротивлением  $r$  в нагрузку  $R$  и расчет цепей на переменном токе.

Как неоднократно отмечалось, система MathCAD позволяет разнообразить результаты графиками. Как пример в документе приведены графические зависимости мощности, отдаваемой источником ЭДС и рассеиваемой нагрузкой  $R$ . У последней хорошо виден экстремум, соответствующий  $R = r$ . Приведены также графики зависимости модуля и фазы полного сопротивления последовательно-резонансного контура от частоты  $f$ , демонстрирующие эффект резонанса.

Большинство серьезных расчетов линейных цепей выполняется методами контурных токов или узловых потенциалов. Эти методы хорошо известны и приводят к необходимости решения систем линейных уравнений с действительными (на постоянном токе) или комплексными (на переменном токе при наличии реактивных элементов) коэффициентами (см., например, док. 5.11, содержащий решение системы уравнений с комплексными коэффициентами).

К простым можно отнести и расчеты Т- и П-образных аттенуаторов. Документ 6.2 иллюстрирует расчет ослабления (в децибелах), вносимого аттенуатором при включении их между нагрузкой и источником ЭДС (в режиме согласования), а также расчет резисторов аттенуаторов по заданному безразмерному коэффициенту ослабления мощности  $N$ . Во всех приведенных примерах исходные данные можно легко изменить под нужные пользователю и тут же получить для них новые результаты.

Последовательное соединение резисторов

```

      R1  R2  R3
    --www--www--www--
R1 := 10 Ом   R2 := 20 Ом   R3 := 30 Ом
i := 1 .. 3
R0 :=  $\sum_i R_i$    R0 = 60

```

Параллельное соединение резисторов

```

      R1
    ---
    ---www---
      R2
    ---www---
      R3
    ---www---
j := 1 .. 3
R :=  $\frac{1}{\sum_j \frac{1}{R_j}}$    R = 5.455

```

Преобразование звезды R1, R2, R3 в треугольник

```

R1 := 100   R2 := 150   R3 := 220
R := R1 R2 + R2 R3 + R3 R1

```

```

R12 :=  $\frac{R}{R3}$    R23 :=  $\frac{R}{R1}$    R31 :=  $\frac{R}{R2}$ 
R12 = 318.182   R23 = 700   R31 = 466.667

```

Преобразование треугольника R12, R23, R31 в звезду

```

R12 := 47   R23 := 82   R31 := 12
R := R12 + R23 + R31

```

```

R1 := R31  $\cdot \frac{R12}{R}$    R2 := R12  $\cdot \frac{R23}{R}$    R3 := R23  $\cdot \frac{R31}{R}$ 
R1 = 4   R2 = 27.333   R3 = 6.979

```

Передача энергии от источника напряжения к нагрузке

```

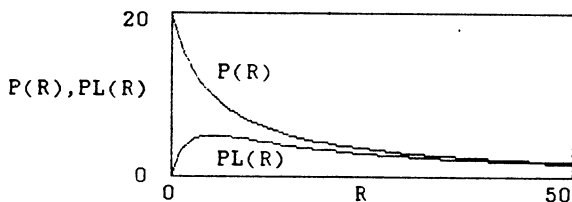
E := 10   r := 5   параметра источника питания
I(R) :=  $\frac{E}{r + R}$    Ток в цепи нагрузки R

```

P(R) := E I(R) <sup>2</sup> Мощность, отдаваемая источником

PL(R) := R I(R) <sup>2</sup> Мощность в нагрузке

R := 0 .. 50



Док. 6.1. Примеры простейших электротехнических расчетов (начало)

Полное сопротивление RC-цепи

<-----R-----C----->  
 <---www-----||----->

R := 100 Ом

C :=  $10 \cdot 10^{-6}$  Ф

f := 50 Гц

i :=  $\sqrt{-1}$  Задание мнимой единицы pi := 3.141

$Z := R + \frac{1}{i \cdot 2 \cdot \pi \cdot f \cdot C}$

Z = 100 - 318.37i

Полное сопротивление последовательного LC-контура

<-----r-----C-----L----->  
 <---www-----||-----UUU----->

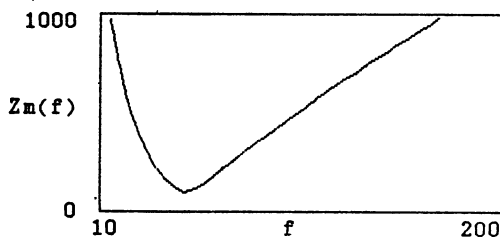
r := 100 Ом C :=  $10 \cdot 10^{-6}$  Ф L := 1 Гн

$Z(f) := r + \left[ i \cdot 2 \cdot \pi \cdot f \cdot L + \frac{1}{i \cdot 2 \cdot \pi \cdot f \cdot C} \right]$

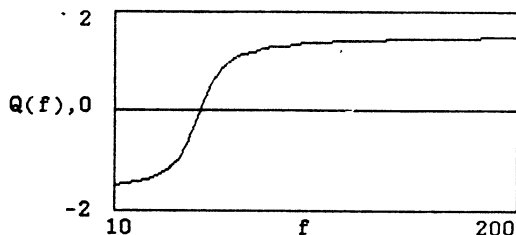
$Z_m(f) := \sqrt{\text{Re}(Z(f))^2 + \text{Im}(Z(f))^2}$  Модуль Z

$Q(f) := \arg(Z(f))$  Фаза Z

f := 10, 15 .. 200



Зависимость  
модуля Z от  
от частоты f



Зависимость  
фазы Z от  
от частоты f

Док. 6.1. Примеры простейших электротехнических расчетов (окончание)

1. Расчет ослабления Т-образного аттенюатора

$$R1 := 75 \quad R2 := 38 \quad R3 := 19 \quad Z0 := 100 \quad Z1 := 50$$

$$A := 1 + \frac{Z0}{R1} \quad B := 1 + \frac{Z1}{R3} \quad C := Z0 + Z1$$

$$AT := 20 \cdot \log \left[ \frac{\left[ 1 + A \frac{R1}{R2} \right] \cdot B \cdot R3 + A \cdot R1}{C} \right]$$

$$AT = 11.469 \text{ (ДБ)}$$

2. Расчет ослабления П-образного аттенюатора

$$AP := 20 \log \left[ \frac{\left[ 1 + A \frac{R2}{Z0} \right] B \cdot Z0 + A \cdot Z1}{C} \right]$$

$$AP = 14.56 \text{ (ДБ)}$$

3. Расчет резисторов П-образного аттенюатора по заданному ослаблению мощности N ( безразм. )

$$N := 100 \quad RG := 100 \quad RL := 50$$

$$R2 := \left[ \frac{2}{N - 1} \sqrt{\frac{N}{RG \cdot RL}} \right]^{-1} \quad R1 := \left[ \left[ \left[ \frac{1}{RG} \right] \frac{N + 1}{N - 1} \right] - \frac{1}{R2} \right]^{-1}$$

$$R3 := \left[ \left[ \left[ \frac{1}{RL} \right] \frac{N + 1}{N - 1} - \left[ \frac{1}{R2} \right] \right] \right]^{-1}$$

$$R1 = 136.147 \quad R2 = 350.018 \quad R3 = 56.99$$

4. Расчет резисторов Т-образного аттенюатора по заданному ослаблению мощности N ( безразм. )

$$N := 50 \quad RG := 200 \quad RL := 100$$

$$R2 := 2 \frac{\sqrt{N \cdot RG \cdot RL}}{N - 1} \quad R1 := RG \frac{N + 1}{N - 1} - R2$$

$$R3 := RL \frac{N + 1}{N - 1} - R2$$

$$R1 = 167.347 \quad R2 = 40.816 \quad R3 = 63.265$$

Док. 6.2. Расчет резистивных аттенюаторов

$$C := 1000 \cdot 10^{-12}$$

$$R := 100000$$

$$PI := 3.14159$$

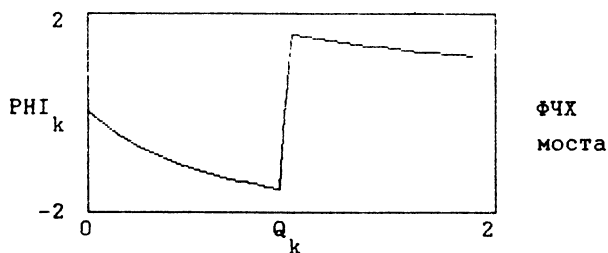
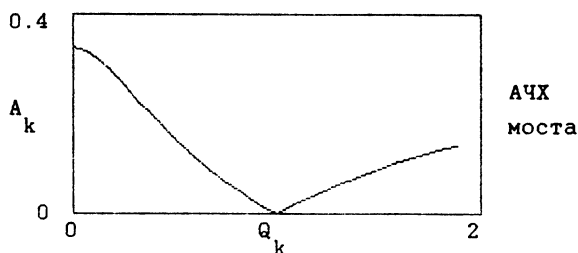
$$f_0 := \frac{1}{2 \cdot PI \cdot R \cdot C} \quad f_0 = 1.592 \cdot 10^3 \quad \text{Частота подавления}$$

$$k := 0 \dots 30$$

$$Q_k := k \cdot \frac{100}{f_0}$$

$$A_k := \frac{|1 - Q_k^2|}{3 \sqrt{[1 - Q_k^2]^2 + 9 \cdot Q_k^2}} \quad \text{Отсчеты АЧХ}$$

$$PHI_k := \text{atan} \left[ 3 \cdot \frac{Q_k}{Q_k^2 - 1} \right] \quad \text{Отсчеты фазы}$$



Док. 6.3. Расчет АЧХ и ФЧХ RC-моста Винна—Робинсона

Документ 6.3 иллюстрирует расчет по известным формулам амплитудно-частотной и фазо-частотной характеристик (АЧХ и ФЧХ) моста Винна—Робинсона. Представлена и конечная цель расчетов — графики АЧХ и ФЧХ данной схемы.



## 6.2. Расчет индуктивных элементов

Любая электрическая цепь обладает индуктивностью — либо паразитной, либо специально созданной. Это может быть отрезок провода или виток, проволочная спираль или специальный индуктивный элемент. Расчет индуктивности для

1. Индуктивность отрезка круглого провода  
 $d := 0.1$  Диаметр провода в см  
 $l := 1$  Длина провода в см

$$L := 2 \cdot l \cdot \left[ \operatorname{asinh} \left[ \frac{l}{d} \right] + \left[ \frac{d}{l} \right] - \sqrt{\left[ \frac{d}{l} \right]^2 + 1} \right]$$

$L = 4.186$  Индуктивность в нГн

2. Индуктивность отрезка провода прямоугольного сечения  
 $b := 0.1$  Ширина проводника в см  
 $t := 0.001$  Толщина проводника в см  
 $l := 1$  Длина провода в см

$$L := 2 \cdot l \cdot \left[ \ln \left[ 2 \frac{l}{b + t} \right] + 0.447 \cdot \frac{t + b}{2 \cdot l} + \frac{1}{2} \right]$$

$L = 7.017$  Индуктивность в нГн

3. Индуктивность одного круглого витка из круглого провода  
 $D := 2$  Средний диаметр витка в см  
 $d := 0.1$  Диаметр провода в см

$$L := 6.283 \cdot D \cdot \left[ \ln \left[ 8 \frac{D}{d} \right] - 1.75 \right]$$

$L = 41.784$  Индуктивность в нГн

4. Индуктивность одиночного тонкопленочного круглого витка  
 $D := 1$  Средний диаметр в см  
 $b := 0.01$  Ширина полоски в см

$$L := 6.283 \cdot D \cdot \left[ \ln \left[ 6.283 \cdot \frac{D}{b} \right] - 2.451 \right]$$

$L = 25.082$  Индуктивность в нГн

5. Индуктивность одного тонкопленочного квадратного витка  
 $l := 1$  Сторона квадрата в см  
 $b := 0.01$  Ширина полоски в см

$$L := 8 \cdot l \cdot \left[ \ln \left[ 8 \cdot \frac{l}{b} \right] - 2.853 \right]$$

$L = 30.653$  Индуктивность в нГн

6. Индуктивность однослойной круглой катушки  
 $R := 1$  Средний радиус в см  
 $l := 2$  Длина катушки в см  
 $w := 10$  Число витков

$$L := 12.57 \cdot w^2 \cdot R \cdot \left[ \ln \left[ 1 + \frac{3.14}{\frac{l}{R}} \right] + \frac{1}{2.3 + 1.6 \cdot \left[ \frac{l}{R} \right] + 0.44 \cdot \left[ \frac{l}{R} \right]^2} \right]$$

$L = 1.36 \cdot 10^3$  Индуктивность в нГн

Док. 6.4. Расчет типовых индуктивных элементов (начало)

7. Индуктивность однослойной катушки квадратного сечения  
 $a := 1$  Половина длины стороны квадрата в см  
 $l := 2$  Длина катушки в см  
 $w := 40$  Число витков

$$L := 16 a w^2 \left[ \ln \left[ 1 + \frac{3.141}{\frac{l}{a}} \right] + \frac{1}{3.64 + 2 \cdot \left[ \frac{l}{a} \right] + 0.51 \cdot \left[ \frac{l}{a} \right]^2} \right]$$

$$L = 2.681 \cdot 10^4 \quad \text{Индуктивность в нГн}$$

8. Индуктивность однослойной тороидальной катушки  
 круглого сечения  
 $D := 3$  Средний диаметр тора в см  
 $d := 1$  Диаметр сечения в см  
 $w := 300$  Число витков

$$L := 6.283 w^2 \left[ D - \sqrt{D^2 - d^2} \right]$$

$$L = 9.702 \cdot 10^4 \quad \text{Индуктивность в нГн}$$

9. Индуктивность однослойной тороидальной катушки  
 квадратного сечения  
 $D1 := 1$  Внутренний диаметр тора в см  
 $D2 := 3$  Внешний диаметр тора в см  
 $h := 1$  Высота тора в см  
 $w := 100$  Число витков

$$L := 2 \cdot h \cdot w^2 \ln \left[ \frac{D2}{D1} \right]$$

$$L = 2.197 \cdot 10^4 \quad \text{Индуктивность в нГн}$$

10. Индуктивность круглой тонкопленочной катушки  
 $A := 1$  Диаметр внешнего витка в см  
 $a := 0.5$  Диаметр внутреннего витка в см  
 $w := 5$  Число витков

$$L := 2.33 w^{\frac{5}{3}} (A + a) \cdot \ln \left[ 4 \cdot \frac{A + a}{A - a} \right]$$

$$L = 126.972 \quad \text{Индуктивность в нГн}$$

11. Индуктивность квадратной тонкопленочной катушки  
 $A := 1$  Ширина внешнего витка в см  
 $a := 0.5$  Ширина внутреннего витка в см  
 $w := 5$  Число витков

$$L(w) := 12.05 w^{\frac{5}{3}} (A + a) \cdot \ln \left[ 8 \cdot \frac{A + a}{A - a} \right]$$

$$L(w) = 839.826 \quad \text{Индуктивность в нГн}$$

Нахождение  $w$  по заданной индуктивности  $L0$

$$L0 := 2000 \quad \text{Заданная индуктивность } L$$

$$\text{root}(L(w) - L0, w) = 8.415 \quad \text{Расчет } w$$

Док. 6.4. Расчет типовых индуктивных элементов (окончание)

Индуктивность многослойной тороидальной круглой катушки

$D := 4$  Диаметр тора в см

$d := 0.8$  Диаметр сечения тора в см

$w := 100$  Число витков

$$L := 6.283 D w^2 \left[ \ln \left[ 3 \frac{D}{d} \right] - 1.75 \right]$$

$$L = 4.873 \cdot 10^5 \quad \text{нГн}$$

Индуктивность многослойной цилиндрической катушки

$D := 2.5$  Средний диаметр в см

$l := 1$  Длина катушки в см

$c := 1$  Толщина обмотки в см

$w := 100$  Число витков

$$L := 78.54 \cdot D^2 \frac{w^2}{3 \cdot D + 9 l + 10 \cdot c}$$

$$L = 1.852 \cdot 10^5 \quad \text{нГн}$$

Индуктивность тороидальной катушки на ферромагнитном сердечнике с квадратным сечением

$D := 4$  Внешний диаметр кольца в см

$d := 2$  Внутренний диаметр кольца в см

$h := 1$  Высота кольца в см

$\mu := 1000$  Магнитная проницаемость

$w := 120$  Количество витков

$$L := \text{if} \left[ \frac{D}{d} < 1.75, 4 \cdot \mu (D - d) \frac{w^2}{D + d}, 2 w^2 \mu h \ln \left[ \frac{D}{d} \right] \right]$$

$$L = 1.996 \cdot 10^7 \quad \text{нГн}$$

Индуктивность катушек на П- и Ш-образных сердечниках

$F_s := 1$  Сечение сердечника в квадр. см

$d := 0.05$  Толщина зазора в см

$l := 10$  Длина магнитной линии сердечника

$\mu := 1000$  Магнитная проницаемость

$a := 1.5$  Поправочный коэффициент

$w := 400$  Число витков

Док. 6.5. Расчет индуктивности многослойных катушек (начало)

$$L := 12.566 \cdot \mu \cdot F_s \cdot \frac{w^2}{1 \cdot \left[ 1 + \mu \frac{d}{a \cdot l} \right]}$$

$$L = 4.64 \cdot 10^7 \quad \text{нГн}$$

Индуктивность катушки в броневом сердечнике

D1 := 1      Внутренний диаметр кольца сердечника в см

D2 := 2      Внутренний диаметр обмотки в см

D3 := 4      Внешний диаметр обмотки в см

D4 := 5      Внешний диаметр кольца сердечника

h1 := 3      Высота обмотки в см

h2 := 4      Высота сердечника в см

$\mu$  := 10      Магнитная проницаемость сердечника

w := 100      Число витков

$$A := (h1 + h2) \cdot \left[ \frac{1}{D4^2 - D3^2} + \frac{1}{D2^2 - D1^2} \right]$$

$$B := \frac{\ln \left[ \frac{D3 + D4}{D1 + D2} \right]}{h2 - h1}$$

$$L := 19.74 \cdot \mu \cdot \frac{w^2}{A + B}$$

$$L = 4.689 \cdot 10^5$$

Пример вычисления числа витков по заданной индуктивности L0

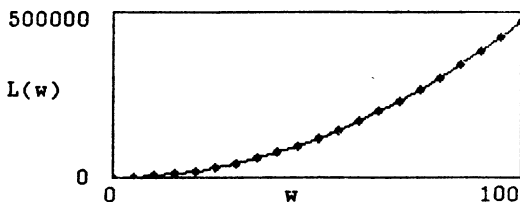
$$L(w) := 19.74 \cdot \mu \cdot \frac{w^2}{A + B}$$

L0 := 500000      Заданная индуктивность в нГн

W := root(L0 - L(w), w)

W = 103.261      Требуемое число витков

w := 0,5 .. 100



Зависимость L  
от числа витков w

Док. 6.5. Расчет индуктивных многослойных катушек (окончание)

многих типов индуктивных элементов несложен и может выполняться по известным аналитическим выражениям.

В док. 6.4 приведены расчеты индуктивности  $L$  для почти полутора десятков наиболее распространенных индуктивных элементов — от отрезков провода до тонкопленочных катушек индуктивности, применяемых в интегральных микросхемах.

Любую формулу для индуктивности можно представить в виде

$$L = f(x, y, z, \dots, w),$$

где  $x, y, z, \dots, w$  — некоторые конструктивные параметры, например  $w$  — обычно число витков катушки.

Система MathCAD позволяет в считанные секунды построить график зависимости индуктивности  $L$  любого индуктивного элемента от значения того или иного параметра. Это очень удобно при проектировании индуктивных элементов.

Нередко на практике нужно вычислить не саму индуктивность, а значение какого-либо параметра, при котором получается необходимая индуктивность  $L$ . Для этого следует выразить индуктивность как функцию от требуемого параметра, например

$$L(w) := f(x, y, z, \dots, w).$$

Фактически для этого в нужную формулу достаточно добавить после  $L$  скобки с аргументом  $(w)$ .

С помощью функции `root` теперь можно найти значение параметра, обеспечивающее получение требуемой индуктивности, например

$$\text{root}(L(w) - L_0, w) =$$

дает значение  $w$ , при котором  $L = L_0$ .

В док. 6.5 приведены расчеты индуктивности  $L$  еще для ряда индуктивных элементов — многослойных катушек и катушек с ферромагнитным сердечником. Такие элементы широко применяются на практике, когда необходимо получить сравнительно большие значения  $L$  — свыше тысяч наногенри.

Более редко встречается необходимость в расчете взаимной индуктивности индуктивных элементов или в расчете специальных типов индуктивных элементов, например дросселей с зазором у ферромагнитного сердечника или трансформаторов. Такие расчеты при необходимости могут выполняться по известным аналитическим выражениям и методикам [1—3].

### 6.3. Расчет емкостных элементов

Емкость, как и индуктивность, принципиально присуща любой электрической цепи. Если во многих случаях паразитной индуктивностью можно пренебречь, то пренебрежение емкостью встречается реже. Расчет емкости для многих емкостных элементов также выполняется по аналитическим выражениям.

Документ 6.6 иллюстрирует расчет емкости  $C$  для ряда наиболее распространенных емкостных элементов — пластинчатого, дискового, шарового и горшкового конденсаторов. Представлены также расчеты емкости шара, круглого диска и одиночного провода в свободном пространстве.

Как и в случае расчета индуктивности, емкость для емкостных элементов можно представить в виде зависимости  $C(x, y, z, \dots)$ , где  $z$  — конструктивные параметры. Если необходимо найти значение какого-либо параметра для заданного значения емкости  $C$ , следует использовать функцию `root`.

На практике емкостные элементы обычно специально не конструируются — используются стандартные конденсаторы. В то же время нередко требуется оценка паразитных емкостей. Ее примерный расчет может базироваться на представлении цепи в виде совокупности конструктивных емкостных элементов, частичные емкости которых вычисляются по аналогии с расчетами, приведенными выше.

Емкость пластинчатого конденсатора  
без учета краевых эффектов

S := 1            Площадь пластин в квадр. см

h := 0.1            Толщина диэлектрика в см

E := 1            Диэлектрическая проницаемость

$$C := 0.0866 E \frac{S}{h} \quad C = 0.866 \quad \text{пФ}$$

Емкость дискового конденсатора с обкладками,  
занимающими всю поверхность диэлектрика

D := 1            Диаметр диска в см

h := 0.1            Толщина диэлектрика в см

E := 4.7            Диэлектрическая проницаемость

$$C := 0.0695 E \frac{D}{h} \quad C = 3.267 \quad \text{пФ}$$

Емкость трубчатого конденсатора с обкладками,  
занимающими всю длину трубки диэлектрика

l := 2            Длина трубки в см

D := 0.5            Наружный диаметр трубки в см

d := 0.4            Внутренний диаметр трубки в см

E := 4.7            Диэлектрическая проницаемость

$$C := 0.241 E \frac{l}{\log \left[ \frac{D}{d} \right]} \quad C = 23.376 \quad \text{пФ}$$

Емкость трубчатого конденсатора с обкладками,  
занимающими часть длины трубки (данные см. выше)

$$C := \frac{0.241 \cdot E \cdot l + 0.08 D}{\log \left[ \frac{D}{d} \right]} \quad C = 23.789 \quad \text{пФ}$$

Емкость трубчатого конденсатора с обкладками  
разной длины, меньшая-1 (данные см. выше)

$$C := 0.241 E \frac{l + 0.17 D}{\log \left[ \frac{D}{d} \right]} \quad C = 24.37 \quad \text{пФ}$$

Емкость шарового полого конденсатора

D := 10            Наружный диаметр шара в см

d := 9            Внутренний диаметр шара в см

E := 4.7            Диэлектрическая проницаемость

$$C := 0.555 E \frac{D}{\frac{D}{d} - 1} \quad C = 234.765 \quad \text{пФ}$$

Док. 6.6. Расчет емкости конденсаторов и емкостных элементов (начало)

Емкость горшкового (цилиндр с полусферой) конденсатора

D := 5 Диаметр наружный в см  
 d := 4 Диаметр внутренний в см  
 l := 10 Длина цилиндра в см  
 E := 4.7 Диэлектрическая проницаемость

$$C := E \cdot \left[ 0.241 \cdot \frac{1}{\log \left[ \frac{D}{d} \right]} + 0.278 \cdot \frac{D}{\frac{D}{d} - 1} \right]$$

C = 143.014 пФ

Емкость шара в свободном пространстве

D := 10 Диаметр шара в см  
 E := 1 Диэлектрическая проницаемость  
 C := 0.555 · E · D C = 5.55 пФ

Емкость между двумя одинаковыми шарами

D := 10 Диаметр шаров в см  
 l := 15 Расстояние между центрами (l > D)  
 E := 4.7 Диэлектрическая проницаемость

$$C := 0.278 \cdot E \cdot D \cdot \left[ 1 + D \cdot \frac{4 \cdot l^2 - D^2}{2 \cdot l \cdot [4 \cdot l^2 - 2 \cdot l \cdot D - D^2]} \right]$$

C = 20.035 пФ

Емкость круглого диска в свободном пространстве

D := 10 Диаметр диска в см  
 h := 1 Толщина диска в см  
 E := 1 Диэлектрическая проницаемость

$$C := 0.353 \cdot E \cdot D \cdot \left[ 1 + 0.637 \cdot \frac{h}{D} \right] C = 3.755 \text{ пФ}$$

Емкость одиночного провода над землей

l := 1000 Длина провода в см  
 d := 0.2 Диаметр провода в см  
 x := 200 Высота подвески в см  
 E := 1 Диэлектрическая проницаемость

$$C := 0.24 \cdot E \cdot \frac{1}{\log \left[ 4 \cdot \frac{x}{d} \right]} C = 66.629 \text{ пФ}$$

Док. 6.6. Расчет емкости конденсаторов и емкостных элементов (окончание)

Расчет параметра  $x=D/d$  для коаксиальной линии ( кабеля )

$D := 0.8$  Внутренний диаметр экрана в см

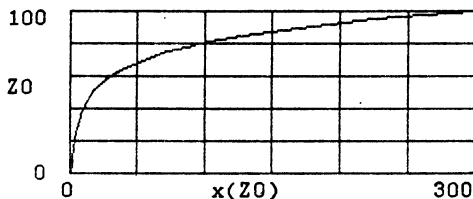
$Z_0 := 50$  Волновое сопротивление линии в Ом

$\epsilon := 11.7$  Диэлектрическая проницаемость

$$x(Z_0) := \exp \left[ Z_0 \frac{\sqrt{\epsilon}}{60} \right] \quad x(Z_0) = 17.295$$

$$d := \frac{D}{x(Z_0)} \quad d = 0.046 \quad \text{Диаметр жилы в см}$$

$Z_0 := 0,5 \dots 100$



Зависимость  $Z_0$  от параметра  $x$

Расчет параметра  $x=D/d$  для линии в виде круглого проводника, расположенного над проводящей плоскостью

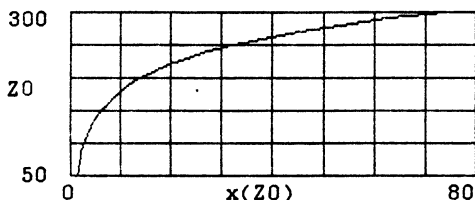
$\epsilon := 1$  Диэлектрическая проницаемость

$Z_0 := 300$  Волновое сопротивление в Ом

$$x(Z_0) := \cosh \left[ Z_0 \frac{\sqrt{\epsilon}}{60} \right]$$

$$x(Z_0) = 74.21$$

$Z_0 := 50, 60 \dots 300$



Зависимость  $Z_0$  от параметра  $x$

Расчет волнового сопротивления двухпроводной линии, расположенной в свободном пространстве

$d := 0.1$  Диаметр проводников линии

$\epsilon := 1$  Диэлектрическая проницаемость

Док. 6.7. Расчет простых линий передачи (начало)



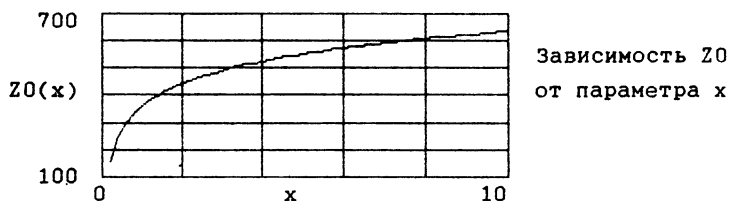
$$Z_0(x) := 120 \operatorname{acosh} \left[ \frac{x}{d} \right] \quad Z_0(2) = 442.59$$

Вычисление расстояния между проводами  $x$   
по заданному волновому сопротивлению  $Z=300$  Ом

$x := 1$  Инициализация  
 $x_0 := \operatorname{root}(Z_0(x) - 300, x)$  Поиск корня  
 $x_0 = 0.613$  Искомое значение  $x$

Построение графика зависимости  $Z_0$  от  $x$  вплоть  
до соприкосновения проводов при  $x=2d$ "

$x := 2d, 2d + 0.2 \dots 10$



Вычисление волнового сопротивления коаксиальной линии  
со смещенным от центра на расстояние  $L$  проводником

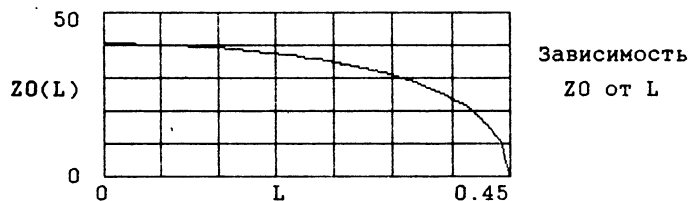
$D := 1$  Внутренний диаметр экрана  
 $d := 0.1$  Диаметр проводника  
 $E := 11.7$  Диэлектрическая проницаемость

$$Z_0(L) := \frac{60}{\sqrt{E}} \operatorname{acosh} \left[ \frac{D^2 + d^2 - 4L^2}{2dD} \right]$$

$$Z_0(0) = 40.39 \quad Z_0(0.2) = 37.291 \quad Z_0(0.4) = 21.5$$

$$L := 0, 0.01 \dots \frac{D}{2} - \frac{d}{2}$$

Построение графика зависимости  $Z_0$  от смещения  $L$   
внутреннего проводника коаксиальной линии



Док. 6.7. Расчет простых линий передачи (окончание)

Расчетная часть

$$E1(P, E) := \frac{E + 1}{2} + \frac{E - 1}{2} \cdot \left[ \left[ 1 + \frac{12}{P} \right]^{\frac{-1}{2}} + 0.041 (1 - P)^2 \right]$$

$$E2(P, E) := \frac{E + 1}{2} + \frac{E - 1}{2} \cdot \left[ 1 + \frac{12}{P} \right]^{-0.5}$$

$$C(P, H, T, E) := (E - 1) \cdot \frac{T}{4.6 \cdot H \cdot P}$$

$$Ee(P, H, T, E) := \text{if}(P < 1, E1(P, E), E2(P, E)) - C(P, H, T, E)$$

$$Z1(P, H, T, E) := \frac{60}{\sqrt{Ee(P, H, T, E)}} \cdot \ln \left[ \frac{8}{P} + \frac{P}{4} \right]$$

$$Z2(P, H, T, E) := \frac{377}{\sqrt{Ee(P, H, T, E)}} \cdot \frac{1}{P + 1.393 + 0.677 \ln(P + 1.444)}$$

$$Z(P, H, T, E) := \text{if}(P < 1, Z1(P, H, T, E), Z2(P, H, T, E))$$

$$P1(W, T, H) := \frac{W}{H} + 1.25 \cdot \frac{T}{3.142 \cdot H} \cdot \left[ 1 + \ln \left[ 2 \cdot \frac{H}{T} \right] \right]$$

$$P2(W, T, H) := \frac{W}{H} + 1.25 \cdot \frac{T}{3.142 \cdot H} \cdot \left[ 1 + \ln \left[ 4 \cdot 3.142 \cdot \frac{W}{T} \right] \right]$$

Пример расчета волнового сопротивления линии

T := 0.01      Толщина полоски в см

E := 11.7      Диэлектрическая проницаемость подложки

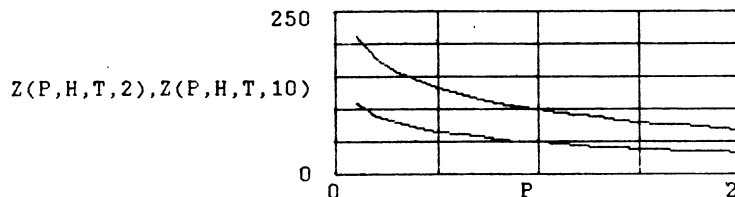
H := 1      Толщина подложки

W := 1      Ширина полоски

$$P := \text{if} \left[ \frac{W}{H} > \frac{1}{6.283}, P1(W, T, H), P2(W, T, H) \right]$$

$$Z0 := Z(P, H, T, E) \quad Z0 = 44.471$$

P := 0.1, 0.2 ... 2      Зависимость Z0 от P=W/H для E=2 и 10



Док. 6.8. Расчет волнового сопротивления несимметричной микрополосковой линии

## 6.4. Расчет линий передачи

Линии передачи — еще один широко распространенный тип компонентов электрических и электронных цепей. Они могут быть проводами, кабелями, полосками в виде тонких пленок (микрополосковые линии). Последние часто применяются в микроэлектронике, тогда как кабельные линии коаксиальной конструкции — в радиотехнических устройствах.

Наиболее важным параметром линий передачи является волновое сопротивление

$$Z_0 = \sqrt{L_n / C_n},$$

где  $L_n$  и  $C_n$  — погонные индуктивность и емкость (на единицу длины).

В электронных и радиотехнических устройствах обычно осуществляется согласование с волновым сопротивлением сопротивлений генератора и нагрузки. В этом случае сложные сигналы, например импульсные, передаются по линии передачи с наименьшими искажениями формы.

Расчеты ряда типов линий передачи даны в док. 6.7. Система MathCAD позволяет проводить расчет зависимости  $Z_0$  от какого-либо конструктивного параметра линии и выводить результаты расчета в графической форме.

Однако для некоторых линий расчет оказывается не столь простым и требует вполне обоснованного применения компьютеров. Примером может служить расчет довольно распространенных в электронике микрополосковых линий по формулам, приведенным в [11] и дающим погрешность не более 1—2 %. Такой расчет иллюстрирует док. 6.8. Предоставим читателю судить, хватит ли у него смелости взяться за такой расчет, не располагая компьютером.

## 6.5. Диаграмма Бодэ

В электротехнике и электронике есть немало методов расчета, дающих наглядное (разумеется, для специалиста) представление о работе цепей. Например, весьма плодотворным является операторный метод расчета. Он получается из другого символического метода заметной величины  $i\omega$ , где  $i$  — мнимая единица и  $\omega$  — круговая частота, на оператор Лапласа  $s$  (или  $p$ ). Это сильно упрощает аналитические преобразования цепей, сводя их интегродифференциальные уравнения к алгебраическим уравнениям. После таких преобразований цепь описывается некоторой операторной функцией  $F(s)$ . Затем, используя хорошо развитый аппарат преобразования изображения  $F(s)$  в оригинал, можно вернуться к обычному представлению цепи в частотной или временной области.

Здесь мы рассмотрим цепи, которые описываются дифференциальными уравнениями второго порядка. Роль таких цепей поистине фундаментальна — к ним сводятся избирательные усилители, генераторы синусоидальных и релаксационных колебаний, системы автоматического управления с перерегулированием и т. д. В операторном виде цепи второго порядка описываются выражением

$$F(s) = \omega^2 / (s^2 + 2 \cdot p \cdot \omega \cdot s + \omega^2),$$

где  $p = \sqrt{2}/2$ ;  $\omega$  — круговая частота;  $s$  — оператор Лапласа.

Важными характеристиками таких систем являются логарифмическая амплитудно-частотная характеристика (ЛАЧХ) и ФЧХ. Кроме того, эти зависимости, называемые диаграммой Бодэ, нередко представляются на комплексной плоскости, что удобно, например, для анализа устойчивости систем второго порядка.

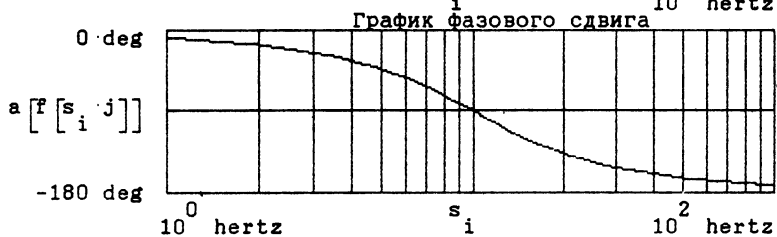
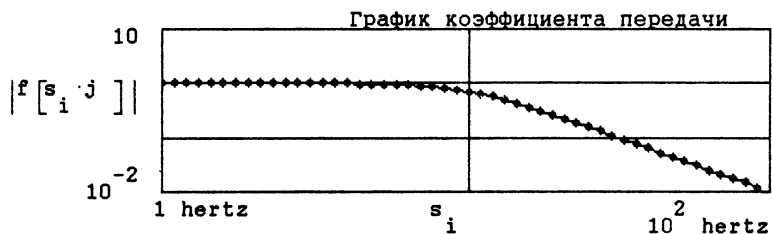
Документ 6.9 иллюстрирует построение ЛАЧХ и ФЧХ для системы или цепи второго порядка, представленной операторной функцией  $F(s)$ . Строится и диаграмма на комплексной плоскости.

```

p :=  $\frac{\sqrt{2}}{2}$        $\omega := 10 \text{ hertz}$ 

f(s) :=  $\frac{\omega^2}{s^2 + 2 \cdot p \cdot \omega \cdot s + \omega^2}$       Передаточная функция
                                         для системы 2-го порядка
n := 50      i := 0 .. n      min := 1      max := 100
r :=  $\ln \left[ \frac{\text{max}}{\text{min}} \right]$ 
                                          $s_i := \min \cdot e^{i \cdot \frac{r}{n}}$  hertz
Глобальные определения
sec  $\approx$  1t      hertz  $\approx \frac{6.283}{\text{sec}}$       deg  $\approx \frac{6.283}{360}$       j :=  $\sqrt{-1}$ 
a(g)  $\approx$  angle(Re(g),Im(g)) - 360 deg

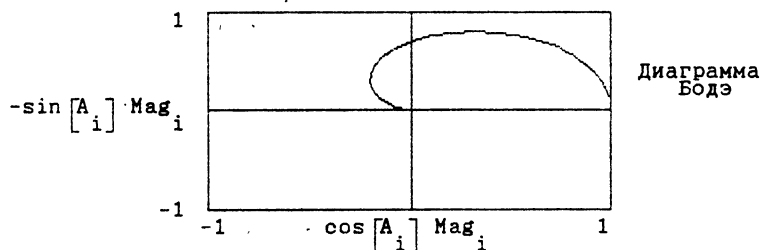
```



```

A_i := a[f[s_i * j]]      Mag_i := |f[s_i * j]|

```



Док. 6.9. Построение диаграммы Бодэ для системы второго порядка

## 6.6. Преобразование Лапласа

К числу фундаментальных преобразований в теоретических основах электротехники и радиотехники относится также преобразование Лапласа [12]. С помощью этого преобразования можно перейти от описания линейной цепи во временной

```
inf := 10      Верхний предел интегрирования
TOL := .01     Погрешность
Lap(F,P) := ∫0inf [e-(P t) · F(t)] dt    Интеграл Лапласа
                                           приближенный
```

### Пример 1

$F(t) := \sin(t)$       Заданная функция  $F(t)$

$$\text{Lsin}(p) := \frac{1}{1 + p^2}$$

Преобразование Лапласа  
аналитическое

Lap(F,1) = 0.4999	Сравнение численного и
Lsin(1) = 0.5	аналитического преобразований

$$\text{Lap}(F, 2) = 0.1994$$
$$L_{\sin}(2) = 0.2$$

### Пример 2

$F(t) := \Phi(t)$       Функция Хевисайда

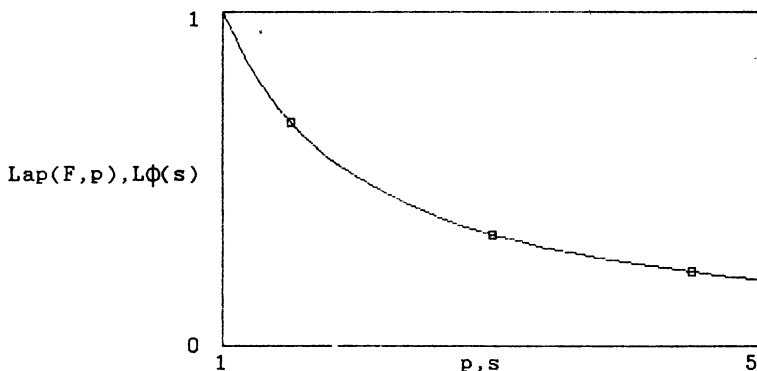
$$L\phi(s) := \frac{1}{s}$$

Преобразование Лапласа аналитическое

Построение графика преобразованной функции Хевисайда по аналитическому выражению, а точек – для численного расчета

$p := 1.5, 3 \dots 5$       Точки для аргумента  $p=1.5, 3$  и  $4.5$

<b>s := 1,1.2 ..5</b>	<b>Задание аргумента для аналитической зависимости</b>
-----------------------	--



### Док. 6.10. Преобразование Лапласа

области к описанию в операторной форме. Такое преобразование называется прямым преобразованием Лапласа.

Прямое преобразование Лапласа осуществляется с помощью следующего интегрального преобразования:

$$L(p) = \int_0^{\infty} F(t) \cdot e^{-p \cdot t} dt,$$

где  $p$  — оператор Лапласа (выше отмечалось, что он может обозначаться и буквой  $s$ ).

Если функция  $F(t)$  полностью определена на конечном интервале времени, то бесконечный верхний предел можно заменить на конечный. Тогда преобразование Лапласа легко реализовать, используя оператор вычисления определенного интеграла, имеющийся в системе MathCAD.

Документ 6.10 демонстрирует проведение преобразования Лапласа для любой функции  $F(t)$ , заданной как функция пользователя. Преобразование Лапласа также задается функцией пользователя  $Lap(F, p)$ , имеющей два аргумента. Первый аргумент (на это стоит обратить внимание) является указанием не на переменную, а на функцию  $F(t)$ , тогда как второй  $p$  — обычная переменная.

Для проверки правильности преобразования можно использовать примеры известных преобразований Лапласа, представленные в аналитическом виде. Так, первый пример в док. 6.10 иллюстрирует преобразование Лапласа для синусоидальной функции:

$$F(t) = \sin(t) \rightarrow L(p) = 1/(1 + p^2).$$

Нетрудно заметить, что значения  $L(p)$ , вычисленные по аналитическому выражению, практически совпадают со значениями  $Lap(F, p)$ .

Во втором примере док. 6.10 в качестве  $F(t)$  задана функция Хевисайда — единичный скачок. Для нее преобразование Лапласа дает  $L(p) = 1/p$ . Построен график этой функции, на котором квадратиками отмечены результаты численного преобразования Лапласа для трех точек. Эти точки точно укладываются на график функции  $L(p)$ .

Аналогично можно определить и обратное преобразование Лапласа, дающее переход от операторной функции к ее временному представлению.

## 6.7. Гармонический синтез сигналов

Еще одна фундаментальная закономерность математики — это возможность представления любой периодической функции  $y(t)$  в виде суммы бесконечного числа синусоидальных или гармонических составляющих. Другими словами, функцию  $y(t)$  можно представить в виде ряда Фурье:

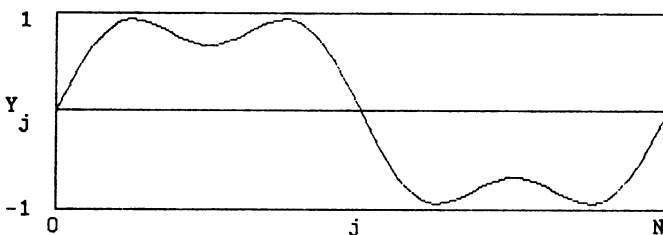
$$y(t) = y_0 + Y_{1m} \cdot \sin\left(2\pi \frac{t}{T} + \psi_1\right) + Y_{2m} \cdot \sin\left(2\pi \frac{2 \cdot t}{T} + \psi_2\right) + \dots,$$

где  $Y_{im}$  — амплитуда  $i$ -й гармоники;  $\psi_i$  — фаза этой гармоники;  $T = 1/f$  — период повторения функции;  $f$  — частота первой гармоники.

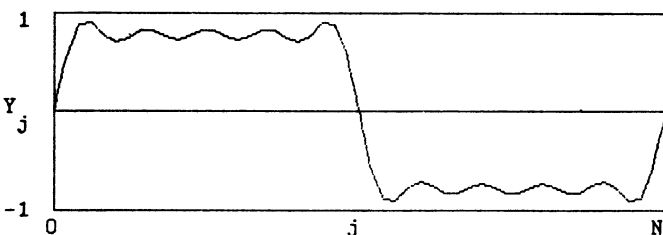
Разложение  $y(t)$  в ряд Фурье называется спектральным анализом. Он обсуждался в § 4.8, где приводился и пример спектрального анализа прямоугольного импульса. Обратная задача — получение функции  $y(t)$  путем суммирования гармоник — называется гармоническим синтезом. Разумеется, на практике синтез ведется не по бесконечному, а по конечному числу гармоник. Это означает, что реально полученная после синтеза функция несколько отличается от ожидаемой. Большим подспорьем в проведении синтеза является возможность получения графика синтезируемой функции при различном числе гармоник.

Гармонический синтез имеет вполне самостоятельное значение. Известны генераторы сигналов специальной формы, получаемой суммированием ряда синусоидальных сигналов с кратными частотами и заданными амплитудами и фазами.

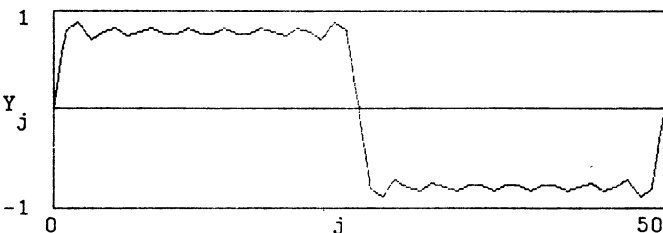
$N := 50$       Число точек графика  
 $j := 0 \dots N$       Задание цикла  
 $i := 1, 3 \dots 3$       Синтез по 1 и 3 гармоникам  
 $Y_j := \sum_i \frac{1}{i} \sin \left[ 2 i y \frac{j}{N} \right]$       Формула синтеза



$i := 1, 3 \dots 9$       Синтез по 1, 3, 5, 7 и 9 гармоникам  
 $Y_j := \sum_i \frac{1}{i} \sin \left[ 2 i y \frac{j}{N} \right]$



$i := 1, 3 \dots 15$       Синтез по 1, 3, ..., 13 и 15 гармоникам  
 $Y_j := \sum_i \frac{1}{i} \sin \left[ 2 i y \frac{j}{N} \right]$



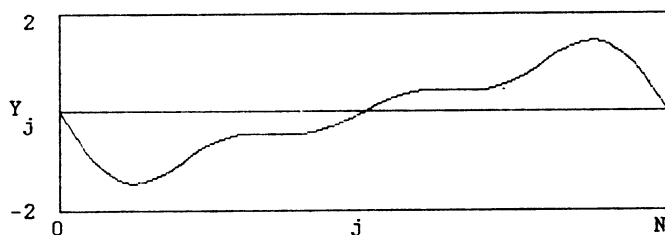
Док. 6.11. Гармонический синтез сигнала-меандра

К тому же результату можно прийти, используя гармонический синтез с помощью системы MathCAD.

Документ 6.11 иллюстрирует синтез прямоугольных колебаний — меандра. Теоретически такие колебания содержат только нечетные гармоники с нулевыми

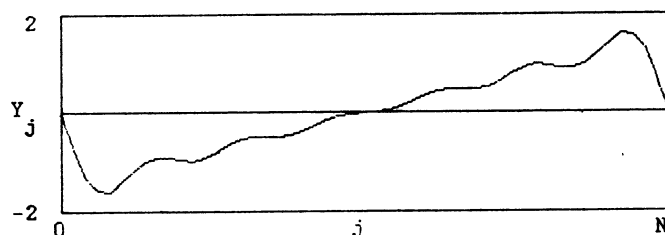
$N := 50$             Число точек графика  
 $j := 0 \dots N$         Задание цикла  
 $i := 1 \dots 3$         Синтез по 3 первым гармоникам

$$Y_j := \sum_i \frac{-1^{i+1}}{i} \sin \left[ 2 i y \frac{j}{N} \right]$$



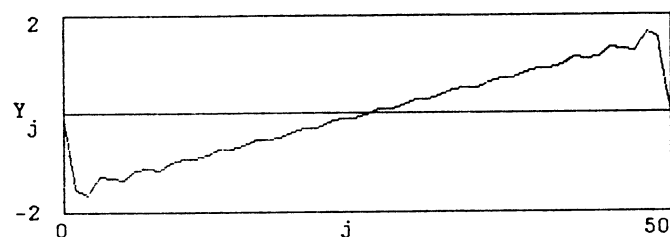
$i := 1 \dots 6$             Синтез по первым 6 гармоникам

$$Y_j := \sum_i \frac{-1^{i+1}}{i} \sin \left[ 2 i y \frac{j}{N} \right]$$



$i := 1 \dots 15$         Синтез по первым 15 гармоникам

$$Y_j := \sum_i \frac{-1^{i+1}}{i} \sin \left[ 2 i y \frac{j}{N} \right]$$



Док. 6.12. Гармонический синтез сигнала-пилы

фазами и амплитудами, обратно пропорциональными их номеру  $i$ . Из док. 6.11 видно, как по мере увеличения числа гармоник функция  $y(t)$  постепенно приближается к меандру.

Документ 6.12 иллюстрирует синтез пилообразных колебаний также по ряду



гармоник. Амплитуда гармоник (как четных, так и нечетных) обратно пропорциональна  $i$ , а знак ее определяется как знак  $-i^{i+1}$ . Смена знака означает возникновение фазового сдвига в  $\pi$  радиан.

Можно заметить, особенно при синтезе меандра, что после скачков синтезируемой функции возникают характерные постепенно затухающие выбросы. Это явление получило известность как эффект Гиббса. К сожалению, увеличение числа гармоник при синтезе не устраняет этот эффект. Выбросы становятся короче, но их относительная амплитуда почти не меняется и достигает 18 %.

Синтеза — операция довольно медленная, так как приходится вычислять большое число тригонометрических функций. Однако формирование с помощью плавных синусоид довольно резко изменяющихся и почти импульсных сигналов вполне оправдывает те несколько минут, которые нужны для синтеза в приведенных выше примерах. Синтез легко совместить с оценкой линейных искажений сигналов, вызванных ограниченной полосой пропускания электронных усилительных устройств и возникновением фазовых сдвигов. Для этого амплитуду каждой гармоники достаточно умножить на модуль коэффициента передачи на частоте этой гармоники, а к фазовому сдвигу гармоники добавить фазовый сдвиг усилителя. В результате получим мощный аппарат анализа линейных цепей как в частотной, так и во временной области. К примеру, резко ослабить эффект Гиббса можно, умножив амплитуду каждой  $i$ -й гармоники на корректирующий множитель  $\sin(\pi i t) / (\pi i t)$ .

И последнее. Достоинство описанной методики синтеза в том, что число гармоник можно выбирать любым. Напоминаем, что при прямом и обратном БПФ число гармоник может быть равно только  $2^n$ , где  $n$  — целое число. В то же время БПФ реализуется быстро. В § 6.8 приводится интересный пример синтеза с применением обратного БПФ.

## 6.8. Фильтрация сигналов на фоне шумов

В электронике и радиотехнике одной из важнейших задач является фильтрация сигналов на фоне шумов. Для этого применяются различные методы. Здесь рассматривается спектральный метод. Его суть заключается в том, что из спектра суммы сигнала с помехами выделяется та область частот, в которой находятся гармонические составляющие сигнала с достаточной амплитудой.

Документ 6.13 иллюстрирует фильтрацию. В начале документа с помощью ряда Фурье синтезируется сам сигнал достаточно сложной формы (см. первый график). Затем с помощью генератора случайных чисел имитируется шумовая компонента сигнала, которая добавляется к нему. В результате получается зашумленный сигнал (второй график).

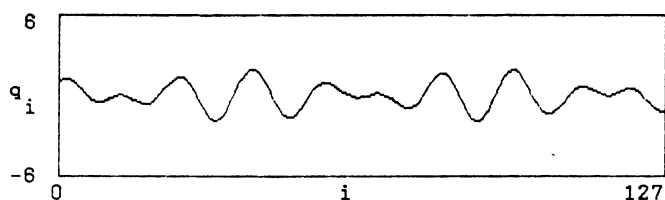
Далее идет реализация фильтрации. Вначале сигнал подвергается прямому БПФ, что означает переход от временного представления сигнала к частотному и позволяет применить частотную фильтрацию. Отфильтрованный сигнал подвергается затем обратному преобразованию Фурье, что обеспечивает возврат к временному представлению сигнала. Последний график показывает отфильтрованный сигнал. Сравнение его формы с формой исходного сигнала (первый график) позволяет судить об эффективности фильтрации.

В данном случае применение БПФ способствует уменьшению времени математической обработки сигнала, включая его гармонический синтез. Тем не менее это время значительно и не позволяет применять данный вид реализации фильтрации в реальном масштабе времени. Он, однако, дает прекрасные результаты при обработке измерительной или предварительно накопленной информации, когда время анализа не имеет особого значения. Оно резко уменьшается при реализации фильтрации на персональных компьютерах класса IBM PC AT и PS/2 с повышенной тактовой частотой работы процессора и наличием математического сопроцессора, ускоряющего математические вычисления.

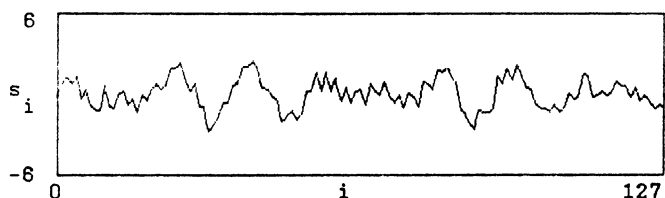
Приведенный пример дает хорошее представление о математических основах спектрального метода фильтрации. В настоящее время имеются микросхемы, реализующие БПФ с элементами фильтрации и синтеза сигналов (фурье-процессоры). Их применение делает возможным проведение фильтрации и в реальном масштабе времени.

$i := 0 \dots 127$       Задание сигнала

$$q_i := \sin\left[\frac{i}{128} \cdot 14 \cdot \pi\right] + \cos\left[\frac{i}{128} \cdot 19 \cdot \pi\right]$$



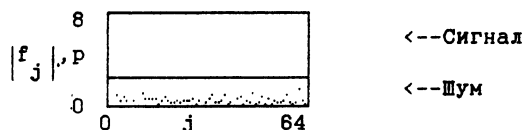
$s_i := q_i + \text{rnd}(2) - 1$       Добавка к сигналу шума



Дискретное преобразование Фурье

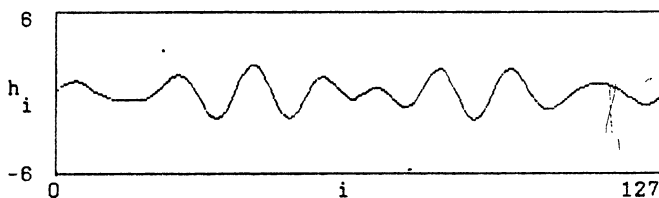
$f := \text{fft}(s)$        $j := 0 \dots 64$

$p := 2.5$       Задание показателя фильтрации



Фильтрация и обратное преобразование Фурье

$g_j := f_j \cdot \Phi[|f_j| - p]$        $h := \text{ifft}(g)$



Док. 6.13. Фильтрация зашумленного сигнала с помощью БПФ

## 6.9. Расчет нелинейных цепей в статическом режиме

Под статическим режимом работы цепей подразумевается работа цепей на постоянном токе. Расчет статических режимов нелинейных цепей в большинстве случаев, даже весьма простых, невозможен аналитическими методами, поскольку уравнения, описывающие работу нелинейных цепей, нелинейны и чаще всего transcendентны. Поэтому целесообразен расчет численными методами. Применение системы MathCAD, имеющей встроенные средства для реализации таких методов, обеспечивает простоту и наглядность расчета статических режимов нелинейных цепей.

Рассмотрим простейшую цепь, содержащую диод, который через резистор  $R$  подключен к источнику напряжения  $E$ . Падение напряжения на диоде можно найти, решив систему уравнений

$$\begin{aligned} I &= f(U), \\ I &= (E - U)/R, \end{aligned}$$

где  $f(U)$  — нелинейная зависимость напряжения на диоде от протекающего через него тока, т. е. ВАХ диода. Первое уравнение геометрически является графиком ВАХ, а второе — графиком нагрузочной прямой. Их точка пересечения дает значения  $I$  и  $U$  в статическом режиме.

Обычный диод имеет ВАХ, описываемую выражением

$$I(U) = I_0 \cdot (e^{k \cdot U} - 1),$$

где  $I_0$  — обратный ток диода;  $k = 1/m\psi_T$ ;  $m$  — коэффициент, учитывающий рекомбинацию носителей в p-n переходе и  $\psi_T$  — тепловой потенциал.

Для столь простой схемы систему из двух уравнений целесообразно свести к одному нелинейному уравнению

$$F(U) = I(U) - (E - U)/R = 0.$$

В этом случае можно использовать функции системы, обеспечивающие поиск действительных корней нелинейного уравнения.

Документ 6 14 содержит два примера расчета диодных цепей. Первый соответствует только что описанному случаю — расчету падения напряжения на обычном диоде. На графике представлено построение ВАХ диода и линии нагрузочки резистора  $R$ . Такое построение всегда целесообразно, например с целью грубого определения решения, которое затем уточняется.

Второй пример не столь тривиален. Он поясняет вычисление падения напряжения на туннельном диоде, имеющем N-образную ВАХ. Как видно из геометрического представления решения, в данном случае ВАХ пересекается нагрузочной прямой уже в трех точках. Не вдаваясь в подробности, отметим, что средняя точка, лежащая на падающем участке ВАХ, неустойчива на постоянном токе. Так что решение соответствует двум стабильным состояниям цепи, при которых рабочая точка лежит на восходящих ветвях ВАХ с положительной дифференциальной проводимостью.

Во втором примере ВАХ диода

$$I(U) = A1 \cdot U \cdot e^{-a1 \cdot U} + A2 \cdot (e^{a2 \cdot U} - 1),$$

где первый член описывает туннельную составляющую тока, текущего через p-n переход диода, а второй — диффузионную составляющую. Методика подбора параметров аппроксимации  $A1$ ,  $a1$ ,  $A2$  и  $a2$  описана в [1].

В данном случае решение уравнения  $F(U) = 0$  с помощью функции `root` требует задания приближенных и разных начальных значений  $U$ . Их можно определить из графического решения. Чтобы переменная  $U$  не ассоциировалась с ранее введенной индексной переменной  $U_i$ , напряжение на диоде в данной части документа обозначено как  $V$ .

1. Расчет падения напряжения  $U$  на обычном диоде  
 $E := 0.5$      $R := 25$      $I_0 := 0.000001$      $K := 20$   
 $U := 0.2$  Начальное приближение для  $U$   
 Given    Блок решения

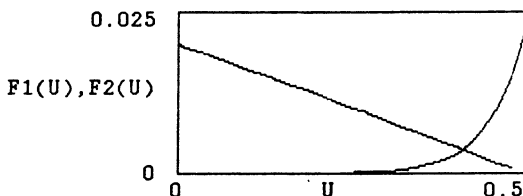
$$\frac{E - U}{R} \leftarrow I_0 \cdot [e^{K \cdot U} - 1]$$

$U0 := \text{Find}(U)$      $U0 = 0.41$     Найденное решение

Графическая иллюстрация

$U := 0, 0.02 \dots E$

$$F1(U) := \frac{E - U}{R} \quad F2(U) := I_0 [e^{K \cdot U} - 1]$$

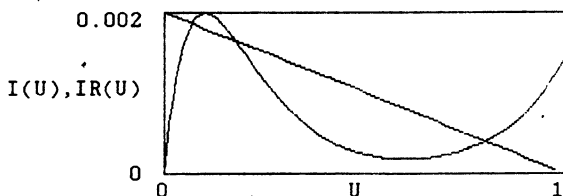


2. Расчет падения напряжения на туннельном диоде

$A1 := 0.0543$      $a1 := 10$      $A2 := 2.187 \cdot 10^{-6}$      $a2 := 6.493$   
 $I(U) := A1 \cdot U \cdot e^{-(a1 \cdot U)} + A2 \cdot [e^{a2 \cdot U} - 1]$     N-образная ВАХ ТД  
 $E := 1$      $R := 500$

$$IR(U) := \frac{E - U}{R} \quad \text{Уравнение нагрузочной прямой}$$

$U := 0, 0.025 \dots 1$



$$F(U) := I(U) - IR(U)$$

$V := 0$      $V1 := \text{root}(F(V), V)$      $V1 = 0.036$

$V := 0.2$      $V2 := \text{root}(F(V), V)$      $V2 = 0.177$

$V := 0.8$      $V3 := \text{root}(F(V), V)$      $V3 = 0.799$

Три  
решения  
для функции  
 $F(U)=0$

Док. 6.14. Примеры расчета падения напряжения на диоде

Даже из этих простых примеров хорошо видно, что центральным моментом в расчете статического режима нелинейных цепей является правильное описание нелинейных ВАХ. Обычно при этом используются подходящие аппроксимации,

по возможности полученные из рассмотрения физических основ работы нелинейных приборов, но нередко формальные.

Документ 6.15 иллюстрирует эффективные возможности системы MathCAD в оценке более сложных аппроксимаций, описывающих семейства выходных ВАХ

Расчет и построение семейства ВАХ

$$U_0 := 1 \quad S := 0.15 \quad B := 0.01 \quad P := 1$$

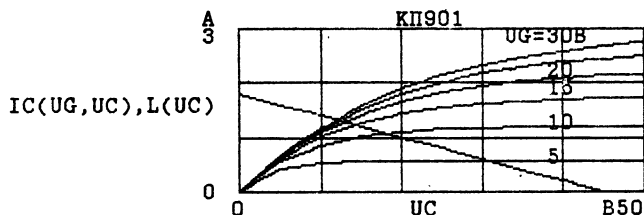
$$EC := 45 \quad RC := 25$$

$$UG := 5, 10 \dots 30 \quad UC := 0, 5 \dots 55$$

$$IC(UG, UC) := S \cdot [UG - U_0 - B \cdot UC^2] \cdot \left[ 1 - e^{-P \cdot \frac{UC}{UG - U_0 - B \cdot UC^2}} \right]$$

$$L(UC) := \frac{EC - UC}{RC}$$

График семейства выходных ВАХ



Расчет и построение сквозной динамической характеристики

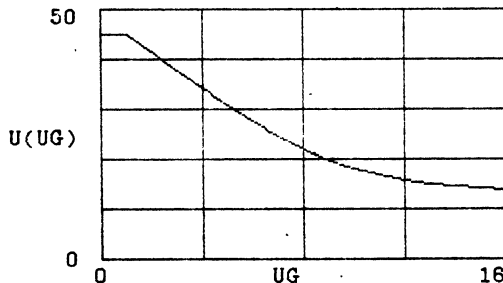
$$Im(UG) := S \cdot [UG - U_0 - B \cdot UC^2] \quad UC := 25$$

$$IC(UG) := \text{if} \left[ UG > U_0, Im(UG) \cdot \left[ 1 - e^{-P \cdot S \cdot \frac{UC}{Im(UG)}} \right], 0 \right]$$

$$UG := 0, 1 \dots 16$$

$$U(UG) := \text{root}(EC - UC - RC \cdot IC(UG), UC)$$

График передаточной характеристики



Док. 6.15. Расчет и построение сквозной передаточной характеристики каскада на мощном МДП-транзисторе

мощных полевых транзисторов. Эти ВАХ характеризуют зависимость тока стока транзистора от напряжения на стоке для нескольких фиксированных значений напряжения на затворе. Построение семейств ВАХ в графическом общепринятом виде позволяет наглядно сравнить эти семейства с приводимыми в технических документах и скорректировать параметры аппроксимаций. Их можно использовать в специальных программах.

Наряду с ВАХ можно построить и статические передаточные характеристики, отражающие зависимость тока стока от напряжения на затворе при фиксированном напряжении на стоке. Такое построение также дано.

Если напряжение питания на сток подается через резистор, то по мере увеличения напряжения на затворе и роста тока стока напряжение на затворе будет уменьшаться. Тогда зависимость тока стока или напряжения на стоке от напряжения на затворе будет описываться динамическими сквозными передаточными характеристиками.

Документ 6.15 иллюстрирует построение сквозной динамической передаточной характеристики каскада с общим истоком на мощном полевом транзисторе. Трудоемкость расчетов в этом случае возрастает, так как нелинейную систему уравнений нужно решать несколько раз — для каждой точки передаточной характеристики.

Динамическая передаточная характеристика несет много информации о работе схем. Так, можно оценить линейность каскада, выбрать положение рабочей точки, провести спектральный анализ выходного сигнала цепи при гармоническом сигнале на входе, оценить дифференциальный коэффициент усиления.

### 6.19. Спектральный анализ со сплайн-интерполяцией передаточной характеристики

Только что мы показали, как строится передаточная характеристика численным методом. Но нередко она бывает задана рядом экспериментальных точек, получена с помощью характеристикиографа или просто взята из технической документации или литературы. Словом, такая характеристика в общем виде задается рядом известных точек  $X_i$  и  $Y_i$ , где  $X_i$  — значения входного напряжения или тока, а  $Y_i$  — соответствующие им значения выходного напряжения или тока.

С помощью известных методов спектрального анализа можно вычислить параметры ряда гармоник при воздействии на вход схемы гармонического сигнала. Наиболее распространенным инженерным методом спектрального анализа такого рода является метод пяти ординат. Сначала определяются пять ординат передаточной характеристики при диапазоне изменений входного сигнала, соответствующем двойной амплитуде его синусоидальной составляющей. Затем по этим ординатам рассчитываются постоянная составляющая  $Y_0$ , амплитуды  $Y_{M1}$  четырех гармоник и коэффициент нелинейных искажений (коэффициент гармоник):

$$Kg = \sqrt{Y_{M2}^2 + Y_{M2}^2 + Y_{M4}^2} / (Y_{M1}).$$

Данные об ординатах передаточной характеристики могут быть получены как из нее самой, так и из точек пересечения кривых семейства ВАХ линиями нагрузки. В любом случае расчет для ряда амплитуд входных воздействий оказывается весьма трудоемким и неточным. Нередки, скажем, случаи, когда студенты в результате таких расчетов приносят абсурдные данные, например  $Kg > 1$  при довольно линейной передаточной характеристике или  $Kg$  порядка 0.01 % при явно нелинейной передаточной характеристике.

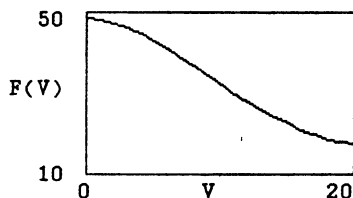
Документ 6.16 описывает нетривиальную реализацию спектрального анализа методом пяти ординат по заданной передаточной характеристике. Характеристика может задаваться любым числом отсчетов, а не обязательно пятью. Напротив, для повышения точности анализа лучше задавать 7—8 точек. Необходимая для анализа непрерывная передаточная характеристика получается путем сплайн-

```

X := [ 1
      3
      7
      10
      13
      16
      19 ]
Y := [ 48
      46.1
      38.9
      32
      26
      21
      18 ]
X-вектор входного сигнала
Y-вектор выходного сигнала
K := 0 .. 6
Сплайн-аппроксимация F(V)
YS := cspline(X,Y)
F(V) := interp(YS,X,Y,V)
V := 0,1 .. 20

```

Построение графика  
передаточной функции  
 $F(V)$



Спектральный анализ

```

X0 := 10   Xmax := 8   i := 1 .. 10   DX_i := 0.1 * Xmax * i

```

```

Y1_i := F[X0 - DX_i]   Y2_i := F[X0 - 0.5 * DX_i]

```

```

Y3_i := F(X0)           Y4_i := F[X0 + 0.5 * DX_i]

```

```

Y5_i := F[X0 + DX_i]

```

```

Y0_i := 0.1667 * [Y1_i + Y5_i + 2 * Y2_i + 2 * Y4_i]

```

```

YM1_i := 0.3333 * [Y1_i - Y5_i + Y2_i - Y4_i]

```

```

YM2_i := 0.25 * [Y1_i + Y5_i - 2 * Y3_i]

```

```

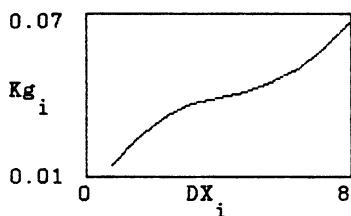
YM3_i := 0.5 * [YM1_i - Y2_i + Y4_i]

```

```

YM4_i := YM2_i - Y0_i + Y3_i

```

$$Kg_i := \frac{\sqrt{YM2_i^2 + YM3_i^2 + YM4_i^2}}{YM1_i}$$


DX <sub>i</sub>
0.8
1.6
2.4
3.2
4
4.8
5.6
6.4
7.2
8

Kg <sub>i</sub>
0.015
0.025
0.032
0.037
0.039
0.041
0.045
0.05
0.057
0.067

Док. 6.16. Спектральный анализ методом пяти ординат со сплайн-интерполяцией передаточной характеристики

интерполяции (и экстраполяции) исходных данных. Поэтому их не нужно вводить снова для каждого значения амплитуды входного сигнала.

В док. 6.16 выполняется спектральный анализ методом пяти ординат десятикратно для различных значений амплитуд входного гармонического сигнала. В итоге строится зависимость коэффициента гармоник  $K_g$  от амплитуды входного сигнала. Эта зависимость дает гораздо больше информации о поведении схемы, чем одиночный расчет. В частности, по ней можно найти компромисс между допустимым значением  $K_g$  и максимальной амплитудой входного и выходного сигналов, определяющей максимальную выходную мощность.

Здесь не описан сам метод пяти ординат как ввиду его общеизвестности [1—3], так и в силу того, что все относящиеся к нему расчетные выражения приведены в док. 6.16. Их громоздкость явно свидетельствует в пользу проведения таких расчетов на компьютере.

## 6.11. Расчет выпрямителя в стационарном режиме

Диодные выпрямители переменного напряжения — одни из наиболее часто применяемых нелинейных устройств. Они встречаются едва ли не в каждом электронном или радиотехническом устройстве. Специфика расчета выпрямителя связана со стационарным режимом его работы, зависящим от характера нагрузки. Мы будем рассматривать наиболее часто встречаемый случай, когда выпрямитель нагружен на сглаживающий конденсатор  $C$  большой емкости и включенный параллельно ему резистор  $R$ .

Диод (или диоды при многофазном выпрямлении) открывается только на небольшую часть периода синусоидального входного напряжения, которая определяется углом отсечки  $Q$ . За это время конденсатор  $C$  заряжается и затем разряжается на нагрузку  $R$ . Угол отсечки в стационарном режиме работы выпрямителя (его не следует путать со статическим режимом, так как на вход подано переменное, а не постоянное напряжение) определяется из решения трансцендентного уравнения

$$\tan(Q) - Q = \pi \cdot r / (m \cdot R) = A,$$

где  $r$  — общее сопротивление открытого диода и источника переменного напряжения;  $m$  — число фаз выпрямителя. Например, для наиболее распространенного двухфазного мостового выпрямителя  $m=2$ . Решение данного уравнения аналитическим методом невозможно.

Документ 6.17 иллюстрирует хорошо известную методику расчета выпрямителя. Вначале из приведенного уравнения, решаемого численным методом, находится угол отсечки  $Q$ . Затем определяются функции  $K(Q)$ ,  $F(Q)$ ,  $B(Q)$  и  $D(Q)$ . Через них находят основные параметры выпрямителя (см. формулы в док. 6.15).

## 6.12. Расчет переходных процессов для RC-цепи

Простейшая RC-цепь имеет важное самостоятельное значение и может применяться для отладки численных методов расчета переходных процессов. Последнее связано с тем, что реакцию RC-цепи на простое входное воздействие легко рассчитать аналитически и сравнить результаты такого расчета с результатами численного расчета.

Вначале рассмотрим интегрирующую цепь, в которой конденсатор  $C$  подключен к источнику напряжения  $e(t)$  через резистор  $R$ , причем выходным является напряжение конденсатора  $u(t)$ . Такая цепь описывается линейным дифференциальным уравнением первого порядка

$$du/dt = [e(t) - u(t)] / (R \cdot C).$$

Как отмечалось, решение его при простых  $e(t)$  не представляет трудностей. Однако при достаточно сложных  $e(t)$  решение становится громоздким, что оправ-



$m := 2$  Число фаз выпрямителя  
 $r := 100$  Внутреннее сопротивление выпрямителя  
 $R := 1000$  Сопротивление нагрузки  
 Вычисление угла отсечки

$$A := 3.141 \cdot \frac{r}{m R} \quad A = 0.157$$

$$x := 0 \quad F(x) \approx \tan(x) - x$$

$$Q := \text{root}(F(x) - A, x) \quad Q = 0.72$$

Вычисление вспомогательных коэффициентов

$$K(Q) := \sin(Q) - Q \cos(Q) \quad K(Q) = 0.118$$

$$F(Q) := 3.141 \cdot \frac{1 - \cos(Q)}{K(Q)} \quad F(Q) = 6.601$$

$$E(Q) := \frac{1}{\sqrt{2} \cos(Q)} \quad B(Q) = 0.941$$

$$\Gamma(Q) := \frac{\sqrt{3.141 \cdot \left[ Q \cdot \left[ 1 + \frac{\cos(2 \cdot Q)}{2} \right] - 3 \cdot \frac{\sin(2 \cdot Q)}{4} \right]}}{K(Q)}$$

$$D(Q) = 2.294$$

Расчет напряжений и токов

$$U_0 := 150$$

$$I_0 := .15$$

$$f := 50$$

$$K_p := .1$$

$$U_2 := U_0 \cdot B(Q) \quad U_2 = 141.087 \quad A$$

$$I_{2m} := I_0 \cdot \frac{F(Q)}{n} \quad I_{2m} = 0.495 \quad A$$

$$I_e := D(Q) \cdot \frac{I_0}{n} \quad I_e = 0.172 \quad A$$

$$I_2 := \sqrt{2} \cdot I_e \quad I_2 = 0.243 \quad A$$

$$C := \frac{\frac{1}{n} + \frac{Q}{3.141}}{2 \cdot K_p \cdot f \cdot R} \quad C = 7.293 \cdot 10^{-5} \quad \Phi$$

Док. 6.17. Расчет сетевого выпрямителя

Моделирование интегрирующей RC-цепи простым методом Эйлера

$T1 := 1$      $T2 := 0.5$      $T := 1$      $dt := 0.1$

$UM := 10$      $n := 50$

$i := 0 \dots n$      $t_0 := 0$      $u_0 := 0$     Начальные условия

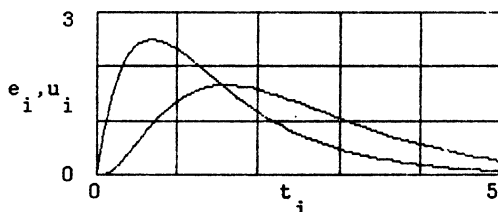
$t_i := i \cdot dt$     Текущее время

$$e_i := UM \left[ e^{\frac{-t_i}{T1}} - e^{\frac{-t_i}{T2}} \right]$$

Входное воздействие  $e(t)$

$$u_{i+1} := u_i + \frac{e_i - u_i}{T} \cdot dt$$

Формула метода Эйлера



Графики  $e(t)$  и  $u(t)$

Моделирование интегрирующей RC-цепи комбинированным (явным и неявным) методом Эйлера

$T1 := 1$      $T2 := 0.5$      $T := 1$      $dt := 0.2$

$UM := 10$      $n := 25$      $p := 2 \cdot \frac{T}{dt}$

$$e(t) := UM \left[ \exp\left[\frac{-t}{T1}\right] - \exp\left[\frac{-t}{T2}\right] \right]$$

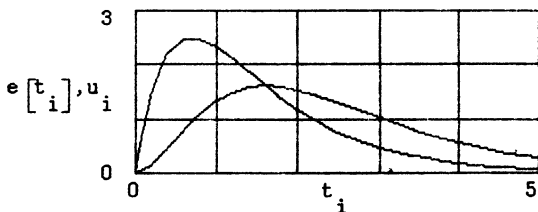
Задание  $e(t)$

$i := 0 \dots n$      $t_0 := 0$      $u_0 := 0$     Начальные условия

$t_i := i \cdot dt$     Текущее время

$$u_{i+1} := \frac{p \cdot u_i - u_i + e[t_i] + e[t_i + dt]}{1 + p}$$

Формула комбинированного метода Эйлера



Графики  $e(t)$  и  $u(t)$

Док. 6.18. Примеры моделирования интегрирующей RC-цепи

дывает применение численных методов. Мы рассмотрим вначале реакцию RC-цепи на двухэкспоненциальный импульс:

$$e(t) = UM \cdot (e^{-t/T_1} - e^{-t/T_2}),$$

где  $UM$  — амплитуда импульса;  $T_1$  и  $T_2$  — временные параметры, характеризующие постоянные времени нарастания и спада импульса.

Документ 6.18 демонстрирует моделирование переходных процессов в RC-цепи двумя математическими методами. Первый — простой метод Эйлера. Он является явным, поскольку новое значение  $u(t)$  находится по известному предыдущему значению. При больших шагах  $h = dt$  такой метод приводит к числовой неустойчивости решения. Впрочем, шаг приходится уменьшать по другой причине — погрешность метода пропорциональна  $h^2$  и при больших  $h$  оказывается неприемлемой даже для прикидочных расчетов.

Второй метод — комбинированный ( неявный и явный ). При нем

$$u_{i+1} = u_i + h \cdot [A \cdot f(t_i, u_i) + (1 - A) \cdot f(t_i, u_{i+1})].$$

Если  $A=0$ , то реализуется простой (явный) метод Эйлера, если  $A=1$  — неявный метод. При последнем  $u_{i+1}$  входит как в левую, так и в правую часть итерационной формулы. Наконец, при  $A=0,5$  реализуется комбинированный метод. В этом случае еще нет неустойчивости решения при больших  $h$  и в то же время существенно уменьшается погрешность метода. Связано это с тем, что погрешности явного и неявного методов по модулю близки, но по знаку противоположны. Поэтому они частично взаимно компенсируются.

Определенные трудности связаны с тем, что сохраняется неявный вид формулы. Разрешить ее относительно  $u_{i+1}$  можно итерационным методом, но это заметно увеличивает время вычислений на каждом шаге. В док. 6.18 эта трудность преодолена оригинальным, но частным способом: из итерационной формулы путем несложных аналитических преобразований [1] получено значение  $u_{i+1}$ .

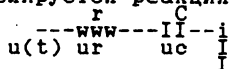
Полезно обратить внимание и на два способа расчета зависимости  $e(t)$ . В первом случае она формируется внутри цикла вычислений, что дает дискретные значения  $e_i$ . Это несколько ускоряет вычисления. Во втором случае  $e(t)$  задана в виде функции пользователя. При обращении к ней аргументом являются дискретные значения времени  $t$ .

Графики переходных процессов в обоих случаях оказываются практически одинаковыми. Отсюда можно сделать вывод, что предпочтителен простой метод Эйлера. Но не стоит спешить с таким выводом — в общем случае комбинированный метод имеет заметные преимущества и позволяет при сохранении заданной погрешности значительно увеличить шаг  $h$  и уменьшить общее время вычислений.

Дальнейшее значительное повышение точности вычислений достигается при решении дифференциального уравнения RC-цепи методом Рунге—Кутты четвертого порядка. Однако реализация такого метода для столь простой задачи чрезмерно сложна. Документ 6.19 реализует еще один оригинальный частный прием [13] — в нем коэффициенты  $K_i$  метода Рунге—Кутты предварительно вычислены аналитически и полученные значения подставлены в конечную итерационную формулу метода — она дана в тексте документа. Это резко упрощает реализацию метода. Документ 6.19 иллюстрирует вычисление реакции RC-цепи на воздействие в виде треугольного импульса.

Обычно применяются линейные RC-цепи. Однако весьма сложно провести и моделирование нелинейных RC-цепей. Например, если нелинейна емкость, то вместо значения  $C = \text{const}$  следует использовать зависимость  $C(u_i)$ .

Моделируется реакция RC-цепи на треугольный импульс



$u_1(p) := \text{if}(p < 1, p, 2 - p)$  · Задание входного сигнала

$u(p) := \text{if}(p > 2, 0, u_1(p))$  —треугольного импульса

$H := 0.1$  Нормированный шаг  $dt/rC$

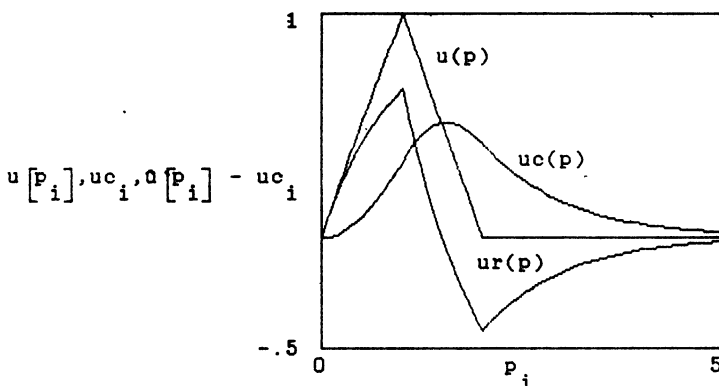
$i := 0 \dots 50$  Задание 50 точек расчета

$uc_0 := 0$  Задание начального значения  $uc$

$p_i := H \cdot i$  Вычисление параметра  $t/rC$

Преобразованная итерационная формула  
метода Рунге-Кутты четвертого порядка

$$uc_{i+1} := uc_i + H \cdot [u[p_i] - uc_i] \cdot \frac{6 - 3H + H^2 - \frac{H^3}{4}}{6}$$



Примечание: данный метод основан на аналитических вычислениях коэффициентов  $k_1, k_2, k_3$  и  $k_4$  в формуле Рунге-Кутты, что позволяет резко упростить итерационную часть метода и увеличить скорость вычислений

Док. 6.19. Моделирование реакции RC-цепи на треугольный импульс

### 6.13. Моделирование цепи с негатроном

Негатроны — двухполюсники, имеющие N- или S-образную вольт-амперную характеристику, содержащую падающий участок. Наиболее известными и распространенными негатронами являются туннельные диоды.

Представим себе цепь, содержащую туннельный диод, который подключен к источнику питания  $E$  через последовательно соединенные резистор  $R$  и катушку индуктивности  $L$ . Учет также емкости туннельного диода и монтажа  $C$ .

Как уже отмечалось (см. док. 6.14), возможны пересечение ВАХ линией нагрузки резистора  $R$  в трех точках и работа схемы в качестве двухстабильного триггера. Здесь выберем  $E$  и  $R$  так, чтобы точка пересечения была одной и располагалась на падающем участке ВАХ. Это позволит убедиться в существовании целого ряда других режимов работы такой схемы, свидетельствующих о ее многофункциональности и многообразии областей применения.

Переходные процессы с момента включения схемы описываются системой из двух дифференциальных уравнений:

$$\begin{aligned} di/dt &= (E - i \cdot R - u) / (R \cdot C), \\ du/dt &= [i - I(u)] / C, \end{aligned}$$

где  $I(u)$  — зависимость тока диода от напряжения, т. е. его ВАХ. Поскольку ВАХ нелинейна, то нелинейна и представленная система дифференциальных уравнений.

Документ 6.20 иллюстрирует моделирование цепи с туннельным диодом. Отметим несколько оригинальных моментов, связанных с реализацией моделирования.

Прежде всего интересно задание ВАХ в табличном виде с применением сплайн-интерполяции и экстраполяции для получения значений  $I(u)$  для любых значений  $u$ . Это придает данному документу универсальность — буквально заменой нескольких координат точек ВАХ можно задать ВАХ любой формы, например для пары Готто (встречно включенные туннельные диоды), N-транзистора, лавинного транзистора [14],  $\Lambda$ -диода и т. д. Конкретно задана типичная ВАХ туннельного диода.

При заданных начальных условиях в цепи возникают незатухающие релаксационные колебания и цепь выполняет функции автоколебательного релаксационного генератора. Фазовый портрет колебаний (он построен точками на графике N-образной ВАХ) вырождается в предельный цикл. Форма колебаний сильно отличается от синусоидальной и имеет быстрые участки, характерные для регенеративных процессов.

Изменив параметры схемы (например, на  $E = 0,35$  В и  $C = 100$  пФ), можно наблюдать вырождение релаксационных колебаний в затухающие синусоидальные колебания, что характерно для применения данной цепи в качестве избирательного усилителя. Несложно (задав  $E$  как функцию времени) промоделировать работу и ждущего релаксационного генератора на туннельном диоде.

Приведенные примеры свидетельствуют о больших возможностях системы MathCAD в моделировании сложных физических явлений, наблюдаемых при работе даже простых электронных схем. В отличие от специальных моделирующих программ (таких как MicroCAP или PSpICE) моделирование с помощью системы MathCAD физически более наглядно, и пользователь может изменять как метод моделирования, так и применяемые аппроксимации нелинейных зависимостей (например ВАХ) по своему усмотрению.

Интерполяция N-образной ВАХ кубическими сплайнами

```

U := [ -0.05
       .2
       .4
       .6
       .8
       1.0
       1.2 ]
I := [ -0.01
       .01
       .004
       .001
       .0009
       .003
       .01 ]
U, I - векторы напряжения и тока
k := 0 .. 6 IS := cspline(U, I)
V := -0.1, -0.05 .. 1.2
J(V) := interp(IS, U, I, V)

```

Исходные данные для моделирования

```

E := 0.3 R := 10 C := 20 * 10^-12 L := 100 * 10^-9
dt := 0.2 * 10^-9 N := 200 k := 0 .. N

```

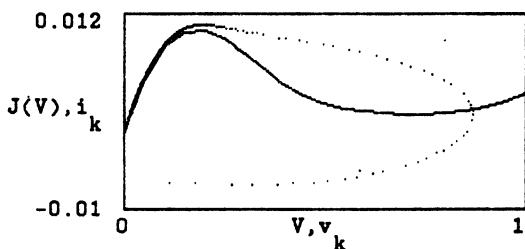
```

[ i
  v ] := [ 0 ]
[ 0 ]      Задание начальных условий

```

Решение системы дифференциальных уравнений состояния

$$\begin{bmatrix} i_{k+1} \\ v_{k+1} \end{bmatrix} := \begin{bmatrix} i_k + \frac{E - i_k \cdot R - v_k}{L} \cdot dt \\ v_k + \frac{i_k - \text{interp}[IS, U, I, v_k]}{C} \cdot dt \end{bmatrix}$$



N-образная  
ВАХ и  
фазовый портрет  
колебаний  
(точками)

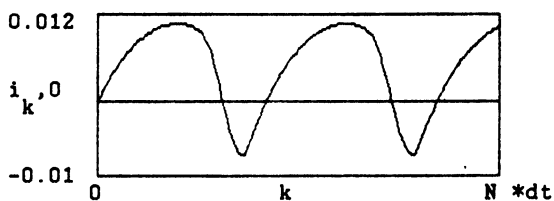


График  $i(t)$

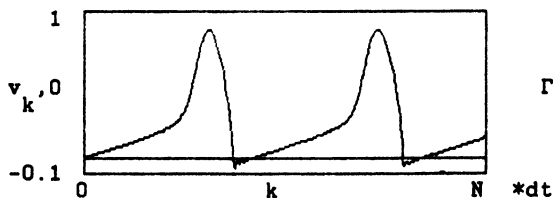


График  $v(t)$

Док. 6.20. Моделирование релаксационных колебаний

## Список литературы

1. **Дьяконов В. П.** Справочник по расчетам на микрокалькуляторах. Изд. 3-е.— М.: Наука, 1989.— 464 с.
2. **Дьяконов В. П.** Справочник по алгоритмам и программам на языке Бейсик для персональных ЭВМ.— М.: Наука, 1987.— 240 с.
3. **Дьяконов В. П.** Применение персональных ЭВМ и программирование на языке Бейсик.— М.: Радио и связь, 1989.— 288 с.
4. **Дьяконов В. П.** Автоматизация математических расчетов с помощью системы MathCAD//Мир ПК.— 1991.— № 8.— С. 43.
5. **Бабенко К. И.** Основы численного анализа.— М.: Наука, 1986.— 744 с.
6. **Бахвалов Н. С., Жидков Н. П., Кобельков Г. Н.** Численные методы.— М.: Наука, 1987.— 600 с.
7. **Иванов В. В.** Методы вычислений на ЭВМ.— Киев: Наукова думка, 1986.— 584 с.
8. **Самарский А. А., Гулин А. В.** Численные методы.— М.: Наука, 1989.— 432 с.
9. **MathCAD.** MathSoft Inc.— Cambrige, Massachusetts, 1987.
10. **Рыбасенко В. Д., Рыбасенко И. Д.** Элементарные функции. Формулы, таблицы, графики.— М.: Наука, 1987.— 416 с.
11. **Бал И., Дж., Гарт Р.** Простые и точные формулы для несимметричной полосковой линии с конечной толщиной полоски//ТИИЭР.— 1977.— Т. 65, № 11.— С. 104.
12. **Зернов Н. В., Карпов В. Г.** Теория радиотехнических цепей.— Л.: Энергия, 1972.— 816 с.
13. **Дьяконов В. П.** Расчет нелинейных и импульсных устройств на программируемых микрокалькуляторах: Справочное пособие.— М.: Радио и связь, 1984.— 176 с.
14. **Дьяконов В. П.** Лавинные транзисторы и их применение в импульсных устройствах.— М.: Сов. радио, 1973.— 208 с.
15. **Браун М.** Система MathCAD становится более гибкой//Мир ПК.— 1991.— № 8.— С. 48.

## Содержание

Предисловие . . . . .	3
<b>1. Основы работы в системе MathCAD . . . . .</b>	<b>4</b>
1.1. Назначение системы . . . . .	4
1.2. Первые шаги . . . . .	5
1.3. Ввод текстов . . . . .	7
1.4. Азы вычислений . . . . .	8
1.5. Построение графиков . . . . .	10
1.6. Редактирование, запись и считывание документов . . . . .	13
<b>2. Управление системой MathCAD . . . . .</b>	<b>14</b>
2.1. Основное меню . . . . .	14
2.2. Системные команды (System) . . . . .	15
2.3. Работа с файлами (File) . . . . .	17
2.4. Режимы работы системы (Compute) . . . . .	17
2.5. Редактирование документов (Edit/move) . . . . .	19
2.6. Редактирование текстов (Text) . . . . .	21
2.7. Управление окнами и форматирование страниц (Window/page) . . . . .	22
2.8. Различия версий системы MathCAD . . . . .	23
<b>3. Алфавит и типы данных системы MathCAD . . . . .</b>	<b>26</b>
3.1. Идентификаторы и алфавит . . . . .	26
3.2. Десятичные, восьмеричные и шестнадцатеричные числа . . . . .	27
3.3. Комплексные числа . . . . .	28
3.4. Константы, переменные и равенства . . . . .	29
3.5. Массивы, матрицы, векторы и таблицы . . . . .	34
3.6. Файлы данных . . . . .	38
3.7. Графические данные . . . . .	38
<b>4. Операторы и функции системы MathCAD . . . . .</b>	<b>41</b>
4.1. Арифметические операторы . . . . .	41
4.2. Встроенные функции и функции пользователя . . . . .	44
4.3. Векторные и матричные операторы и функции . . . . .	44
4.4. Функции линейной и сплайн-интерполяции . . . . .	49
4.5. Функции статистики и линейной регрессии . . . . .	51
4.6. Функции для решения уравнений и минимизации . . . . .	54
4.7. Функции с условными выражениями . . . . .	58
4.8. Функции прямого и обратного быстрого преобразования Фурье . . . . .	61
4.9. Сообщения об ошибках . . . . .	63
4.10. Численные методы, реализованные в системе . . . . .	64
<b>5. Специальные виды математических и физических расчетов . . . . .</b>	<b>67</b>
5.1. Логические функции и их реализация . . . . .	67
5.2. Итерации в векторной форме . . . . .	69



5.3. Решение дифференциального уравнения первого порядка . . . . .	71
5.4. Решение дифференциального уравнения второго порядка . . . . .	75
5.5. Решение систем дифференциальных уравнений . . . . .	77
5.6. Вычисление специальных функций . . . . .	79
5.7. Операции с комплексными числами . . . . .	82
5.8. Графики функций в полярной системе координат . . . . .	84
5.9. Графики на комплексной плоскости . . . . .	85
5.10. Примеры физических расчетов . . . . .	91
<b>6. Расчет электрических и электронных цепей и устройств . . . . .</b>	<b>91</b>
6.1. Простейшие электротехнические расчеты . . . . .	91
6.2. Расчет индуктивных элементов . . . . .	96
6.3. Расчет емкостных элементов . . . . .	100
6.4. Расчет линий передачи . . . . .	106
6.5. Диаграмма Бодэ . . . . .	106
6.6. Преобразование Лапласа . . . . .	108
6.7. Гармонический синтез сигналов . . . . .	109
6.8. Фильтрация сигналов на фоне шумов . . . . .	112
6.9. Расчет нелинейных цепей в статическом режиме . . . . .	114
6.10. Спектральный анализ со сплайн-аппроксимацией передаточной ха- рактеристики . . . . .	117
6.11. Расчет выпрямителя в стационарном режиме . . . . .	119
6.12. Расчет переходных процессов для RC-цепи . . . . .	119
6.13. Моделирование цепи с негатроном . . . . .	124
Список литературы . . . . .	126

## ВНИМАНИЕ

ВЫ МОЖЕТЕ ЗАМЕТНО ОБЛЕГЧИТЬ СЕБЕ РАБОТУ С СИСТЕМОЙ МАТНСАД, ЕСЛИ ПРИОБРЕТЕТЕ ВСЕ ОПИСАННОЕ В ЭТОЙ КНИГЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ. ОНО ПОСТАВЛЯЕТСЯ НА ДИСКЕТКЕ УЧЕБНО-НАУЧНЫМ ЦЕНТРОМ ПО КОМПЬЮТЕРНЫМ И ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ «КИТ». ПРИОБРЕТАЮЩИМ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ГАРАНТИРУЮТСЯ КОНСУЛЬТАЦИИ ПО УСТАНОВКЕ СИСТЕМЫ МАТНСАД И ПОЛУЧЕНИЕ ИНФОРМАЦИИ О НОВЫХ ПАКЕТАХ ПРИМЕНЕНИЯ СИСТЕМЫ И ДРУГИХ ПРОГРАММНЫХ И УЧЕБНО-МЕТОДИЧЕСКИХ РАЗРАБОТКАХ ЦЕНТРА.

ЗА СПРАВКАМИ ОБРАЩАТЬСЯ ПО АДРЕСУ: 214013, СМОЛЕНСК, ЭНЕРГЕТИЧЕСКИЙ ПРОЕЗД, 1, ГНЦ КИТ.

