Глава 2

Некоторые возможности Mathcad

Научившись строить математические выражения, функции пользователя и графики, познакомимся с основными возможностями Mathcad, наиболее часто используемыми функциями и приемами работы.

2.1. Ступенчатые и разрывные функции и выражения. Условие в Mathcad

Используемые в расчетах функции не всегда бывают непрерывными. Часто при разных значениях аргумента функция описывается различными выражениями. Бывают функции ступенчатые или с разрывами. При вычислении производных или интегралов от таких функций приходится их брать по частям. Условный оператор позволяет записать такие функции в виде одного выражения, что упрощает расчеты и украшает Mathcad-документ.

В Mathcad существует три различных способа ввода условий:

- с помощью функции условия if;
- с помощью оператора if с панели программирования;
- с использованием булевых операторов.

Рассмотрим пример вычисления перемещения балки при изгибе с помощью интеграла Мора (рис. 2.1). На балке есть два участка, на которых изгибающий момент описывается различными функциями, $M_1(x)$ и $M_2(x)$.

Для использования функции условия if нужно:

- записать имя выражения и оператор присваивания (:=);
- на стандартной панели нажать кнопку f(x) и в списке встроенных функций выбрать if, после чего нажать кнопку Insert (Вставить). Появится шаблон функции if с тремя местами ввода;
- заполнить места ввода.

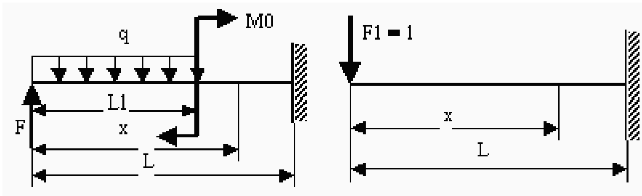
Обращение к функции:

$$\text{if}(\text{cond}, x, y),$$

где cond — условие типа $x \le L_1$, x и y — значения, возвращаемые функцией. Если условие выполняется, то выражению присваивается значение x, если не выполняется, то значение y.

Чтобы записать условный оператор с панели программирования, следует:

- записать имя выражения и оператор присваивания (:=);
- вызвать панель программирования Programming Toolbar нажатием соответствующей кнопки математической панели и щелкнуть мышью на кнопке Add lines (Добавить линию);



Изгибающие моменты от внешних сил

$M1(x) := F\cdot x - q\cdot\frac{x^2}{2}$ $\qquad M2(x) := F\cdot x - q\cdot L1\cdot\left(x - \frac{L1}{2}\right) + M0$

Изгибающий момент от единичной силы $\quad MM(x) := x$

Условный оператор программирования if $\qquad$ Функция условия if

$M(x) := \left| \begin{array}{ll} M1(x) & \text{if } x \le L1 \\ M2(x) & \text{otherwise} \end{array} \right.$ $\qquad M(x) := \text{if}(x \le L1, M1(x), M2(x))$

если то иначе

Применение булевых операторов $\quad M(x) := M1(x)\cdot(x \le L1) + M2(x)\cdot(x > L1)$

Все 3 выражения M(x) эквивалентны

F, q, M0, L1, L - заданы глобально около графика

поперечная сила $\qquad$ прогиб балки $\qquad E := 2\cdot 10^{11} \quad J := 2000\cdot 10^{-8}$

$Q(x) := \frac{d}{dx}M(x) \qquad \Delta := \int_0^L \frac{M(x)\cdot MM(x)}{E\cdot J}\,dx \qquad \Delta = 0.013$

Рис. 2.1. Определение перемещения балки

- в верхнем поле ввода (черный квадратик) ввести выражение для изгибающего момента на первом участке;
- щелкнуть мышью на кнопке if на панели программирования (выражение для изгибающего момента при этом должно быть полностью выделено синим уголком или взято в скобки); появится место ввода, куда надо вписать условие, например, $x \le L_1$ или $0 \le x \le L_1$;
- в нижнем поле ввода (у вертикальной черты) ввести изгибающий момент для второго участка и выделить его целиком синим уголком (клавишей пробела);
- щелкнуть мышью на кнопке Otherwise (Иначе) на панели программирования или выбрать if и вписать условие $x > L_1$. Функция $M(x)$ готова.

Использование логических (булевых) операторов состоит в умножении слагаемых заданного выражения на соответствующий логический оператор. Логические операторы вводятся с панели булевых операторов (кнопка Boolean Toolbar). Булевы операторы могут возвращать только 0 или 1. Если условие справедливо, то значением оператора является 1, если несправедливо, то 0. В математическом выражении умножение на логический оператор обращает соответствующее слагаемое в ноль или не меняет его значения. Пример применения всех трех форм записи условия показан на рис. 2.1.

Схемы грузового и единичного состояния балки, изображенные на рис. 2.1, выполнены автором в Corel Draw. Для вставки любого объекта, подготовленного в другом приложении, в Mathcad надо выбрать пункт меню Insert ▸ Object (Вставка ▸ Объект). Появится стандартное диалоговое окно Windows Вставка объекта. В этом окне надо выбрать команду Создать из файла ▸ Обзор, затем выбрать имя файла, в котором сохранен рисунок. Чтобы в дальнейшем можно было редактировать рисунок, не выходя из оболочки Mathcad, рекомендуется установить флажок Связь в диалоговом окне Вставка объекта. Рисунок появился в документе.

Подробнее об использовании условия в Mathcad говорится в главе 12 (раздел 12.7).

2.2. Глобальное присвоение значений

Посмотрите на функцию $M(x)$ (рис. 2.2), она содержит константы — нагрузки и длины, P, q, M_0 и L_1. До сих пор эти константы не были заданы. Но строгий Mathcad тем не менее не указывает на ошибку. Дело в том, что константы заданы глобально около графика.

$x := 0, \frac{L}{m} .. L$

самостоятельно построить эпюры M и Q

$F := 100 \quad q := 20$

для $x < 2$ $M(x) := F \cdot x$ для $2 \le x < 5$ $M(x) := F \cdot x - \frac{q \cdot (x-2)^2}{2}$

Рис. 2.2. Эпюры внутренних усилий для балки, изображенной на рис. 2.1

Чтобы присвоить некоторой константе глобальное значение, нужно:

- набрать с клавиатуры имя константы;
- в математической панели щелкнуть мышью на кнопке x=;
- щелкнуть на кнопке ≡ в открывшейся панели Evaluation (Вычисление) или нажать комбинацию клавиш Shift+~.

Обратите внимание: если локальный оператор присваивания (:=) действует вправо и вниз от места ввода, то глобальный оператор присваивания (≡) действует по всему документу, и вверх, и вниз. При открытии документа интерпретатор

Mathcad вначале просматривает весь документ сверху вниз, отыскивая глобальные операторы присваивания, затем при втором проходе выполняет локальные присваивания.

СОВЕТ

Следует избегать присвоения одному и тому же имени константы и глобального, и локального значений, так как при этом глобальное значение отменяется локальным.

Глобальные операторы присваивания значений константам удобно размещать вблизи таблиц или графиков, чтобы, изменяя присваиваемое значение, сразу видеть изменение результатов расчета.

В рассматриваемом примере поменяйте значения нагрузок P, q, M_0, длин L_1, L и число интервалов разбиения длины балки n. Самостоятельно постройте и отформатируйте графики для функции, заданной на рис. 2.2.

2.3. Символьные вычисления

Наряду с числовыми расчетами Mathcad может производить вычисления в символьном виде. Существует два способа символьных вычислений:

1. С использованием меню Symbolics (Символьные вычисления) из главного меню Mathcad.
2. С использованием панели Symbolic из математической панели.

Оба способа будут подробно рассмотрены в главе 5. Пример использования меню Symbolics приведен в следующем разделе (рис. 2.4), а пока ограничимся наиболее простым и часто используемым методом — применением символьного знака равенства (→). В качестве примера рассмотрим вычисление неопределенного интеграла (рис. 2.3). Далее описан порядок символьных вычислений:

- В математической панели щелкнуть мышью на кнопке Calculs Toolbar (Панель вычислений) со значком интеграла.
- В открывшейся панели Calculus (Вычисления) выбрать, щелкнув мышью, шаблон неопределенного интеграла.
- Заполнить места ввода — вписать интегрируемое выражение или имя интегрируемой функции, а также имя переменной в шаблоне дифференциала.
- Ввести символьный знак равенства (→). Это можно сделать с помощью двух панелей, вызываемых из математической: Symbolic Toolbar и Evaluation Toolbar. Можно ввести этот оператор и с клавиатуры, нажав комбинацию клавиш Ctrl+. (точка). На экране появится результат символьного вычисления.

Аналогично можно выполнить все операции, предусмотренные на панели Calculus, а именно: вычисление производной любого заданного порядка, определенного и неопределенного интегралов, суммы или произведения ряда, предела выражения. Решите самостоятельно приведенные на рис. 2.3 примеры.

$p := 2 \qquad p := p$ отмена предыдущего численного значения p

$$f(z) := \sin(g \cdot z) \cdot \cos\left(\frac{z}{p}\right)$$

вместо p1 поставьте p,
оно будет заменено числом

$$\frac{d}{dv}(\sin(v) \cdot \cos(v)) \rightarrow \cos(v)^2 - \sin(v)^2$$

$$\int f(z)\,dz \rightarrow \frac{-1}{2} \cdot \frac{\cos\left(\frac{g \cdot p + 1}{p} \cdot z\right)}{g \cdot p + 1} \cdot p - \frac{1}{2} \cdot \frac{\cos\left(\frac{g \cdot p - 1}{p} \cdot z\right)}{g \cdot p - 1} \cdot p$$

$$\sum_{n=1}^{\infty} \frac{1}{n^2} \rightarrow \frac{1}{6} \cdot \pi^2 \qquad \lim_{t \to \infty} \frac{\sin(3 \cdot t)^2}{\ln(1 + 2 \cdot t)^2} \rightarrow 0$$

самостоятельно проинтегрируйте и продифференцируйте функции

$$\tan(t)^2 \qquad (\tan(t) + \cot(t))^2 \qquad e^{3 \cdot t} \cdot 3^t$$

вычислите сумму и произведение членов ряда при n, стремящемся к бесконечности $\quad \frac{2 \cdot n + 1}{n^2 \cdot (n+1)^2}$

вычислить пределы функций при t, стремящемся к нулю $\quad \frac{\sin(t) - \operatorname{atan}(t)}{t^3}$

Рис. 2.3. Символьное вычисление производной, интеграла, сумм, произведений, пределов

Недостаток использования символьного знака равенства заключается в том, что величины, которым были ранее присвоены численные значения, сохраняют их и при символьном вычислении, то есть вместо символьного вычисления получается численное.

СОВЕТ

Чтобы получить символьное вычисление вместо численного, необходимо отменить предыдущие численные значения всех параметров, присвоив каждому параметру значение самого себя, например: x := x (см. рис. 2.3).

Достоинство использования символьного знака равенства в том, что найденное решение пересчитывается автоматически при изменении выражения или входящих в него величин и участвует в последующих расчетах.

2.4. Решение уравнений

Mathcad позволяет решить любое алгебраическое, а также многие дифференциальные и интегральные уравнения. Произвольно, «с потолка» взятые дифференциальные и интегральные уравнения и системы уравнений вообще могут не иметь решений, и Mathcad не в силах сотворить чудо.

Для примера возьмем квадратное уравнение и найдем его решение вначале символьным, затем численным способом.

2.4.1. Символьное решение

Для символьного решения уравнения нужно:

- ❑ набрать решаемое уравнение и синим уголком курсора выделить переменную, относительно которой нужно решить уравнение;
- ❑ в главном меню выбрать команду Symbolics ▸ Variable ▸ Solve (Символьные вычисления ▸ Переменная ▸ Решить). Появится ответ (рис. 2.4).

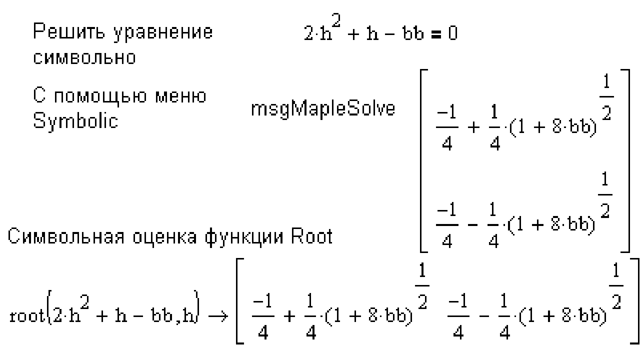


Рис. 2.4. Символьное решение уравнения

Недостаток использования меню Symbolics заключается в том, что найденное решение не пересчитывается автоматически при изменении выражения или входящих в него величин и не участвует в последующих расчетах.

Достоинством использования меню Symbolics является то, что ранее принятые численные значения величин не учитываются в символьных расчетах.

Если выделенное выражение не имеет символьного решения (а большинство уравнений не имеет символьного решения), то Mathcad сообщает об ошибке: «No solution was found» («Решение не найдено»).

В Mathcad есть и другие возможности символьного решения уравнений, например, с использованием функции root (см. рис. 2.4, *внизу*). О прочих возможностях символьных вычислений говорится в главе 5.

2.4.2. Численное решение (функция root)

Рассмотрим одну из ряда функций, позволяющих решать алгебраические уравнения, — функцию root.

Обращение к функции:

$$\mathrm{root}(f(x), x),$$

где $f(x) = 0$. Возвращает значение x, при котором функция $f(x) = 0$.

Функция root решает уравнения итерационным методом секущих и поэтому требует задания перед собой начальных значений. Кроме того, функция root, выполняя вычисления методом спуска, находит и выводит только один корень, ближайший к начальному приближению.

Прежде чем решать уравнение, желательно построить график функции $f(x)$ (рис. 2.5). На графике видно, пересекает ли кривая $f(x)$ ось абсцисс, то есть имеет ли действительные корни. Если точки пересечения кривой с осью есть, нужно выбирать начальное приближение поближе к значению корня. Если корней несколько, для нахождения каждого корня надо задавать свое начальное приближение.

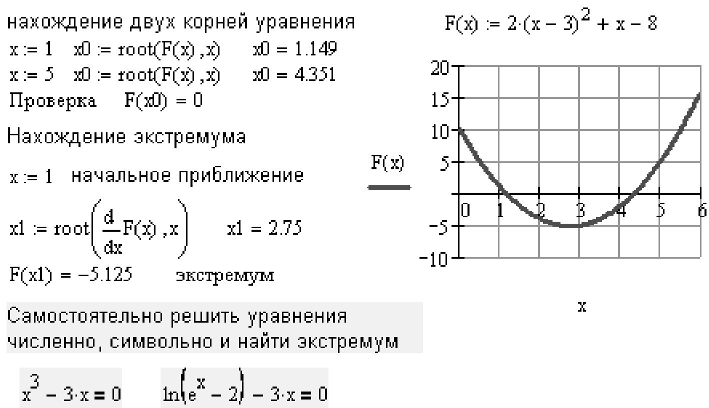


Рис. 2.5. Численное решение уравнения

Если же точек пересечения нет, то корни уравнения могут быть только мнимыми числами. Для их нахождения нужно задавать начальное приближение в комплексной форме. В результате расчета корни уравнения появляются в комплексном виде.

ВНИМАНИЕ

Для ввода мнимой единицы надо на клавиатуре набрать $1i$ или $1j$. При выходе из области мнимого числа единица исчезает, и мы видим комплексное число в обычном виде.

С помощью функции root можно найти и экстремум функции, приравняв производную к нулю (см. рис. 2.5). Функции

$$\mathrm{root}\left(\frac{df(x)}{dx},\ x\right)$$

должно предшествовать начальное приближение (по умолчанию $\dfrac{df(x)}{dx} = 0$).

Для нахождения экстремума функции следует:

- задать начальное приближение поближе к экстремуму;
- записать выражение с функцией root, включив в качестве функции, которая должна быть равна нулю, производную по заданной переменной;
- вычислить значение заданной функции от найденного корня уравнения $\frac{df(x)}{dx}=0$. Экстремум найден (см. рис. 2.5).

Функция root в старых версиях (до Mathcad 12) позволяет решить уравнение и в символьном виде. При этом задавать начальное приближение не требуется, нужно лишь ввести выражение, содержащее функцию root, и выбрать символьный знак равенства, нажав комбинацию клавиш Ctrl+. (точка). Если символьное решение существует, появится ответ, содержащий сразу все корни уравнения, а не один, как при численном решении уравнения (см. рис. 2.4).

2.5. Решение систем уравнений (функция find)

В Mathcad системы уравнений решаются с помощью вычислительного блока Given–find. Так как системы уравнений решаются итерационным методом, перед решением необходимо задать начальные приближения для всех неизвестных.

Чтобы решить систему алгебраических уравнений, нужно:

- задать начальные приближения для всех неизвестных, входящих в систему;
- напечатать ключевое слово Given (Дано). Убедитесь, что при печати вы не находитесь в текстовой области. Если нажать клавишу пробела, то математическое выражение становится текстовой областью и слово Given перестает восприниматься как ключевое;
- ввести уравнения и неравенства, входящие в систему, правее и ниже ключевого слова Given. Между левой и правой частями уравнения должен стоять знак равенства. Это не знак присвоения значения, а знак логического равенства. Для его ввода используйте комбинацию клавиш Ctrl+= или выберите его на панели Boolean (Булевы операторы);
- введите любое выражение, содержащее функцию find. При печати слов Given и find можно использовать любой шрифт, прописные и строчные буквы.

Обращение к функции:

$$\mathrm{find}(x, y, z\ldots),$$

где x, y, z — неизвестные. Количество неизвестных должно быть равно количеству уравнений. Возвращает значения неизвестных x, y, z, обращающих уравнения в тождества.

Функция find может решать и одно уравнение с одним неизвестным как частный случай системы уравнений. Для системы из нескольких уравнений функция find выводит решение в виде вектора.

Пример решения системы уравнений приведен на рис. 2.6. Заданная система уравнений состоит из уравнения окружности и уравнения прямой. Если прямая и окружность пересекаются (рис. 2.8), то для нахождения двух точек пересечения нужно задавать два разных начальных приближения. Если окружность и прямая не пересекаются, действительных корней нет. Для нахождения мнимых корней следует задавать начальное приближение в виде комплексного числа.

окружность и прямая пересекаются - действительные решения есть

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

Given

$$x^2 + y^2 = 36 \qquad x + y = 2 \qquad F := \mathrm{Find}(x,y) \qquad F = \begin{pmatrix} 5.123 \\ -3.123 \end{pmatrix}$$

Для вывода второго корня надо задаться другим начальным приближением x= -1 y=0

Если окружность и прямая не пересекаются - действительных решений нет

$x := 1 \quad y := 1$ начальное приближение

$$\text{Given} \qquad x^2 + y^2 = 36 \qquad x + y = 10 \qquad \boxed{F := \mathrm{Find}(x,y)} \qquad F = \blacksquare$$

No solution was found. Try changing the guess value or the value of TOL or CTOL.

$x := i \quad y := 0$ начальное приближение комплексное число (введите x=1i)

$$\text{Given} \qquad x^2 + y^2 = 36 \qquad x + y = 10 \qquad F := \mathrm{Find}(x,y) \qquad F = \begin{pmatrix} 5 - 2.646i \\ 5 + 2.646i \end{pmatrix}$$

Рис. 2.6. Решение системы уравнений

Функция find, так же как и функция root, позволяет решать системы уравнений и в численном, и в символьном виде (рис. 2.7).

ПРИМЕЧАНИЕ

В Mathcad 13 и 14 функция root не решает уравнения в символьном виде.

Given - начало ввода системы уравнений

$$x^2 + y^2 = aa \qquad x + y = bb$$

$$\mathrm{Find}(x,y) \rightarrow \begin{bmatrix} \frac{1}{2}\cdot bb - \frac{1}{2}\cdot\left(-bb^2 + 2\cdot aa\right)^{\frac{1}{2}} & \frac{1}{2}\cdot bb + \frac{1}{2}\cdot\left(-bb^2 + 2\cdot aa\right)^{\frac{1}{2}} \\ \frac{1}{2}\cdot bb + \frac{1}{2}\cdot\left(-bb^2 + 2\cdot aa\right)^{\frac{1}{2}} & \frac{1}{2}\cdot bb - \frac{1}{2}\cdot\left(-bb^2 + 2\cdot aa\right)^{\frac{1}{2}} \end{bmatrix}$$

Рис. 2.7. Символьное решение системы уравнений

Если система уравнений не имеет точного решения, для приближенного решения в Mathcad существует функция minerr. Обращение к ней и использование функции не отличаются от обращения к функции find и ее использования.

На рис. 2.8 представлена геометрическая интерпретация системы уравнений в виде окружности и прямой линии. Если прямая пересекает окружность, корни системы уравнений действительные, если не пересекает — мнимые.

2.6. Приближенное решение систем уравнений (функция minerr)

Для приближенного решения систем уравнений используется вычислительный блок Given–minerr. Обращение к нему аналогично обращению к блоку Given–find.

Если решение системы уравнений существует, $x + y = 2$ (рис. 2.6 и пунктирная линия на рис. 2.8), функция minerr дает тот же ответ, что и функция find. Если решение системы уравнений не существует, функция minerr возвращает минимум невязки решения (рис. 2.8, сплошная линия). В случае $x + y = 10$ функция minerr возвращает значения координат точки, при которых расстояние между кривыми минимально.

$x := 1 \quad y := 1$ Приближенное решение

Given $\quad x^2 + y^2 = 36 \qquad x + y = 10 \qquad F := \mathrm{Minerr}(x, y) \qquad F = \begin{pmatrix} 4.274 \\ 4.22 \end{pmatrix}$

Проверим решение системы уравнений с помощью графика

$n := 50 \qquad \phi := 0, 2\cdot\frac{\pi}{n} .. 2\cdot\pi \qquad x1 := -5, -5 + \frac{6}{n} .. 8$

параметрическое уравнение окружности $\quad y(\phi) := 6\cdot\sin(\phi)$

$x(\phi) := 6\cdot\cos(\phi)$

уравнение прямой $\quad y1(x1) := 10 - x1 \qquad y2(x1) := 2 - x1$

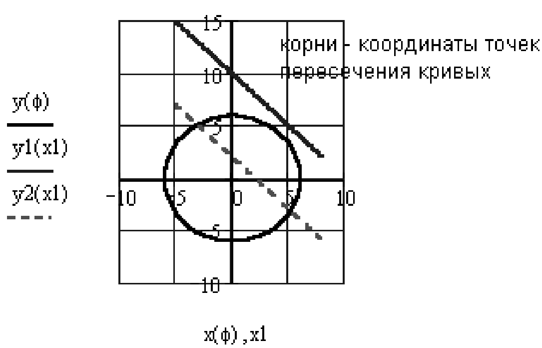


Рис. 2.8. Приближенное решение системы уравнений, если прямая не пересекается с окружностью

Функцию minerr удобно использовать для поиска экстремума негладких функций с переломами на графике.

2.7. Исследование функции на экстремум

Отметим четыре пути поиска экстремума:

1. Для непрерывной функции используем равенство нулю производной от заданной функции. В этой процедуре используют функцию root.
2. Для функции с переломами используем функцию minerr. Для этого по графику выбираем число заведомо большее (или меньшее) экстремального значения функции и записываем его в качестве ограничения в блоке Given–minerr. Функция minerr возвращает значение аргумента, при котором расхождение между заданным числом и значением функции минимально. Возвращаемый результат зависит от выбора начального приближения.
3. Для непрерывных функций удобно использовать функции maximize и minimize (они вводятся аналогично функции find). Ключевое слово given обычно можно опускать — оно необходимо лишь при наличии ограничений.
4. Для ступенчатых функций целесообразно использовать функцию автора $F_{\max}$, реализующую простой метод перебора значений функции (функция $F_{\max}$ описана в разделе 9.8). Другого средства определения максимума ступенчатой функции в Mathcad нет.

СОВЕТ

При анализе конкретного уравнения рекомендуется внимательно изучить поверхностный график функции, на котором хорошо видны области нахождения экстремумов.

Все четыре способа определения экстремумов функции имеют свои плюсы и минусы. Творческий подход к их выбору почти всегда позволяет правильно найти экстремумы функции.

На рис. 2.9 в качестве функции рассмотрена кубическая парабола и показан ее график с двумя экстремумами (максимумом и минимумом). На рис. 2.10–2.12 показано определение этих экстремумов разными способами. Результаты расчета, естественно, совпадают.

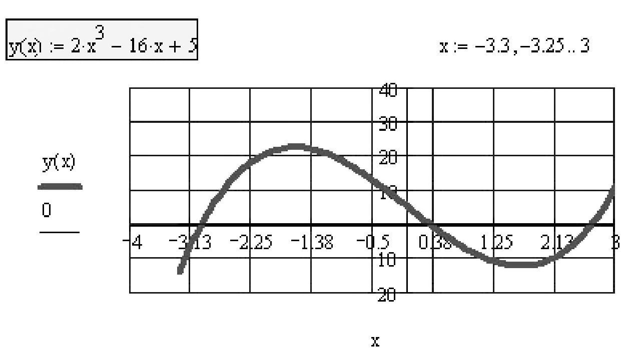


Рис. 2.9. Заданная функция и ее график

$x := -3$ начальное значение $x_{max} := \mathrm{root}\left(\frac{d}{dx}y(x), x\right)$

$x_{max} = -1.633$ $y(x_{max}) = 22.419$ максимум

$x := 3$ другое начальное приближение $x_{min} := \mathrm{root}\left(\frac{d}{dx}y(x), x\right)$

$x_{min} = 1.633$ $y(x_{min}) = -12.419$ минимум

Рис. 2.10. Нахождение экстремума функции путем приравнивания к нулю ее первой производной

$x := -3$ начальное приближение

Given $y(x) = 30$ $x_{max} := \mathrm{Minerr}(x)$

$x_{max} = -1.633$ $y(x_{max}) = 22.419$ максимум

$x := 3$ начальное приближение

Given $y(x) = -20$ $x_{min} := \mathrm{Minerr}(x)$

$x_{min} = 1.633$ $y(x_{min}) = -12.419$ минимум

Рис. 2.11. Нахождение экстремума функции с использованием функции minerr

ключевое слово given необходимо
только в случае наличия ограничений

$x := -3$ начальное значение $x_{max} := \mathrm{Maximize}(y, x)$

$x_{max} = -1.633$ $y(x_{max}) = 22.419$ максимум

$x := 3$ Given $x > 0$ $x_{min} := \mathrm{Minimize}(y, x)$

$x_{min} = 1.633$ $y(x_{min}) = -12.419$ минимум

Рис. 2.12. Нахождение экстремума функции с помощью функций minimize и maximize

Подробнее о нахождении экстремума функции говорится в разделе 3.7.

2.8. Работа с матрицами

Преимущества Mathcad особенно явно видны при работе с матрицами. Операции с матрицами трудоемки и, как правило, требуют компьютерного программирования. В системе Mathcad мы видим традиционную запись матричных выражений, как на листе бумаги, но уже с готовым численным или символьным ответом.

2.8.1. Создание матриц

Чтобы определить вектор или матрицу, следует:

- записать имя матрицы, ввести оператор присваивания (:=);
- в математическом меню выбрать кнопку с изображением матрицы. Откроется панель Matrix, на которой нужно вновь выбрать кнопку с изображением матрицы. На этот раз откроется диалоговое окно, в котором надо ввести число

строк и число столбцов матрицы и нажать кнопку OK. На экране появится шаблон матрицы. То же действие вызывается нажатием комбинации клавиш Ctrl+m;

- каждое место ввода в шаблоне заполнить числами или буквенными выражениями. Матрица готова.

С помощью шаблона можно ввести матрицу, содержащую не более 100 элементов. Как ввести матрицы больших размеров, рассказано в главе 4 (раздел 4.1).

Вектор — это матрица, состоящая из одного столбца.

Доступ к любому элементу матрицы можно получить, задав имя матрицы с двумя индексами. Первый индекс обозначает номер строки, второй — номер столбца. Произвольный элемент вектора задается одним индексом.

Для набора нижнего индекса можно открыть панель Vector and Matrix Toolbar нажатием соответствующей кнопки на математической панели, после чего щелкнуть на кнопке X_n (Subscript), но лучше использовать клавишу [(открывающая квадратная скобка), так как при работе с матрицами ставить нижний индекс приходится очень часто.

Нумерация элементов массива

Нумерация элементов массива (вектора или матрицы) может начинаться с 0, 1 или с любого другого числа (положительного или отрицательного). Порядком нумерации элементов массива управляет встроенная переменная ORIGIN. По умолчанию ORIGIN = 0. Это означает, что первый элемент массива имеет номер 0.

Чтобы нумерация членов векторов и матриц начиналась, как обычно принимается в математике, с 1, нужно перед вводом матрицы, а лучше в начале документа, напечатать ORIGIN:=1 (все буквы прописные).

На рис. 2.13, *вверху* показано создание элементов матрицы D по формуле с использованием нижних индексов. По умолчанию ORIGIN = 0, поэтому $D_{0,0} = 10$. После ввода ORIGIN:=1 элемент $D_{0,0}$ не имеет смысла, а $D_{1,1} = 10$.

2.8.2. Основные действия с матрицами

Mathcad позволяет выполнять с матрицами основные арифметические действия: сложение, вычитание, умножение, а также операции транспонирования, обращения, вычисления определителя матрицы, нахождения собственных чисел и собственных векторов и т. д. Примеры численного и символьного выполнения этих операций приведены на рис. 2.13–2.15.

СОВЕТ

Во время работы с матрицами внимательно следите за размерами матриц. При появлении сообщения о несоответствии размеров матриц (рис. 2.13) напишите в сторонке имя матрицы и нажмите клавишу =. Если размер матрицы отличается от того, что вы вводили, обнулите матрицу перед очередным оператором присваивания, написав, например, A:=0.

$$\mathrm{ORIGIN} = 0 \quad \text{(по умолчанию)} \qquad i := 0..2 \qquad j := 0..4$$

$$D_{i,j} := 10 - i - j \qquad D = \begin{pmatrix} 10 & 9 & 8 & 7 & 6 \\ 9 & 8 & 7 & 6 & 5 \\ 8 & 7 & 6 & 5 & 4 \end{pmatrix} \qquad D := D^T \quad D = \begin{pmatrix} 10 & 9 & 8 \\ 9 & 8 & 7 \\ 8 & 7 & 6 \\ 7 & 6 & 5 \\ 6 & 5 & 4 \end{pmatrix}$$

$$D_{0,0} = 10 \qquad \mathrm{ORIGIN} := 1 \qquad \boxed{D_{0,0} = \blacksquare} \qquad D_{1,1} = 10$$

Value of subscript or superscript is too big (or too small) for this array.

$$B := \begin{pmatrix} 3 & 4 & 5 \\ 4 & 5 & 1 \\ 5 & 1 & 2 \\ 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \qquad B + D = \begin{pmatrix} 13 & 13 & 13 \\ 13 & 13 & 8 \\ 13 & 8 & 8 \\ 8 & 8 & 8 \\ 8 & 8 & 8 \end{pmatrix} \qquad B - D = \begin{pmatrix} -7 & -5 & -3 \\ -5 & -3 & -6 \\ -3 & -6 & -4 \\ -6 & -4 & -2 \\ -4 & -2 & 0 \end{pmatrix}$$

$$B \cdot D^T = \begin{pmatrix} 106 & 94 & 82 & 70 & 58 \\ 93 & 83 & 73 & 63 & 53 \\ 75 & 67 & 59 & 51 & 43 \\ 52 & 46 & 40 & 34 & 28 \\ 79 & 70 & 61 & 52 & 43 \end{pmatrix} \qquad B^T \cdot D = \begin{pmatrix} 125 & 110 & 95 \\ 125 & 110 & 95 \\ 120 & 105 & 90 \end{pmatrix}$$

$$B \cdot D = \blacksquare$$

The number of rows and/or columns in these arrays do not match.

Рис. 2.13. Арифметические действия с матрицами

$$V := \begin{pmatrix} 12 \\ 11 \\ 10 \\ 9 \\ 8 \end{pmatrix} \qquad V \cdot V^T = \begin{pmatrix} 144 & 132 & 120 & 108 & 96 \\ 132 & 121 & 110 & 99 & 88 \\ 120 & 110 & 100 & 90 & 80 \\ 108 & 99 & 90 & 81 & 72 \\ 96 & 88 & 80 & 72 & 64 \end{pmatrix} \qquad \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{pmatrix}$$

столбец x строку = матрица

$$V^T \cdot V = (510) \qquad \begin{pmatrix} 1 & 2 & 3 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} = (14)$$

строка x столбец = число

Рис. 2.14. Произведение векторов

2.8.3. Решение матричных уравнений

Матричные уравнения представляют собой, как правило, систему линейных алгебраических уравнений $A \cdot X = B$ и решаются путем обращения матрицы коэффициентов $X = A^{-1} \cdot B$ (рис. 2.16).

Транспонирование матрицы

$$D := \begin{pmatrix} 5 & 3 \\ 7 & 1 \\ 4 & 2 \end{pmatrix} \quad C := \begin{pmatrix} 6 & 8 & 2 \\ 3 & 5 & 1 \\ 2 & 3 & 7 \end{pmatrix} \quad D^T = \begin{pmatrix} 5 & 7 & 4 \\ 3 & 1 & 2 \end{pmatrix} \quad C^T = \begin{pmatrix} 6 & 3 & 2 \\ 8 & 5 & 3 \\ 2 & 1 & 7 \end{pmatrix}$$

Определитель квадратной матрицы $|C| = 38$

$\boxed{|D| = \blacksquare}$ D - неквадратная матрица

This matrix must be square. It should have the same number of rows as columns.

Обращение матрицы

$$C^{-1} = \begin{pmatrix} 0.842 & -1.316 & -0.053 \\ -0.5 & 1 & 0 \\ -0.026 & -0.053 & 0.158 \end{pmatrix} \quad \text{Проверка} \quad C \cdot C^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Единичная матрица $E := \mathrm{identity}(5)$

$$E = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Рис. 2.15. Операции с матрицами

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{pmatrix} \quad B := \begin{pmatrix} 10 \\ 20 \\ 50 \end{pmatrix} \quad X := A^{-1} \cdot B \quad X = \begin{pmatrix} 17.5 \\ -22.5 \\ 12.5 \end{pmatrix}$$

Рис. 2.16. Решение системы алгебраических линейных уравнений путем обращения матрицы коэффициентов

Символьные операции с матрицами можно выполнять с помощью меню Symbolics, а также символьного знака равенства. В примерах, приведенных на рис. 2.17, используется только символьный знак равенства $\rightarrow$. Подробно о символьных операциях с матрицами рассказывается в главе 5, разделы 5.2.13 и 5.3.

При выполнении символьных операций с матрицами необходимо помнить, что если какому-либо символу ранее присвоено численное значение, то при использовании символьного знака равенства этот символ участвует в символьных расчетах как число. Если символу ранее присвоено значение вектора или матрицы, то символьные вычисления с его участием становятся невозможными. В этом случае нужно использовать для символьных вычислений меню Symbolics.

2.9. Оператор векторизации

Mathcad допускает вводить в качестве аргумента функции не только числа, но и векторы. При этом вычисляется значение функции для всех элементов вектора.

$$A := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad B := \begin{pmatrix} ee \\ ff \end{pmatrix} \quad X := \begin{pmatrix} s \\ t \end{pmatrix} \quad B \cdot X^T \to \begin{pmatrix} ee \cdot s & ee \cdot t \\ ff \cdot s & ff \cdot t \end{pmatrix}$$

транспонирование — произведение векторов и матриц

$$A^T \to \begin{pmatrix} a & c \\ b & d \end{pmatrix} \quad A \cdot X \to \begin{pmatrix} a \cdot s + b \cdot t \\ c \cdot s + d \cdot t \end{pmatrix} \quad B^T \cdot X \to ee \cdot s + ff \cdot t$$

обратная матрица

$$A^{-1} \to \begin{bmatrix} \dfrac{d}{(a \cdot d - b \cdot c)} & \dfrac{-b}{(a \cdot d - b \cdot c)} \\ \dfrac{-c}{(a \cdot d - b \cdot c)} & \dfrac{a}{(a \cdot d - b \cdot c)} \end{bmatrix}$$

определитель

$$|A| \to a \cdot d - b \cdot c$$

собственные числа

$$\mathrm{eigenvals}(A) \to \begin{bmatrix} \dfrac{1}{2} \cdot a + \dfrac{1}{2} \cdot d + \dfrac{1}{2} \cdot \left(a^2 - 2 \cdot a \cdot d + d^2 + 4 \cdot b \cdot c\right)^{\frac{1}{2}} \\ \dfrac{1}{2} \cdot a + \dfrac{1}{2} \cdot d - \dfrac{1}{2} \cdot \left(a^2 - 2 \cdot a \cdot d + d^2 + 4 \cdot b \cdot c\right)^{\frac{1}{2}} \end{bmatrix}$$

Рис. 2.17. Символьные операции с матрицами

Если аргумент функции — матрица, то, чтобы вычислить значения функции для всех элементов матрицы, надо использовать оператор векторизации (рис. 2.18).

Для использования оператора векторизации нужно:

- ввести выражение или функцию;
- выделить синим уголком необходимую часть выражения (чаще всего выражение целиком);
- на математической панели щелкнуть на кнопке Vector and Matrix Toolbar, а в открывшейся панели — на кнопке Vectorize ($\overrightarrow{f(M)}$) (Векторизация). Над выделенной частью выражения появится стрелка — символ операции векторизации;
- нажать клавишу =.

Оператор векторизации изменяет смысл векторной или матричной операции. Векторизация означает выполнение однотипной операции, предписанной выражением, со всеми элементами массива. Например, $\sqrt{A}$ — операция невозможная, если A — вектор или матрица. Как уже сказано, аргумент функции может быть вектором, и функция, как и в случае использования дискретной переменной, вычисляется для всех элементов вектора. Если аргумент функции — матрица, необходимо применение оператора векторизации, чтобы выполнить то же самое действие, то есть вычислить функцию для всех элементов матрицы. В нашем случае это нахождение корня квадратного из каждого элемента матрицы A. В случае перемножения матриц $A \cdot B$ это матричное произведение, а $\overrightarrow{A \cdot B}$ — это попарное произведение элементов матриц A и B с одинаковыми индексами. Все массивы под знаком векторизации должны быть одного размера, так как операция над всеми массивами производится поэлементно. Примеры использования векторов или матриц в качестве аргументов функции приведены на рис. 2.18.

$$B := \begin{pmatrix} 0.5 \\ 0.3 \\ 0.8 \end{pmatrix} \quad BB := \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix} \quad \sin(B) = \begin{pmatrix} 0.479 \\ 0.296 \\ 0.717 \end{pmatrix} \quad \overrightarrow{\sin(BB)} = \begin{pmatrix} 0.841 & -0.757 \\ 0.909 & -0.959 \\ 0.141 & -0.279 \end{pmatrix}$$

$$\boxed{\sin(BB) = \blacksquare}$$

This must be either a scalar or a vector.

$$A := \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad B := \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix} \quad A \cdot B = \begin{pmatrix} 19 & 22 \\ 43 & 50 \end{pmatrix} \quad \overrightarrow{(A \cdot B)} = \begin{pmatrix} 5 & 12 \\ 21 & 32 \end{pmatrix}$$

$$\overrightarrow{\exp(A + B)} = \begin{pmatrix} 403.429 & 2.981 \times 10^{3} \\ 2.203 \times 10^{4} & 1.628 \times 10^{5} \end{pmatrix} \quad \overrightarrow{\frac{A}{B}} = \begin{pmatrix} 0.2 & 0.333 \\ 0.429 & 0.5 \end{pmatrix}$$

Рис. 2.18. Операция векторизации

ВНИМАНИЕ

Если аргумент — вектор, векторизация не нужна. Если аргумент — матрица, векторизация нужна.

2.10. Решение дифференциальных уравнений

Математически решение дифференциальных уравнений — очень сложная проблема. Mathcad не в состоянии решить без дополнительных упрощений многие дифференциальные уравнения и их системы. Все, что Mathcad может сделать с ними, подробно описано в главе 6. Здесь же рассмотрим использование лишь функции Odesolve.

Имя функции Odesolve можно писать и с прописной, и со строчной буквы. Алгоритм функции Odesolve использует большинство имеющихся в Mathcad функций решения дифференциальных уравнений, фактически заменяя их. Odesolve может решать и системы дифференциальных уравнений. В контекстном меню есть возможность выбора метода решения дифференциальных уравнений.

Функция Odesolve позволяет записывать уравнение в блоке решения в привычном виде, как обычно записывают уравнение на листе бумаги.

Обращение к функции Odesolve требует записи вычислительного блока, который содержит три части:

1. Ключевое слово Given.
2. Дифференциальное уравнение и начальные или граничные условия для него или система дифференциальных уравнений и ее условия.
3. Функция Odesolve(x, xk, n), где x — имя переменной, относительно которой решается уравнение; xk — конец интервала интегрирования. Начало интервала интегрирования указано ранее, в начальных условиях; n — необязательный внутренний параметр, определяющий число шагов интегрирования,

используемый при интерполяции решения, то есть при переходе от матрицы численных значений к функции. Параметр n не является обязательным. Его можно удалить, предоставив Mathcad возможность самому выбирать число шагов интегрирования. По умолчанию Mathcad использует $n = 1000$.

ВНИМАНИЕ

Появление других математических выражений в вычислительном блоке между словами Given и Odesolve недопустимо. Текстовую область внутри вычислительного блока размещать можно. Граничные условия можно задавать лишь в двух точках, одна из которых — начало интервала интегрирования.

Примеры использования функции Odesolve приведены на рис. 2.19–2.22.

(исходное уравнение)

$$\frac{d^3}{dx^3}y(x) + x^2 \cdot \frac{d}{dx}y(x) + x \cdot y(x) = e^x \cdot \cos(x)$$

Дифференциальное уравнение с начальными условиями

Given $\quad y'''(x) + x^2 \cdot y'(x) + x \cdot y(x) = e^x \cdot \cos(x)$

$y(0) = -8 \qquad y'(0) = 3 \qquad y''(0) = 3$

$y := \text{Odesolve}(x, 5) \qquad x := 0, 0.1 .. 5$

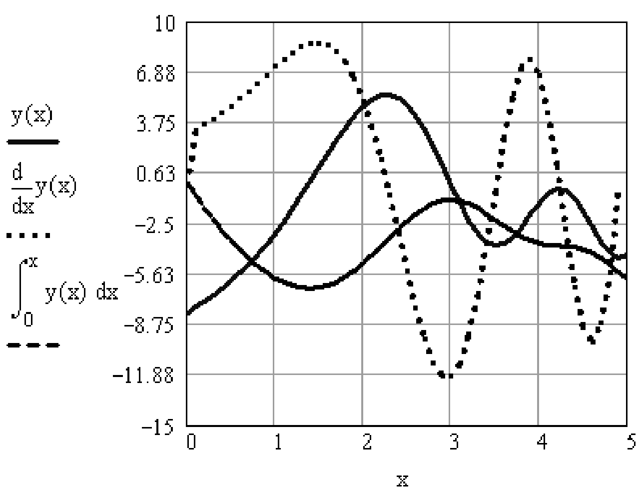


Рис. 2.19. Решение дифференциального уравнения с начальными условиями

дано уравнение $4 \cdot \frac{d^2}{dt^2} x(t) + x(t) = t$

граничные условия $x(C) = A \quad x(D) = B$

$A := 4 \qquad B := 10 \qquad C := -3 \quad D := 6$

Поменяйте граничные условия A,B,C,D

решение уравнения

Given $\quad 4 \cdot \frac{d^2}{dt^2} x(t) + x(t) = t$

$x(C) = A \qquad x(D) = B \qquad x := \mathrm{Odesolve}(t, D + 2)$

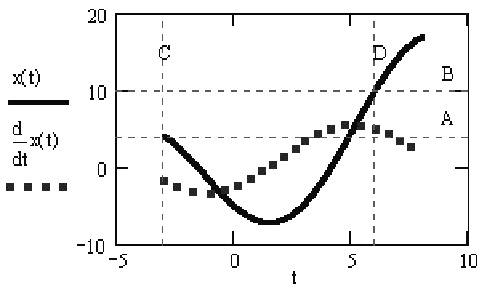


Рис. 2.20. Решение дифференциального уравнения с граничными условиями

ПРИМЕЧАНИЕ

Исходное уравнение можно записывать с использованием как оператора дифференцирования, так и штриха (рис. 2.21). Граничные условия следует записывать только со штрихом. Для набора штриха используйте комбинацию клавиш Ctrl+F7.

Функция Odesolve возвращает решение дифференциального уравнения в виде функции, а не в виде массива, как все остальные функции, описанные в главе 6, поэтому найденное решение можно интегрировать и дифференцировать (см. рис. 2.19–2.21), а также использовать в последующих расчетах как функцию пользователя.

Функция Odesolve решает дифференциальные уравнения как с начальными условиями, когда все условия заданы в начале интервала интегрирования, так и с граничными условиями, заданными в двух точках. Из этих двух точек одна обязательно является началом интервала интегрирования, другая произвольная, но ее аргумент больше, чем в начальной точке. Решение уравнения с начальными условиями показано на рис. 2.19, уравнения с граничными условиями — на рис. 2.20. Решение дифференциального уравнения высокого порядка (4-го) продемонстрировано на рис. 2.21.

$k := 3 \qquad y''''(x) - 2 \cdot k^2 y''(x) + k^4 \cdot y(x) = 0$

Given $\quad \dfrac{d^4}{dx^4} y(x) - 2 \cdot k^2 \cdot \dfrac{d^2}{dx^2} y(x) + k^4 \cdot y(x) = 0$

$y(3) = 0 \qquad y'(3) = 0 \qquad y''(0) = -1 \quad y'''(0) = 3$

$Y := \mathrm{Odesolve}(x, 10)$

$Y1(x) := \dfrac{d}{dx} Y(x) \qquad Y2(x) := \displaystyle\int_0^x Y(x)\,dx \qquad x := 0, 0.1 .. 5$

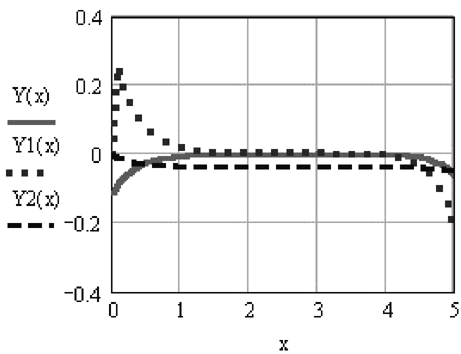


Рис. 2.21. Решение дифференциального уравнения 4-го порядка

Given $\quad 2\,x'(t) + 1\,x(t) - y(t) + z(t) = 0 \qquad x(0) = 0$

$4 \cdot y'(t) + y(t) = x(t) + z(t) \qquad y'(0) = -1$

$\dfrac{d}{dt} z(t) - z(t) = x(t) + y(t) \qquad z'(0) = -1$

Текстовая область между словами given и odesolve допустима

$\begin{pmatrix} x \\ y \\ z \end{pmatrix} := \mathrm{odesolve}\left[\begin{pmatrix} x \\ y \\ z \end{pmatrix}, t, 10\right]$

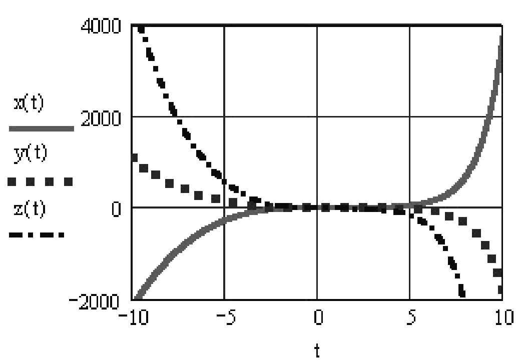


Рис. 2.22. Решение системы дифференциальных уравнений с начальными условиями

Решение системы дифференциальных уравнений приведено на рис. 2.22. При этом обращение к функции Odesolve изменилось. Для решения одного уравнения:

$$\text{odesolve}(x,\ xk,\ n),$$

для решения системы дифференциальных уравнений:

$$\text{odesolve}((\text{вектор имен неизвестных}),\ x,\ xk,\ n).$$

В примере, приведенном на рис. 2.22, при решении системы трех уравнений вектор имен неизвестных содержит три имени.

Потренируйтесь в использовании функции Odesolve (рис. 2.23).

Решить самостоятельно

$$100\cdot\frac{d^2}{dt^2}x(t) + 10\cdot\frac{d}{dt}x(t) + 101\cdot x(t) = 50\cdot\cos\left(\frac{1}{4}\cdot t\right)$$

$x(0) = 0 \qquad x'(0) = 1$

$y' = \dfrac{y}{\ln(y)\cdot\cos(x)} \qquad y(0) = 0$

$y'''\cdot(x-1) - y'' = 0 \qquad y(2) = 2 \qquad y'(2) = 1 \qquad y''(2) = 1$

$y''' - 2\cdot y'' + y' = 0$ поменяйте граничные условия

Решить систему ОДУ $\quad u'(t) - 1\cdot u(t) + 5\cdot v(t) = 0$

$v'(t) - 5\cdot u(t) = 1\cdot v(t)$

при $\quad u(0) = -1 \qquad v(0) = 1$

Рис. 2.23. Примеры для самостоятельной работы

2.11. Анализ экспериментальных данных

При проведении различных экспериментов обычно требуется массив экспериментальных данных представить в виде функции, которую можно использовать в дальнейших расчетах. Если кривая, описываемая этой функцией, должна проходить через все экспериментальные точки, операция получения промежуточных точек и расчетной функции называется *интерполяцией*.

Если кривая, описываемая этой функцией, не должна проходить через все экспериментальные точки и является аппроксимацией (усреднением) исходных данных, операция получения промежуточных точек и расчетной функции называется *регрессией*.

Если необходимо уменьшить разброс данных или исключить некоторую систематическую погрешность, например, в виде наложенных колебаний, используют *сглаживание* данных или фильтрацию спектра колебаний данных.

Подробно о работе с массивами опытных данных рассказано в главе 7. Здесь же поговорим только о наиболее удобных способах интерполяции и регрессии.

2.11.1. Интерполяция

В Mathcad имеется несколько функций интерполяции, различающихся способом «соединения» точек данных (прямой линией или различными кривыми). В этой главе поговорим только о кубической сплайн-интерполяции, при которой экспериментальные точки соединяются отрезками кубических полиномов. В процессе интерполяции одновременно используются две функции, interp и cspline.

Обращение к функциям:

$$\mathrm{interp}(s, x, y, t);\ \mathrm{cspline}(x, y),$$

где x — вектор значений аргумента, элементы которого расположены в порядке возрастания; y — вектор значений функции того же размера; s — вектор вторых производных, создаваемый функцией cspline, которая обеспечивает равенство вторых производных на границах стыковки полиномов, то есть в экспериментальных точках; t — значение аргумента, при котором вычисляется интерполирующая функция.

Координаты экспериментальных точек не могут быть комплексными числами.

Примеры интерполяции приведены на рис. 2.24.

$Y := (38\ \ 15\ \ 5.5\ \ 23\ \ 51\ \ 79\ \ 105\ \ 126\ \ 144\ \ 156\ \ 164\ \ 166\ \ 162\ \ 152\ \ 137\ \ 117\ \ 93\ \ 66)$

$Y := Y^T \quad n := 17 \quad i := 0..n \quad \phi_i := i\cdot\frac{\pi}{9}$ число замеров $n + 1 = 18$

$YY(x) := \mathrm{interp}(\mathrm{cspline}(\phi, Y), \phi, Y, x)$ YY(x) - функция, ее можно интегрировать

$j := 0..nn \quad \phi1_j := \frac{2\cdot\pi}{nn}\cdot j$ nn - новое число точек, задано глобально около графиков. Поменяйте его

$Y1 := \mathrm{interp}(\mathrm{cspline}(\phi, Y), \phi, Y, \phi1)$ Y1 - массив -интегрировать нельзя

Рис. 2.24. Интерполяция исходных данных

При записи функции интерполяции удобно две функции объединять в одну запись:

$$YY(t) := \mathrm{interp}(\mathrm{cspline}(X, Y), X, Y, t).$$

В результате интерполяции можно получить функцию или массив данных с любым количеством точек интерполяции (см. рис. 2.24). Если результатом является функция, ее можно интегрировать, дифференцировать, использовать в функциях пользователя.

На рис. 2.25 число точек nn массива после интерполяции задано глобально. Для удобства наблюдения за изменением результатов расчета $nn \equiv 5$ помещено около графика. Поменяйте nn.

2.11.2. Регрессия

Смысл регрессии состоит в подборе функции, аппроксимирующей экспериментальные данные. Регрессия сводится к подбору коэффициентов в той или иной аналитической зависимости.

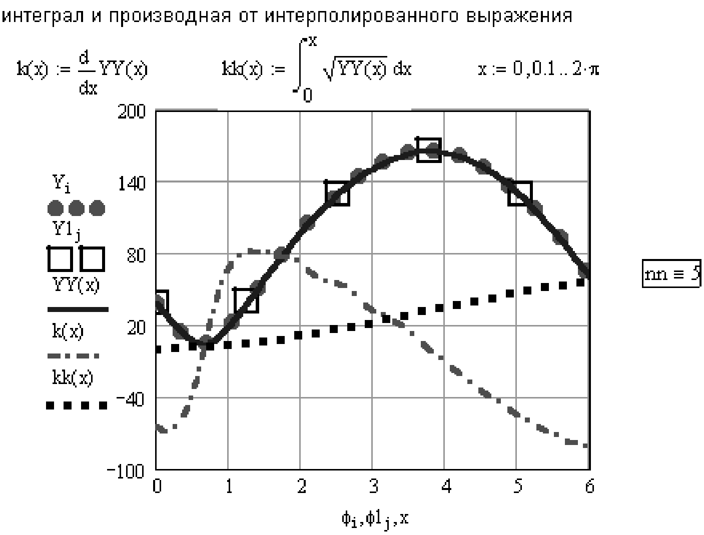


Рис. 2.25. Интерполяция исходных данных (продолжение)

В Mathcad имеется несколько встроенных функций регрессии двух типов:

- позволяющих увидеть аналитическую зависимость, то есть возвращающих набор аппроксимирующих коэффициентов;
- не позволяющих увидеть аналитическую зависимость.

Рассмотрим две функции, которые не выводят коэффициентов и аппроксимируют массив данных одним степенным полиномом или отрезками нескольких полиномов.

В Mathcad регрессия с использованием одного полинома реализуется комбинацией встроенных функций регрессии и интерполяции

$$\text{interp}(s, x, y, t);\ \text{regress}(x, y, n),$$

где x — вектор значений аргумента, элементы которого расположены в порядке возрастания; y — вектор значений функции того же размера; s — вектор коэффициентов для построения аппроксимирующего полинома, создаваемый функцией regress; t — значение аргумента, при котором вычисляется интерполирующая функция; n — степень аппроксимирующего полинома.

Степень аппроксимирующего полинома может быть любой. Практика показывает, что полинома 5-й степени достаточно для аппроксимации почти любой кривой.

Обращение к указанным функциям:

$$s{:=}\text{regress}(X, Y, n),$$

$$YY(t){:=}\text{interp}(s, X, Y, t)$$

или

$$YY(t){:=}\text{interp}(\text{regress}(X, Y, n), X, Y, t).$$

Регрессия с использованием нескольких отрезков полинома реализуется комбинацией встроенных функций регрессии и интерполяции interp(s, x, y, t) и loess (x, y, span), где s:=loess(X, Y, span) — вектор коэффициентов для построения аппроксимирующего полинома 2-й степени, требуемый функцией interp; span > 0 — параметр, определяющий размер отрезков полиномов.

Параметр span задает степень сглаженности данных. На практике 0,2 < span < 2. При span = 2 результат аппроксимации тот же, что и при аппроксимации одной параболой. При span = 0,2 аппроксимирующая кривая почти точно описывает любой набор данных.

Примеры использования полиномиальной регрессии приведены на рис. 2.26.

Регрессия полиномом степени n $n := 3$

$Y_{reg}(x) := \mathrm{interp}(\mathrm{regress}(\phi, Y, n), \phi, Y, x)$

где ϕ и Y-массивы экспериментальных точек,
n -степень полинома (поменяйте степень полинома)

Регрессия отрезками полинома второй степени

Чем больше span,тем больше
длина отрезков (поменяйте span)
$span := 0.4$ $vs1 := \mathrm{loess}(\phi, Y, span)$ $Y_{loess}(x) := \mathrm{interp}(vs1, \phi, Y, x)$

та же запись, но короче $Y_{loess}(x) := \mathrm{interp}(\mathrm{loess}(\phi, Y, span), \phi, Y, x)$

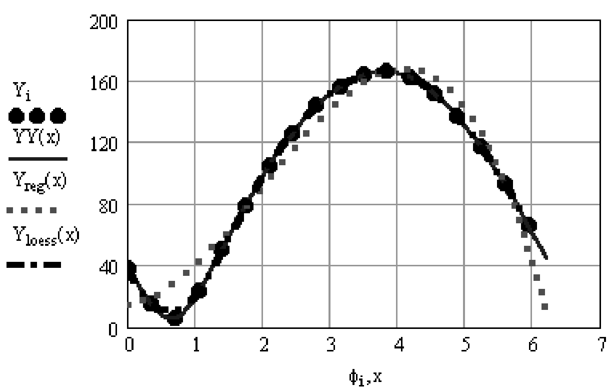


Рис. 2.26. Регрессия исходных данных (см. рис. 2.24) без получения аналитической зависимости

В Mathcad имеется восемь встроенных функций для получения аналитического выражения аппроксимирующей функции. Однако при их использовании необходимо знать форму аналитического выражения. Функции, использующие любой произвольный вид аппроксимирующей функции, будут рассмотрены в главе 7. Здесь же рассмотрим только наиболее простые в применении функции, каждая из которых строит аппроксимирующую функцию лишь определенного вида.

Используйте соответствующий вид регрессии, если хорошо представляете себе, какой зависимостью описывается ваш массив данных. Когда вид регрессии плохо соответствует набору данных, ее результат часто оказывается неудовлетворительным и зависит от выбора начальных приближений.

Из восьми встроенных функций пять требуют предварительного задания вектора начальных приближений:

- expfit(X, Y, g) — регрессия экспонентой $f(t) = a \cdot e^{b \cdot t} + c$;
- sinfit(X, Y, g) — регрессия синусоидой $f(t) = a \cdot \sin(t + b) + c$;
- pwrfit(X, Y, g) — регрессия степенной зависимостью $f(t) = a \cdot t^b + c$;
- lgsfit(X, Y, g) — регрессия логистической функцией $a(e) = a/(1 + b \cdot e^{-c \cdot t})$;
- logfit(X, Y, g) — регрессия логарифмической функцией $f(t) = a \cdot \ln(t + b) + c$.

В этих функциях x — вектор значений аргумента, элементы которого расположены в порядке возрастания; y — вектор значений функции того же размера; g — вектор начальных приближений коэффициентов a, b и c; t — значение аргумента, при котором вычисляется интерполирующая функция.

Примеры использования этих функций приведены на рис. 2.27 и 2.28.

Эти функции требуют начальных приближений

expfit -- $y = a \cdot e^{b \cdot x} + c$ $\qquad c := \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ $\qquad C := \mathrm{expfit}(\phi, Y, c)$

$Y1(x) := C_0 \cdot e^{C_1 \cdot x} + C_2$ $\qquad \mathrm{corr}(Y1(\phi), Y) = 0.60716588$

sinfit -- $y = a \cdot \sin(x + b) + c$ $\qquad C := \mathrm{sinfit}(\phi, Y, c)$

$Y2(x) := C_0 \cdot \sin(x + C_1) + C_2$ $\qquad \mathrm{corr}(Y2(\phi), Y) = 0.98538802$

pwrfit -- $y = a \cdot x^b + c$ $\qquad c2 := \begin{pmatrix} 10 \\ 0.1 \\ 10 \end{pmatrix}$ $\qquad C := \mathrm{pwrfit}(\phi, Y, c2)$

$Y3(x) := C_0 \cdot x^{C_1} + C_2$ $\qquad \mathrm{corr}(Y3(\phi), Y) = 0.68961301$

logfit -- $y = a \cdot \ln(x + b) + c$ $\qquad c1 := \begin{pmatrix} 10 \\ 10 \\ 10 \end{pmatrix}$ $\qquad C := \mathrm{logfit}(\phi, Y, c1)$

$Y4(x) := C_0 \cdot \ln(x + C_1) + C_2$ $\qquad \mathrm{corr}(Y4(\phi), Y) = -0.61951307$

lgsfit --$y = \dfrac{a}{1 + b \cdot e^{-c \cdot x}}$ $\qquad c3 := \begin{pmatrix} 10 \\ 1 \\ 1 \end{pmatrix}$ $\qquad C := \mathrm{lgsfit}(\phi, Y, c3)$

$Y5(x) := \dfrac{C_0}{1 + C_1 \cdot e^{-C_2 \cdot x}}$ $\qquad \mathrm{corr}(Y5(\phi), Y) = 0.88153268$

Рис. 2.27. Регрессия исходных данных (см. рис. 2.24) с выводом аналитической зависимости

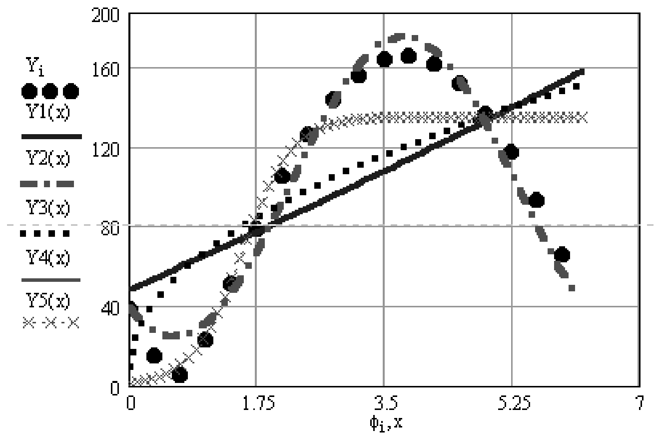


Рис. 2.28. Результаты регрессии исходных данных (см. рис. 2.24) функциями, приведенными на рис. 2.27

Функции, не требующие начальных приближений:

- line(X, Y) — регрессия прямой линией, использующая минимизацию суммы квадратов ошибок $f(t) = a + b \cdot t$;
- medfit(X, Y) — регрессия прямой линией, использующая медиан-медианную линейную регрессию $f(t) = a + b \cdot t$. Функции line и medfit дают близкие результаты, слегка различающиеся наклоном прямых линий;
- lnfit(X, Y) — регрессия логарифмической функцией $f(t) = a \cdot \ln(t) + b$.

Примеры использования этих функций приведены на рис. 2.29 и 2.30.

Эти функции не требуют начальных приближений

medfit a+b*x $C := \text{medfit}(\phi, Y)$ $C = \begin{pmatrix} 34.146 \\ 23.038 \end{pmatrix}$ $Y5(x) := C_0 + C_1 \cdot x$

$\text{corr}(Y5(\phi), Y) = 0.62220544$

line a+b*x $C := \text{line}(\phi, Y)$ $C = \begin{pmatrix} 44.968 \\ 18.538 \end{pmatrix}$ $Y6(x) := C_0 + C_1 \cdot x$

$\text{corr}(Y6(\phi), Y) = 0.62220544$

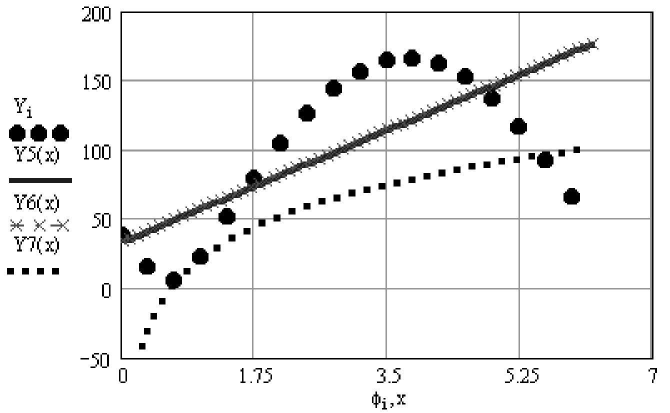
$\phi_0 := 0.01$

lnfit a*ln(x)+b $C := \text{lnfit}(\phi, Y)$ $C = \begin{pmatrix} 22.194 \\ 86.4 \end{pmatrix}$ $Y7(x) := C_0 \cdot \ln(x) + C_1$

$\text{corr}(Y7(\phi), Y) = 0.60503411$

Рис. 2.29. Регрессия исходных данных (см. рис. 2.24) с выводом аналитической зависимости

Во всех примерах, приведенных на рис. 2.27–2.30, использованы одни и те же массивы данных, характер распределения которых, естественно, не соответствует использованным видам регрессии. Для оценки связи между массивом данных и значениями аппроксимирующей функции подсчитан коэффициент корреляции corr. Опытные данные неплохо аппроксимируются синусоидой (коэффициент корреляции 0,98). В остальных случаях связь между величинами плохая (коэффициент корреляции примерно 0,6).



WRITEPRN("Data") := Y   Запись в файл Data.prn

AY := READPRN("Data.prn")   Чтение из файла Data.prn

Рис. 2.30. Результаты регрессии исходных данных (см. рис. 2.24) функциями, приведенными на рис. 2.29

2.12. Элементы математической статистики

В Mathcad встроено много функций для решения задач математической статистики. Подробно они рассматриваются в главе 8. Здесь же остановимся лишь на функциях оценки параметров выборки данных. Все перечисленные далее функции можно применять и для векторов, и для прямоугольных матриц. Примеры вычисления средних арифметических и средних геометрических значений, медианы, средних квадратических отклонений и дисперсий показаны на рис. 2.31.

$$A := \begin{pmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \\ 4 & 5 \\ 5 & 1 \end{pmatrix} \qquad B := \begin{pmatrix} 3 & 4 & 5 \\ 4 & 5 & 1 \\ 5 & 1 & 5 \\ 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix}$$

WRITEPRN("primer") := A^T

APPENDPRN("primer.prn") := B^T

K := READPRN("primer.prn")

$$K = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \\ 3 & 4 & 5 & 1 & 2 \\ 4 & 5 & 1 & 2 & 3 \\ 5 & 1 & 5 & 3 & 4 \end{pmatrix}$$

Рис. 2.31. Оценка параметров выборки данных AY

ВНИМАНИЕ

Функции, начинающиеся с прописной и со строчной буквы, — это разные функции.

2.13. Обмен данными с другими программами

В нижней части рис. 2.30 продемонстрирована еще одна очень важная возможность Mathcad — запись числовых данных в файл и считывание их из файла. Появляется возможность передачи данных из одной программы в другую и даже создания вычислительных комплексов из нескольких программ, исполняемых в различных компьютерных системах. Например, данные из Excel передаются в Fortran, из Fortran — в Mathcad и, если нужно, обратно в Fortran.

В Mathcad существует три функции, позволяющие считать числовые данные из других файлов или записать числовые данные в другой файл:

- APPENDPRN("file") — добавляет числовые данные в существующий файл. Здесь file — имя файла или путь к файлу, если он находится в другом каталоге;
- READPRN("file") — считывает числовые данные из файла;
- WRITEPRN("file") — записывает числовые данные в файл.

Все три функции работают только с файлами, в которых числа образуют прямоугольную матрицу (вектор чисел — частный случай прямоугольной матрицы). Пример обмена данными показан на рис. 2.32. Функция WRITEPRN записывает массив A в файл primer, который сама же и создает с расширением .prn. Функция APPENDPRN добавляет в этот же файл массив B. Функция READPRN считывает данные из файла primer.

$$A := \begin{pmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \\ 4 & 5 \\ 5 & 1 \end{pmatrix} \qquad B := \begin{pmatrix} 3 & 4 & 5 \\ 4 & 5 & 1 \\ 5 & 1 & 5 \\ 1 & 2 & 3 \\ 2 & 3 & 4 \end{pmatrix} \qquad \begin{array}{l} \mathrm{WRITEPRN}(\text{"primer"}) := A^T \\ \mathrm{APPENDPRN}(\text{"primer.prn"}) := B^T \end{array}$$

$$K := \mathrm{READPRN}(\text{"primer.prn"}) \qquad K = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \\ 3 & 4 & 5 & 1 & 2 \\ 4 & 5 & 1 & 2 & 3 \\ 5 & 1 & 5 & 3 & 4 \end{pmatrix}$$

Рис. 2.32. Запись данных в файл и чтение из файла

Для записи прямоугольной матрицы или вектора в отдельный файл необходимо выполнить следующие действия:

- В стандартном меню выбрать кнопку f(x). Появится окно Insert function (Вставить функцию).
- Выбрать группу функций File Access (Доступ к файлам).
- Выбрать имя функции WRITEPRN.
- В появившийся шаблон вписать имя файла, затем оператор присваивания := и имя числового массива. Массив будет записан в файл с указанным вами именем и расширением .prn и помещен в тот же каталог, в котором находится и рабочий файл.

ВНИМАНИЕ

Перед тем как вписать имя файла, не забудьте ввести символ (") (кавычки).

Для передачи в Mathcad данных из какого-либо файла, содержащего прямоугольную матрицу числовых данных, выполните следующие действия:

- Наберите имя, которое присваиваете массиву, считываемому из файла, и знак присвоения значения :=.
- В стандартном меню выберите кнопку f(x). Появится окно Insert function.
- Выберите группу функций File Access.
- Выберите имя функции READPRN.
- В появившийся шаблон впишите имя файла с расширением, если оно есть, не забывая перед этим ввести символ ".

Подробнее об обмене данными говорится в разделе 12.3.

ПРИМЕЧАНИЕ

В Mathcad 13 появилась функция Readfile, позволяющая считывать данные из любых файлов. Подробно о ней написано в разделе 12.3.

2.14. Учет размерностей в Mathcad

Mathcad позволяет вести расчеты с учетом размерностей. Для этого при вводе исходных данных достаточно умножить число на стандартную размерность. Теперь любые действия с введенными таким образом величинами выполняются с учетом размерностей.

Перед началом работы с размерными величинами надо установить систему единиц, в которой вы будете работать:

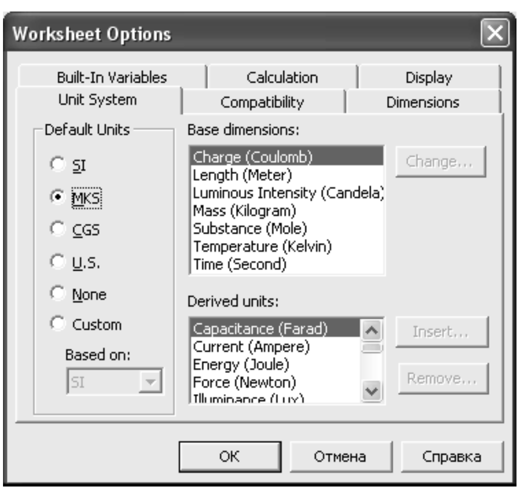


Рис. 2.33. Установка системы единиц

- В главном меню Mathcad выберите Tools ▸ Worksheet Options ▸ Unit System, как показано на рис. 2.33.
- В открывшемся диалоговом окне выберите при помощи переключателя систему единиц, например SI (International). Щелкните на кнопке OK. Если в диа-

логовом окне выбрать None (Нет), то никакие размерности в расчете не учитываются.

В дальнейшем в начале документа при вводе исходных данных для расчета их надо записывать, умножая число на стандартное обозначение размерности (можно ввести его с клавиатуры либо выбрать в стандартном меню Mathcad кнопку Insert Unit с изображением мерной кружки (рис. 2.34), щелкнув мышью на соответствующем обозначении).

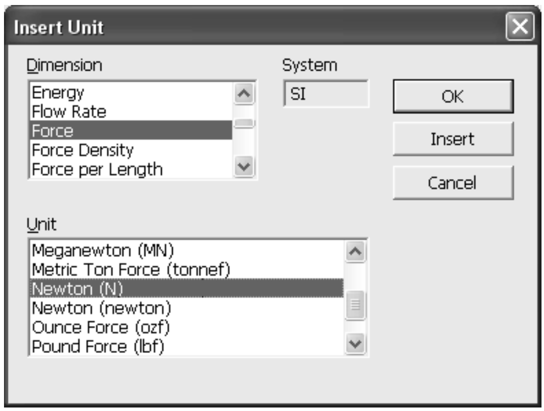


Рис. 2.34. Окно выбора стандартных размерностей

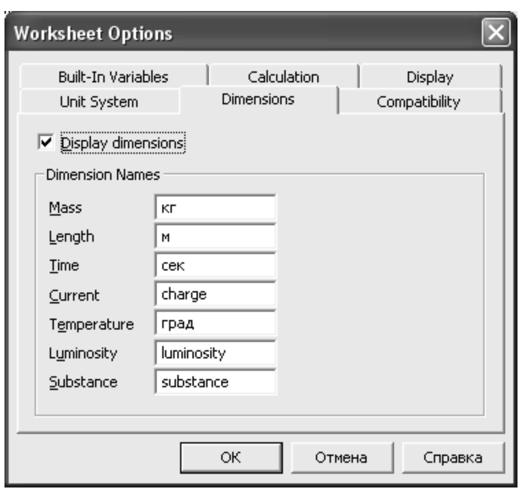


Рис. 2.35. Окно установки своих размерностей

В принципе можно придумать и использовать свои, абсолютно любые размерности. Тогда в начале Mathcad-документа надо ввести свои производные размерности, например, размерности на русском языке, присвоив их значениям размерностей, указанных в стандартном меню Mathcad (рис. 2.35).

При выводе результатов расчета после нажатия клавиши = появляется число и рядом с ним — место ввода единицы размерности. Если число уже имеет размерность, то рядом с размерностью появляется еще одно место ввода. Сюда можно поместить любую переменную, константу или выражение.

ВНИМАНИЕ

Не забывайте, что всякий раз, когда что-нибудь находится в месте ввода единицы измерения, Mathcad изменит полученный результат так, чтобы произведение числа и выражения, стоящего на месте размерности, давало правильное значение.

Можно использовать место ввода, чтобы, например, выразить, значение угла в долях числа π или вывести числовой коэффициент при каком-то буквенном выражении.

Размерность элементов массива вводится так же, как и размерность чисел, путем умножения выражения на стандартную размерность на английском языке или производную размерность, введенную ранее.

Дискретная переменная может быть размерной. Для этого в ней необходимо умножить числа на соответствующую размерность.

Графики всегда строятся в стандартных размерностях, приведенных в List of built-in-units (Список встроенных функций). Ввести пользовательские размерности на графиках нельзя. Для изменения масштаба графика разделите на нем названия аргумента и функции на размерность (рис. 2.36).

Ограничения на проведение расчетов с учетом размерностей:

- При расчетах с матрицами все элементы используемых в расчетах матриц должны иметь одинаковую размерность.
- Ряд встроенных функций не может работать с размерными величинами. Это функции регрессии, сглаживания, функция Odesolve (решение дифференциальных уравнений), вычисления логарифма числа (функции ln и log). Для того чтобы преодолеть это ограничение, нужно сделать безразмерными аргументы функций, разделив размерную величину на ее размерность.
- В сложных расчетах при использовании многих функций и нескольких дискретных переменных учет размерностей может оказаться сложным. В таких случаях размерность (для справки) можно указать рядом с результатом расчета в виде текстового комментария.

Подробные сведения об учете размерностей в расчетах можно найти в главе 11.

ПРИМЕЧАНИЕ

В Mathcad 13 введена статическая проверка размерностей, накладывающая гораздо больше ограничений на учет размерностей. Из-за этого ряд программ, работающих в предыдущих версиях, не действует в этой версии. Подробнее о статической проверке размерностей написано в разделе 11.10.

Квазиразмерности

По сути дела процесс присвоения числу размерности состоит в умножении числа на некоторый числовой коэффициент. Буквенному обозначению размерности достаточно присвоить численное значение, не умножая его на стандартную размерность.

Ввод производных размерностей

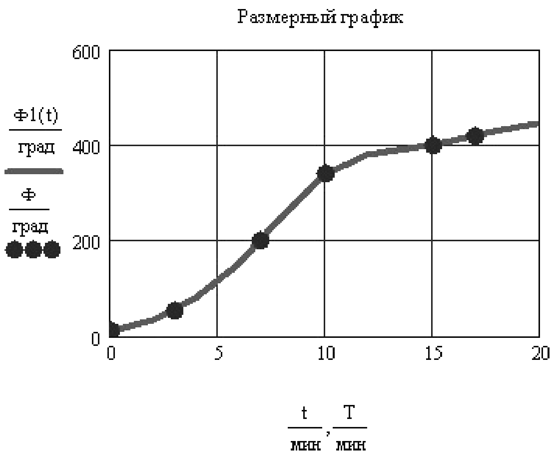
$метр := m \quad сек := s \quad мин := 60 \cdot сек \quad час := 3600 \cdot s \quad град := deg$

Размерные массивы $\quad Ф := (10\ \ 55\ \ 200\ \ 340\ \ 400\ \ 420)^T \cdot град$ угол поворота

$T := (0\ \ 3\ \ 7\ \ 10\ \ 15\ \ 17)^T \cdot мин$ время

Размерная функция $\quad Ф1(t) := interp(lspline(T, Ф), T, Ф, t) \qquad Ф2(t) := \frac{Ф1(t)}{t}$

Размерная дискретная переменная $\quad t := 1 \cdot сек, 2 \cdot мин .. 20 \cdot мин \qquad Ф2(t) =$



$Ф2(t) =$ $\frac{град}{мин}$
611.254
17.919
20.251
25.592
31.18
33.997
32
28.372
25.664
24.054
22.363

Рис. 2.36. Различные действия с размерными величинами

ПРИМЕЧАНИЕ

Стандартный ввод размерности — производной размерности присваивается число, умноженное на стандартную размерность.

Ввод квазиразмерности — производной размерности присваивается число.

Использование квазиразмерностей позволяет избежать всех ограничений, также позволяет использовать размерности в программах, в матричных расчетах. Пример использования квазиразмерностей показан на рис. 11.19.

При использовании квазиразмерностей Mathcad уже не следит за правильностью использования размерностей и вся ответственность за результаты расчета ложится на расчетчика, который сам должен следить за логикой расчета и вписывать в поле ввода ответа правильную размерность.

2.15. Преобразование функции в матрицу и матрицы в функцию

Преобразование функции в матрицу и матрицы в функцию необходимо в связи с особенностями встроенных функций Mathcad. Это делается, например, если одни функции возвращают матрицу, а вам надо иметь функцию для последующего интегрирования или дифференцирования, или, наоборот, Mathcad выводит функцию, а вам нужна матрица для последующих матричных преобразований.

Для преобразования массива данных в функцию используют интерполяцию или регрессию. На рис. 2.37 показано преобразование вектора в функцию. Там же показана возможность дифференцирования полученной функции.

Заданы векторы аргумента X и функции Y

$X := (-3\ \ 1\ \ 5\ \ 6\ \ 9\ \ 10)^T \qquad Y := (2\ \ 5\ \ -3\ \ 4\ \ 9\ \ 2)^T$

$F_{interp}(x) := \mathrm{interp}(\mathrm{cspline}(X,Y), X, Y, x)$ интерполяция

$F_{regress}(x) := \mathrm{interp}(\mathrm{regress}(X,Y,4), X, Y, x)$ регрессия полиномом 4-й степени

$x := X_1 .. X_{rows(X)}$

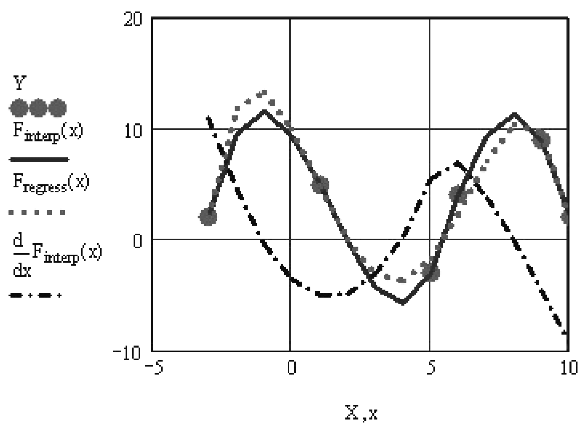


Рис. 2.37. Преобразование вектора данных в функцию

Для получения массива данных с помощью заданной функции надо задать векторы значений аргументов функции и, подставив в функцию значения аргументов, получить массив (рис. 2.38).

ORIGIN := 1

$x := (3\ \ 5\ \ 9)^T \qquad y := (2\ \ -20\ \ 7\ \ 15)^T$

$i := 1 .. \mathrm{rows}(x) \qquad j := 1 .. \mathrm{rows}(y)$

$$M_{i,j} := \frac{\ln\left[\left(x_i\right)^2 + \left(y_j\right)^2\right]}{x_i + y_j} \qquad M = \begin{pmatrix} 0.513 & -0.354 & 0.406 & 0.303 \\ 0.481 & -0.403 & 0.359 & 0.276 \\ 0.404 & -0.561 & 0.304 & 0.238 \end{pmatrix}$$

Рис. 2.38. Получение массива значений с помощью заданной функции

2.16. Строковые функции

Строковые функции в Mathcad позволяют выполнять действия с фрагментами текста, формировать из них необходимую реплику (рис. 2.39). На том же рисунке показана функция CWD, позволяющая узнать путь к рабочему файлу. Эта функция полезна, когда на диске находится несколько копий файла и важно не перепутать, с какой из них идет работа.

```
путь к открытому файлу      CWD = "G:\Быстрый старт\"

n := 6   m := 4      C := "должно быть равно  "

A := "Число строк "   B := "Число столбцов "

D1 := num2str(n)      D2 := num2str(m)

Z := | concat(A,C,D1)  if  n > m
     | concat(A,C,D2)  if  m > n
     | 1  otherwise

Z = "Число строк должно быть равно  6"
```

Рис. 2.39. Формирование реплики с помощью строковых функций

Подробнее о строковых функциях читайте в разделе 12.12.

2.17. Программирование

Раздел «Программирование» занимает особое место в Mathcad. При начальном обучении этот раздел совершенно не нужен. Огромные возможности Mathcad позволяют решать подавляющее число задач без использования программирования, да к тому же, как правило, несколькими способами.

Но есть класс задач, которые невозможно решить без программирования. Это задачи, в которых часть документа из нескольких или многих операторов надо выполнить многократно. В таких случаях документ должен состоять из отдельных подпрограмм, объединенных в единую «головную» программу.

Использование раздела «Программирование» позволяет написать в Mathcad программы любой сложности. Подробные сведения о создании программ содержатся в главе 9.

2.18. Анимация

Mathcad предусматривает возможность анимации графиков и результатов вычислений путем создания AVI-файлов и вставки их в Mathcad-документ. Воспроизводятся AVI-файлы встроенными в Windows средствами. Анимация не упрощает и не улучшает расчеты, а лишь украшает их презентацию. Поэтому на первых шагах обучения Mathcad обращаться к ней не следует. Подробные сведения о создании анимации к расчетам содержатся в главе 10.

2.19. Отладка Mathcad-документов

Mathcad-документ — это набор исходных данных для расчета, расчетных формул, выведенных на экран результатов расчета в виде чисел, таблиц и графиков. Каждый из перечисленных объектов занимает одну математическую область (Math Region). Между ними в произвольном порядке располагаются текстовые области (Text Region). Все математические объекты взаимодействуют друг с другом. Напомним, что Mathcad вычисляет выражения сверху вниз и слева направо последовательно друг за другом и не переходит к вычислению следующего объекта, не закончив работу с предыдущим. Текстовые области никак не влияют на математические области.

Если в каком-либо выражении есть ошибка, величина, содержащая ошибку, отображается красным цветом. Щелкните мышью на объекте с ошибкой. Под объектом появляется сообщение об ошибке на английском языке. При нажатии клавиши F1 на экране появляется часть раздела справки, посвященная расшифровке этой ошибки и возможным действиям по ее устранению. Чтобы просмотреть все сообщения об ошибках, выберите в стандартном меню Mathcad пункт Help ▸ Mathcad Help ▸ Search (Помощь ▸ Справка Mathcad ▸ Поиск). В окно поиска впишите слово «errors» (ошибки). Нажмите кнопку Go. В появившемся списке статей выберите Errors Messages in Numerical Calculations (Ошибки в численных вычислениях). На экране появится перечень возможных сообщений (рис. 2.40).

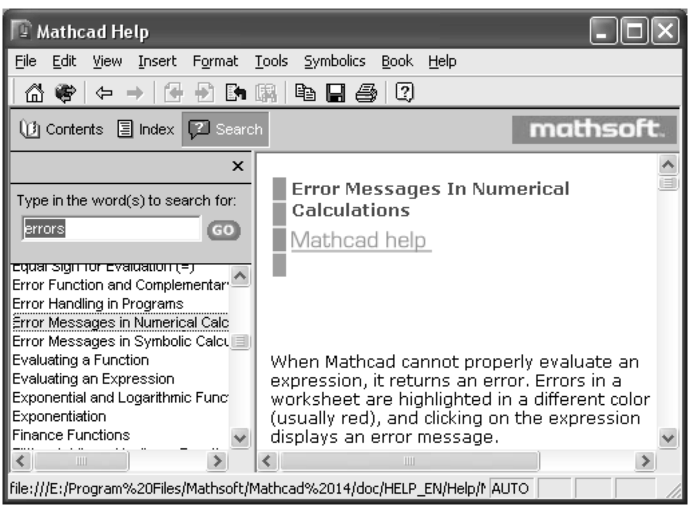


Рис. 2.40. Описание возможных ошибок в меню Help Mathcad 14

Щелчок мышью на любом из сообщений открывает расшифровку ошибки аналогично нажатию клавиши F1.

В Mathcad есть функция Trace Errors (Трассировка ошибок), позволяющая проследить всю цепочку ошибочных вычислений, то есть найти место, где была допущена ошибка. Щелкните правой кнопкой мыши на выражении с ошибкой. Если ошибка возникла не в этом выражении, то в контекстном меню появляется пункт Trace Errors. Выберите этот пункт и, следуя указаниям, найдите выражение, из которого исходит ошибка. На практике эта функция не слишком полезна, так как в простых случаях и без нее понятно, откуда взялась ошибка, а в сложных программах Mathcad сам этого не понимает. В таких случаях в диалоговом окне все кнопки, кроме Close (Закрыть), заблокированы (затемнены).

Многие ошибки устраняются легко. Прочитайте сообщение об ошибке, и станет ясно, что делать. Но встречаются ошибки, на устранение которых уходит несколько дней глубоких раздумий. В таких случаях надо просмотреть численные значения всех входящих в ошибочное выражение величин. Возможно, заданные вами значения где-то были заменены другими.

СОВЕТ

Если правильно записанное выражение выводит неправильный числовой ответ, попробуйте «обнулить» имя выражения и значения ряда параметров перед их последним вычислением, так как возможно наложение значений друг на друга.

На рис. 2.41 показано, как ранее принятое значение матрицы *A* не дает возможности переприсвоить ей значение вектора.

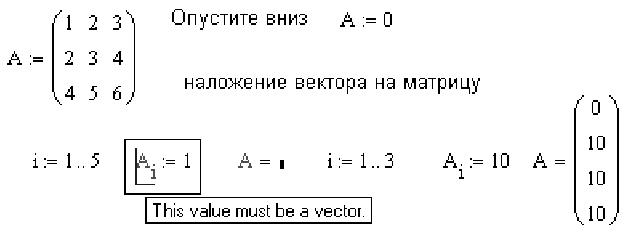


Рис. 2.41. Наложение вектора на матрицу

Иногда встречается наложение векторов друг на друга (рис. 2.42). Например, вектору *A* из шести значений присваивается вектор *A* из трех значений. В некоторых случаях эти вектора накладываются друг на друга, а в некоторых — нет. На рис. 2.42 исправлена ошибка, присутствовавшая на рис. 2.41: массив *A* обнулен. Однако появилась новая ошибка: два вектора *A* наложились друг на друга. Необходимо обнулить вектор *A*.

Рис. 2.42. Наложение вектора на вектор

Другие возможности отладки программ вы узнаете, прочитав части II и III этой книги. Встретившись с непонятной ошибкой, откройте книгу там, где рассказывается о функции, вызвавшей ошибку. Откройте в электронной книге соответствующий фрагмент. Подумайте, почему там функция работает, а у вас — нет. Скопируйте в свой документ фрагмент из электронной книги и попробуйте приспособить его к условиям своей задачи.