Учреждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ
УНИВЕРСИТЕТ»

А. П. ЛАЩЕНКО

ИНЖЕНЕРНО-ЭКОНОМИЧЕСКИЕ ЗАДАЧИ НА БАЗЕ MATHCAD

**Практикум
для студентов экономических специальностей**

Минск 2006

А. П. ЛАЩЕНКО

ИНЖЕНЕРНО-ЭКОНОМИЧЕСКИЕ ЗАДАЧИ НА БАЗЕ MATHCAD

Практикум
для студентов экономических специальностей

Минск БГТУ 2006

УДК 681.3.06 (075.6)
ББК 32.973.26-018.2я7
Л 32

Рассмотрен и рекомендован к изданию редакционно-издательским советом университета

Рецензенты:
доктор физико-математических наук, профессор Белорусского государственного университета *А. И. Калинин*;
кандидат технических наук, доцент Белорусского государственного экономического университета *М. Н. Садовская*

Лащенко, А. П.
Л 32 Инженерно-экономические задачи на базе Mathcad : практикум для студентов экономических специальностей / А. П. Лащенко. – Мн.: БГТУ, 2006. – 68 с.

ISBN 985-434-605-6

Рассматриваются основы работы в системе Mathcad. Представлены задачи инженерно-экономического профиля. Дается краткое изложение математического пакета Mathcad и практические рекомендации (листинги программ) по его использованию при решении инженерно-экономических задач.

УДК 681.3.06 (075.6)
ББК 32.973.26-018.2я7

ISBN 985-434-605-6

ВВЕДЕНИЕ

Изучение основ вычислительной математики с использованием компьютерных технологий является одним из важнейших этапов при подготовке современного инженера. Это обусловлено стремительным внедрением новейших информационных технологий во все сферы человеческой деятельности. Квалифицированный инженер должен владеть методами алгоритмизации задач в области своей профессиональной деятельности и уметь реализовывать эти алгоритмы и задачи отрасли в различных интегрированных средах.

Использование средств, предназначенных для решения задач инженерно-экономического характера, в настоящее время переживает четвертый этап революционных перемен, связанных с появлением мощных компьютерных пакетов: Mathcad, Mathematica, Mathlab, Derive, Theorist и т. д. (первые три этапа этой революции в свое время знаменовались соответственно появлением счетной доски, бухгалтерских счетов и микрокалькулятора). Поэтому цель данной работы – синтезировать традиционные методы решения задач инженерно-экономического характера на примере системы Mathcad. Следует хорошо представлять себе, что в состав Mathcad входит несколько интегрированных между собой компонентов:

- мощный текстовый редактор, позволяющий вводить, редактировать и форматировать как текст, так и математические выражения;
- вычислительный процессор, умеющий проводить расчеты по введенным формулам с использованием встроенных численных методов;
- символьный процессор, являющийся, фактически, системой искусственного интеллекта;

Объединение текстового, формульного и графического редакторов с вычислительным ядром позволяет готовить активные электронные документы с высоким качеством оформления, способные выполнять расчеты с наглядной демонстрацией результатов. Итоговые документы могут трансформироваться в файлы форматов rtf и html и использоваться в пакете MS Office и в сетях Internet, Intranet.

Дополнительную информацию читатель может получить в Интернете на сервере производителя Mathcad http://www.mathcad.com и дистрибьютора Mathcad в России http://www.mathcad.ru.

1. MATHCAD

1.1. Общие сведения о программе Mathcad

Для решения сложных расчетных задач используют специально написанные программы. В то же время в научной работе встречается широкий спектр задач ограниченной сложности, для решения которых можно использовать универсальные средства.

К универсальным программам, пригодным для решения таких задач, относится, например, программа Mathcad, которая представляет собой автоматизированную систему, позволяющую динамически обрабатывать данные в числовом и аналитическом (формульном) виде. Программа Mathcad сочетает в себе возможности проведения расчетов и подготовки форматированных научных и технических документов.

Научно-технические документы обычно содержат формулы, результаты расчетов в виде таблиц данных или графиков, текстовые комментарии или описания, другие иллюстрации. В программе Mathcad им соответствуют два вида объектов: формулы и текстовые блоки. Формулы вычисляются с использованием числовых констант, переменных, функций (стандартных и определенных пользователем), а также общепринятых обозначений математических операций.

Включенные в документ Mathcad формулы автоматически приводятся к стандартной научно-технической форме записи. Графики, которые автоматически строятся на основе результатов расчетов, также рассматриваются как формулы. Комментарии, описания и иллюстрации размещаются в текстовых блоках, которые игнорируются при проведении расчетов.

Чтобы буквенные обозначения можно было использовать при расчетах по формулам, им должны быть присвоены числовые значения. В программе Mathcad буквенные обозначения рассматриваются как переменные и их значения задаются при помощи оператора присваивания. Таким же образом можно задать числовые последовательности, аналитически определенные функции, матрицы и векторы.

Если все значения переменных известны, то для нахождения числового значения выражения (скалярного, векторного или матричного) надо подставить все числовые значения и произвести все заданные действия.

В программе Mathcad для этого применяют оператор вычисления. В ходе вычисления автоматически используются значения переменных и определения функций, заданные в документе ранее. Удобно задать значения известных параметров, провести вычисления с использованием аналитических формул, результат присвоить некоторой переменной, а затем использовать оператор вычисления для вывода значения этой переменной. Изменение значения любой переменной, коррекция любой формулы означает, что все расчеты, зависящие от этой величины, нужно проделать заново. Такая необходимость возникает при выборе подходящих значений параметров или условий, поиске оптимального варианта, исследовании зависимости результата от начальных условий. Электронный документ, разработанный в программе Mathcad, готов к подобной ситуации. При изменении какой-либо формулы Mathcad автоматически производит необходимые вычисления, обновляя изменившиеся значения.

1.2. Входной язык системы Mathcad

При решении задач система Mathcad требует от пользователя описания алгоритма решения задачи на входном языке.

Алфавит входного языка пакета Mathcad – совокупность специальных знаков и слов, которые используются при задании команд, необходимых для решения задачи.

Алфавит содержит:

- прописные и строчные буквы (латинские и греческие);
- цифры от 0 до 9;
- системные переменные;
- операторы;
- имена встроенных функций;
- специальные знаки.
- типы данных: константы, переменные, массивы, данные файлового типа;
- операторы: элементы языка, с помощью которых создаются математические выражения;
- функции: встроенные и определяемые пользователем.

1.2.1. Операторы. Набор основных арифметических операторов системы (сложение «+», вычитание «–», умножение «*», деление «/», возведение в степень «^») возможен с клавиатуры или с использованием кнопок панели инструментов **Арифметика**, которые появляются при выборе из меню **Вид ➢ Панели инструментов ➢ Арифметика** или при щелчке на пиктограмму панели инструментов **Арифметика** на панели инструментов **Математика**.

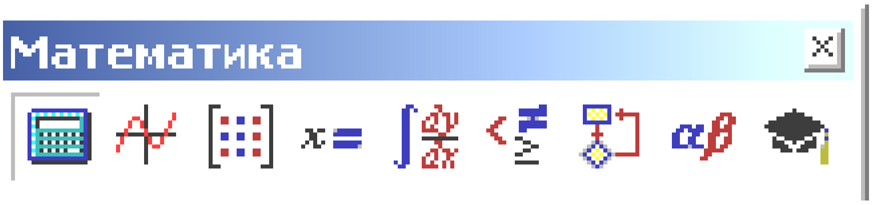
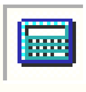
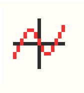


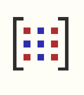
Рис. 1. Панель инструментов Математика

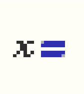
Элементы формул можно вводить с клавиатуры или с помощью панели инструментов **Математика**, которая вызывается командой меню **Вид ➢ Панели инструментов ➢ Математика**.

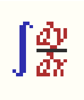
Назначение элементов этой панели следующее:

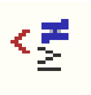
 – панель инструментов **Арифметика** (арифметические инструменты) – ввод чисел, знаков математических операций, наиболее часто используемых стандартных функций;

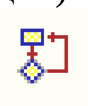
 – панель инструментов **График** (инструменты графиков) – построение графиков;

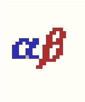
 – панель инструментов **Матрицы** (векторные и матричные инструменты) – ввод векторов и матриц и задание матричных операций;

 – панель инструментов **Вычисления** (инструменты некоторых знаков) – ввод операторов вычисления и знаков логических операций;

 – панель инструментов **Исчисление** (операторы математического анализа) – задание операций, относящихся к математическому анализу;

 – панель инструментов **Булево** (символы логических операций) – ввод знаков логических выражений;

 – панель инструментов **Программирования** (операторы и символы программирования) – ввод операторов программирования;

 – панель инструментов **Греческий алфавит** (символы греческого алфавита) – ввод греческих букв;

 – панель инструментов **Символы** (символические операторы) – осуществление символьных вычислений.

Напомним некоторые операторы:

- оператор присваивания – := (двоеточие и знак равенства);
- оператор вывода – = (знак равенства). Например, после ввода выражения $4 + 19^2 – \log(3)$ надо ввести с клавиатуры символ =, в итоге отображается результат вычисления;
- оператор сравнения (равно) **=** (символический знак равенства). Например, при записи sin(x) **=** 0 необходимо использовать **=** – символический знак равенства, ввод которого осуществляется с помощью пиктограммы **Булево равенство =**, находящейся на панели инструментов **Вычисления**, или нажатием клавиш **Ctrl + =**.

1.2.2. Числовые и размерные константы. Константы – именованные объекты, хранящие некоторые значения, которые не могут быть изменены.

Числовые константы задаются с помощью арабских цифр, десятичной точки, знака «минус».

Например:

567 – целочисленная десятичная константа;

45.6 – десятичная константа с дробной частью;

$3.5*10^{–5}$ – десятичная константа с мантиссой 3.5 и порядком – 5.

Размерные константы задаются в виде

<числовая константа> <знак умножения> <единица измерения>.

Например, 5 секунд в Mathcad выглядит как 5 · sec.

Последовательность действий при вводе размерных констант:

- введите константу или переменную;
- введите знак умножения * (звездочка);
- выберите команду меню **Вставка ➤ Единицы измерения**;
- выберите в окне **Измерения** нужный тип;
- выберите в окне **Единицы** нужную величину;
- щелкните на кнопке **ОК**.

1.2.3. Переменные. Переменные в системе Mathcad – именованные объекты, имеющие некоторое значение, способное изменяться по ходу выполнения программы.

Имена переменных могут иметь произвольную длину, но начинаться должны с буквы. Они могут состоять из букв (латинских и греческих), цифр от 0 до 9, символа бесконечности, символа подчеркивания, апострофа, символа процента (%), нижних индексов.

Для того чтобы можно было вычислить выражение, зависящее от каких-либо переменных, значения этих переменных должны быть определены.

Для присвоения значения переменной необходимо:

- ввести имя переменной;
- ввести двоеточие (:), что приведет к появлению знака присваивания := и следующего за ним поля ввода, или щелкнуть на кнопке **Присвоить значение** на панели инструментов **Арифметика**;
- набрать в поле ввода число или выражение.

Mathcad вычислит соответствующее значение и присвоит его переменной.

Все переменные и функции, присутствующие во введенном выражении, должны быть определены заранее. В противном случае переменные, значения которых не определены к моменту вычисления выражения, будут отмечены на экране дисплея негативным изображением.

Если переменной присваивается начальное значение с помощью оператора :=, то присваивание называется локальным (присваивание должно производиться перед использованием переменной в выражениях).

Если переменной присваивается начальное значение с помощью оператора знака ≡ из панели инструментов **Вычисление**, то присваивание называется глобальным (присваивание может проводиться в любом месте документа).

Размерные переменные характеризуются не только значением, но и указанием физической величины, значение которой они хранят.

Системные переменные имеют значения, которые определены системой Mathcad.

К системным переменным относятся:

- число «пи» – $\pi = 3.14$;

- основание натурального логарифма – e = 2.71;
- системная бесконечность – $\infty = 10^{37}$;
- процент – % = 0.01;
- *tol* – погрешность численных методов = 0.001;
- *origin* – нижняя граница индекса массивов = 0.

1.2.4. Массивы. Массивами в системе Mathcad (*arrays*) называют упорядоченные последовательности чисел, или элементов, массива. Доступ к любому элементу массива возможен по его индексу, т. е. номеру в последовательности чисел (в *листинге 1* *a* – это массив, a_1 – его элемент). Применение массивов чрезвычайно эффективно в математических расчетах.

Листинг 1. Одномерный массив (вектор)

$$a := \begin{pmatrix} 14 \\ 1.4 \\ 4.7 \end{pmatrix}$$

$$a_0 = 14 \quad a_1 = 1.4 \quad a_2 = 4.7$$

В Mathcad условно выделяются два типа массивов:

- векторы (одноиндексные массивы, *листинг 1*), матрицы (двухиндексные, *листинг 2*) и тензоры (многоиндексные);
- ранжированные переменные (*range variables*) – векторы, элементы которых определенным образом зависят от их индекса.

Листинг 2. Двумерный массив (матрица)

$$a := \begin{pmatrix} 0.1 & 2.8 \\ 3.7 & 0 \end{pmatrix}$$

$$a_{0,0} = 0.1 \quad a_{0,1} = 2.8 \quad a_{1,0} = 3.7 \quad a_{1,1} = 0$$

Доступ ко всему массиву осуществляется по имени векторной переменной. Например, последовательность символов «*a*», «=» в *листингах 1* и *2* приведет к выводу соответствующего вектора или матрицы. В Mathcad имеются операторы и встроенные функции, которые действуют на векторы и матрицы целиком, например транспонирование, матричное умножение и т. д.

Над элементами массива можно совершать действия как над обычными числами. Нужно только правильно задать соответствующий индекс или сочетание индексов массива. Например, чтобы получить доступ к нулевому элементу вектора *a* из *листинга 1*:

- введите имя переменной массива (*a*);
- нажмите кнопку **Subscript** (**Нижний индекс**) со значком x_n, на панели **Matrix** (**Матрица**), либо введите [;
- в появившийся справа снизу от имени массива местозаполнитель введите желаемый индекс (0).

Если после этого ввести знак численного вывода, то справа от него появится значение нулевого элемента вектора, как показано во второй строке *листинга 1*.

Чтобы получить доступ к элементу многоиндексного массива (например, элементу $a_{1,0}$ матрицы *a* из *листинга 2*):

- введите имя переменной массива (*a*);
- перейдите к вводу нижнего индекса, введя [;
- введите в местозаполнитель индекса первый индекс (1), запятую «,» и в появившийся после запятой местозаполнитель введите второй индекс (0).

В результате будет получен доступ к элементу, как показано в последней строке *листинга 2*.

В рассмотренных листингах нумерация индексов массивов начинается с нуля, иными словами, первый элемент массива имеет индекс 0. Стартовый индекс массива задается системной переменной, *origin,* которая по умолчанию равна нулю. Если необходимо нумеровать элементы векторов и матриц с единицы, присвойте этой переменной значение 1 (*листинг 3*).

Листинг 3. Изменение нумерации индексов массивов

$$\mathrm{origin} := 1$$

$$a := \begin{pmatrix} 14 \\ 1.4 \\ 4.7 \end{pmatrix}$$

$$a_1 = 14 \quad a_2 = 1.4 \quad a_3 = 4.7$$

Помимо доступа к отдельным элементам массива, имеется возможность совершать действия над его подмассивами (например, векторами-столбцами, образующими матрицу). Делается это с помощью оператора со значком $x^{<>}$ на панели **Matrix**.

Ранжированные переменные – это особый класс переменных, который в Mathcad заменяет управляющие структуры, называемые циклами. Эти переменные имеют ряд фиксированных значений, меняющихся от начального до конечного с определенным шагом. Ранжированные переменные имеют имя и индекс (порядковый номер) каждого элемента.

Для создания ранжированной переменной целочисленного типа используется выражение

$$Name := N_{begin} \,..\, N_{end}.$$

Например, если в вычислениях используется ряд натуральных чисел {1, 2, 3, 4, 5}, в Mathcad это выглядит как M := 1..5 (имя переменной может быть любым, русские символы не допускаются).

Для создания ранжированной переменной общего вида используется выражение

$$Name := N_{begin}, (N_{begin} + Step) \,..\, N_{end},$$

где *Name* – имя переменной; N_{begin} – начальное значение переменной; N_{end} – конечное значение переменной; *Step* – шаг изменения переменной.

Например:

- если в вычислениях используется ряд чисел {1, 3, 5, 7, 9}, в Mathcad это выглядит как M := 1, (1 + 2) .. 9 (имя переменной может быть любым) или M:= l, 3 .. 9;
- если в вычислениях используется ряд чисел {9, 7, 5, 3, 1}, в Mathcad это выглядит как M := 9, 7 .. 1 (имя переменной может быть любым).

Вывод значений ранжированной переменной:

$$Name = .$$

Существует несколько способов создания массива. Самый простой и наглядный способ создания вектора или матрицы заключается в следующем:

- нажмите кнопку **Matrix or Vector** (**Матрица или вектор**) на панели **Matrix** (**Матрица**) (рис. 2.) либо клавиши **<Ctrl> + <M>**, либо выберите пункт меню **Insert / Matrix** (**Вставка / Матрица**);
- в диалоговом окне **Insert Matrix** (**Вставка матрицы**) задайте целое число столбцов и строк матрицы, которую хотите создать. Например, для создания вектора 3 × 1 введите показанные на рис. 2 значения;

- нажмите кнопку **OK** или **Insert** (**Вставить**) – в результате в документ будет вставлена заготовка матрицы с определенным числом строк и столбцов;
- введите значения в местозаполнители элементов матрицы. Переходить от одного элемента матрицы к другому можно с помощью указателя мыши, клавиш со стрелками либо нажимая клавишу **Tab**.

Добавление в уже созданную матрицу строк или столбцов производится точно так же.

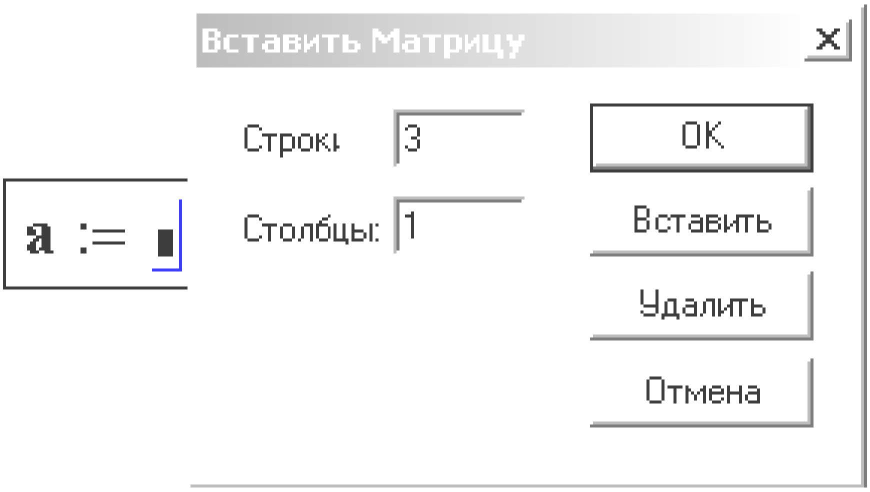


Рис. 2. Вставка Матрицы

В местозаполнители элементов матрицы можно вставлять не только числа (действительные или комплексные), но и любые математические выражения, состоящие из переменных, операторов, встроенных и пользовательских функций.

Другой способ создания массива заключается в определении любого количества его элементов. Это можно сделать, присваивая значения непосредственно отдельным элементам массива, применяя ранжированные переменные (*листинг 4*.)

Листинг 4. Создание матрицы определением ее элементов

$$A_{0,0} := 3 \quad A_{3,2} := 9$$

$$A = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 9 \end{pmatrix}$$

Любой из этих способов позволяет присвоить нужное значение как всем элементам массива, так и части из них либо даже одному-единственному элементу. В последнем случае создается массив, размерность которого задается индексами введенного элемента (*листинг 4*), а неопределенным элементам по умолчанию присваиваются нулевые значения.

В Mathcad легко создать матрицы определенного вида с помощью одной из встроенных функций:

- *identity*(N) – единичная матрица размера $N \times N$;
- *diag*(v) – диагональная матрица, на диагонали которой находятся элементы вектора v;
- *geninv*(A) – создание матрицы, обратной (слева) матрице A;
- *rref*(A) – преобразование матрицы или вектора A в ступенчатый вид,

где N – целое число; v – вектор; A – матрица из действительных чисел.

Примеры использования этих функций приведены в *листинге 5*:

Листинг 5. Создание матриц специального вида

$$\mathrm{identity}(3) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathrm{diag} = \left(\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}\right) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix}$$

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

$$\mathrm{geninv}(A) = \begin{pmatrix} -1.333 & -0.333 & 0.667 \\ 1.083 & 0.333 & -0.417 \end{pmatrix}$$

$$\mathrm{geninv}(A)\cdot A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\mathrm{rref}(A) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}$$

$$\mathrm{rref}\left(\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}\right) = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 2 \end{pmatrix}$$

В любом месте документа допускается как переопределение каждого из элементов массива, так и изменение его размерности. Чтобы поменять размерность всего массива, необходимо присвоить любое значение новому элементу, индексы которого выходят за границы прежней размерности.

1.2.5. Стандартные и пользовательские функции. Произвольные зависимости между входными и выходными параметрами задаются при помощи функций. Функции принимают набор параметров и возвращают значение, скалярное или векторное (матричное). В формулах можно применять стандартные встроенные функции, а также функции, определенные пользователем.

Чтобы использовать функцию в выражении, надо определить значения входных параметров в скобках после имени функции. Имена простейших математических функций можно ввести с панели инструментов **Arithmetic (Счет)**.

Информация о других функциях содержится в справочной системе. Вставить в выражение стандартную функцию можно командой **Insert ➤ Function (Вставка ➤ Функция)**. В диалоговом окне **Insert Function** (**Вставка функции**) слева выбирается категория, к которой относится функция, а справа – конкретная функция. Пользовательские функции должны быть сначала определены. Определение задается при помощи оператора присваивания. В левой части указывается имя пользовательской функции и в скобках формальные параметры – переменные, от которых она зависит. Справа от знака присваивания эти переменные должны использоваться в выражении.

При применении пользовательской функции в последующих формулах ее имя вводят вручную.

1.3. Ввод формул и текста

Документ программы Mathcad называется рабочим листом. Он содержит следующие объекты: формулы и текстовые блоки. В ходе расчетов формулы обрабатываются последовательно, слева направо и сверху вниз, а текстовые блоки игнорируются.

Ввод информации осуществляется в месте расположения курсора. Программа Mathcad использует три вида курсоров. Если ни один объект не выбран, применяется крестообразный курсор, определяющий место создания следующего объекта. При вводе формул используется уголковый курсор, указывающий текущий элемент выражения, при вводе данных в текстовый блок – текстовый курсор в виде вертикальной черты.

Формулы – основные объекты рабочего листа. Новый объект по умолчанию является формулой. Чтобы начать ввод формулы, надо установить крестообразный курсор в нужное место и приступить к вводу букв, цифр, знаков операций. При этом создается область формулы, в которой появляется уголковый курсор, охватывающий текущий элемент формулы, например имя переменной (функции) или число. При вводе бинарного оператора по другую сторону знака операции автоматически появляется заполнитель в виде черного прямоугольника. В это место вводят очередной операнд.

Для управления порядком операций используют скобки, которые можно вводить вручную. Уголковый курсор позволяет автоматизировать такие действия. Чтобы выделить элементы формулы, которые в рамках операции должны рассматриваться как единое целое, используют клавишу **ПРОБЕЛ**. При каждом ее нажатии уголковый курсор «расширяется», охватывая элементы формулы, примыкающие к данному фрагменту. После ввода знака операции элементы в пределах уголкового курсора автоматически заключаются в скобки.

Элементы формул можно вводить с клавиатуры или с помощью специальных панелей управления. Панели управления открывают, используя меню **View (Вид)** или кнопками панели управления **Math** (**Математика**). Для ввода элементов формул предназначены следующие панели управления:

- **Arithmetic (Счет)** – для ввода чисел, знаков типичных математических операций и наиболее часто употребляемых стандартных функций;
- **Evaluation (Вычисление)** – для ввода операторов вычисления и знаков логических операций;
- **Graph (График)** – для построения графиков;
- **Matrix (Матрица)** – для ввода векторов и матриц и задания

матричных операций;

- **Calculus (Исчисление)** – для задания операций, относящихся к математическому анализу;
- **Greek (Греческий алфавит)** – для ввода греческих букв;
- **Symbolic (Аналитические вычисления)** – для управления аналитическими преобразованиями.

Текст, помещенный в рабочий лист, содержит комментарии и описания и предназначен для ознакомления, а не для использования в расчетах. Программа Mathcad определяет назначение текущего блока автоматически при первом нажатии клавиши **ПРОБЕЛ**. Если введенный текст не может быть интерпретирован как формула, блок преобразуется в текстовый и последующие данные рассматриваются как текст. Создать текстовый блок позволяет команда **Insert ➢ Text Region (Вставка ➢ Текстовый блок)**. Иногда требуется встроить формулу внутрь текстового блока. Для этого служит команда **Insert ➢ Math Region (Вставка ➢ Формула)**.

1.4. Форматирование формул и текста

Для форматирования формул и текста в программе Mathcad используется панель инструментов **Formatting (Форматирование).** С ее помощью можно индивидуально отформатировать любую формулу или текстовый блок, задав гарнитуру и размер шрифта, а также полужирное, курсивное или подчеркнутое начертание символов. В текстовых блоках можно также задавать тип выравнивания и применять маркированные и нумерованные списки.

В качестве средств автоматизации используются стили оформления. Выбрать стиль оформления текстового блока или элемента формулы можно из списка **Style (Стиль)** на панели инструментов **Formatting (Форматирование)**. Для формул и текстовых блоков применяются разные наборы стилей.

Для изменения стиля оформления формулы или создания нового стиля используется команда **Format ➢ Equation (Формат ➢ Выражение)**. Изменение стандартных стилей **Variables (Переменные)** и **Constants (Константы)** влияет на отображение формул по всему документу. Стиль оформления имени переменной учитывается при ее определении. Так, переменные **x** и *x* рассматриваются как различные и не взаимозаменяемы.

При оформлении текстовых блоков можно использовать более обширный набор стилей. Настройка стилей текстовых блоков производится командой **Format ➢ Style (Формат ➢ Стиль)**.

1.5. Решение уравнений и систем

Для численного поиска корней уравнения в программе Mathcad используется встроенная функция *root*. Она служит для решения уравнений вида $f(x) = 0$, где $f(x)$ – выражение, корни которого нужно найти, а x – неизвестное. Чтобы найти корни с помощью функции *root*, надо присвоить искомой переменной начальное значение, а затем вычислить корень при помощи вызова функции: *root*($f(x)$, x). Здесь $f(x)$ – функция переменной x, используемой в качестве второго параметра. Функция *root* возвращает значение независимой переменной, обращающее функцию $f(x)$ в 0. Например:

$$x := 1$$
$$\mathrm{root}(2\cdot\sin(x) - x, x) = 1.895$$

Если уравнение имеет несколько корней (как в данном примере), то результат, выдаваемый функцией *root*, зависит от выбранного начального приближения:

$$x := 0$$
$$\mathrm{root}(2\cdot\sin(x) - x, x) = 0$$

Для решения уравнения или систем уравнений (неравенств) численным методом в системе Mathcad можно использовать так называемый блок решения, который начинается с ключевого слова *given* (дано) и заканчивается вызовом функции *find* (найти). Между ними располагают «логические утверждения», задающие ограничения на значения искомых величин, иными словами, уравнения и неравенства. Всем переменным, используемым для обозначения неизвестных величин, должны быть заранее присвоены начальные значения. Например:

$$x := 2$$

Given

$$2\cdot\sin(x) - x = 0$$
$$\mathrm{Find}(x) = 1.895$$

1.6. Построение графиков

Если нужно построить двухмерный график в прямоугольных декартовых координатных осях X – Y, дают команду **Insert ➢ Graph ➢ X – Y Plot (Вставка ➢ График ➢ Декартовы координаты)**. В области размещения графика находятся заполнители для указания отображаемых выражений и диапазона изменения величин. Заполнитель у середины оси координат предназначен для переменной или выражения, отображаемого по этой оси. Обычно используют диапазон или вектор значений. Граничные значения по осям выбираются автоматически в соответствии с диапазоном изменения величины, но их можно задать и вручную.

В одной графической области можно построить несколько графиков (рис. 3). Для этого надо у соответствующей оси перечислить несколько выражений через запятую. Например:

$$z1(x) := \cos(x) \quad z2(x) := x \quad z(x) := z1(x) \cdot z2(x)$$

$$x := -4, -3.9 .. 4$$

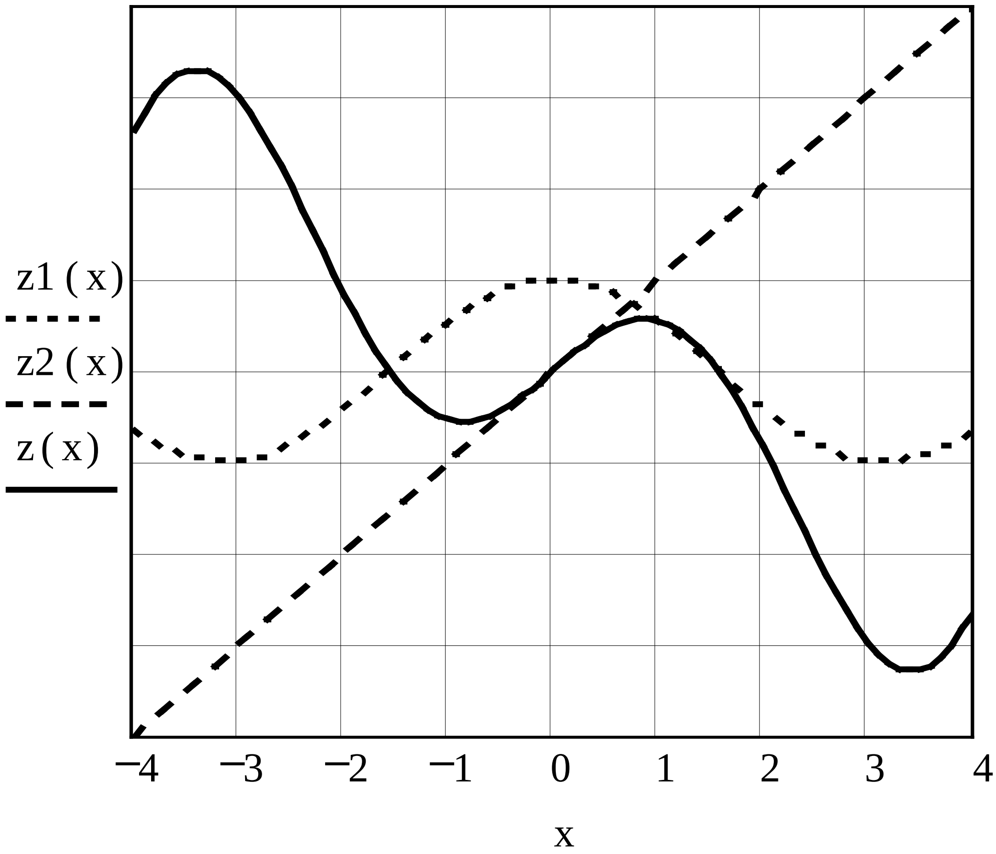


Рис. 3. Графика Mathcad

Разные кривые изображаются разным цветом и начертанием, а для форматирования графика надо дважды щелкнуть на области графика для вызова окна форматирования (рис. 4).

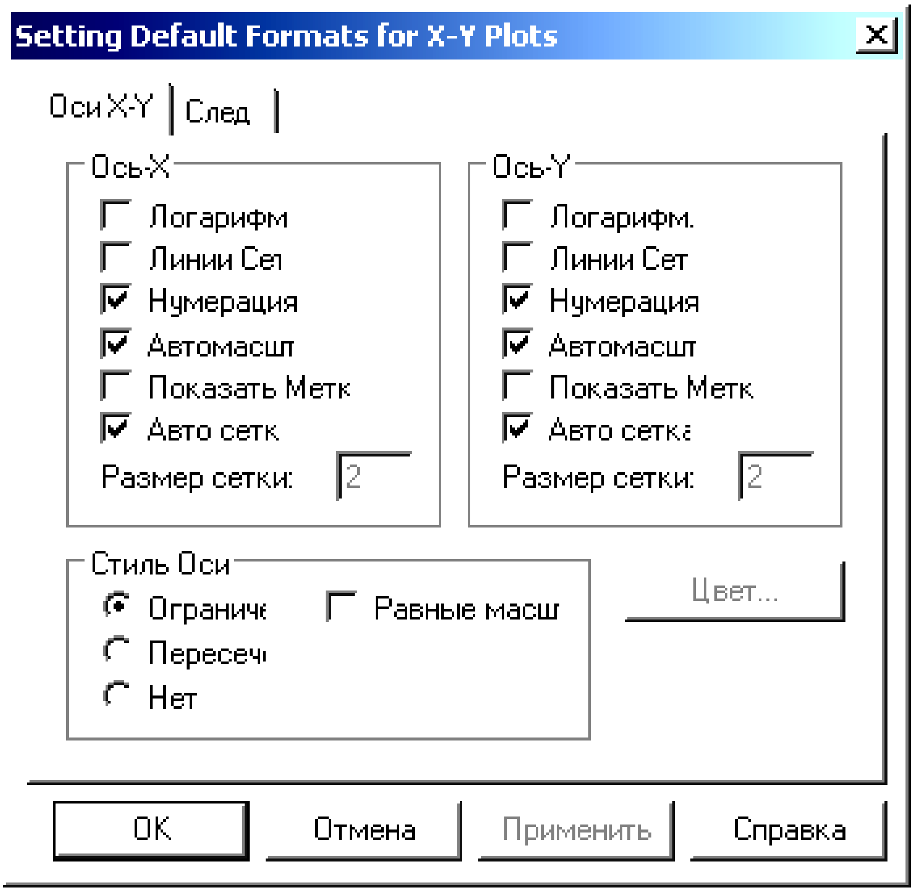


Рис. 4. Окно для форматирования графика

Для управления отображением построенных линий служит вкладка **Traces (Линии)** в открывшемся диалоговом окне. Текущий формат каждой линии приведен в списке, а под списком расположены элементы управления, позволяющие изменять формат. Поле **Legend Label (Описание)** задает описание линии, которое отображается только при сбросе флажка **Hide Legend (Скрыть описание)**.

Список **Symbol (Символ)** позволяет выбрать маркеры для отдельных точек, список **Line (Тип линии)** задает тип линии, список **Color (Цвет)** – цвет. Список **Type (Тип)** определяет способ связи отдельных точек, а список **Width (Толщина)** – толщину линии.

1.7. Аналитические вычисления

С помощью аналитических вычислений находят аналитические или полные решения уравнений и систем, а также проводят преобразования сложных выражений (например, упрощение). Иначе говоря, при таком подходе можно получить нечисловой результат.

В программе Mathcad конкретные значения, присвоенные переменным, при этом игнорируются – переменные рассматриваются как неопределенные параметры. Команды для выполнения аналитических вычислений в основном сосредоточены в меню **Symbolics (Аналитические вычисления)**.

Чтобы упростить выражение (или часть выражения), надо выбрать его при помощи уголкового курсора и дать команду **Symbolics ➢ Simplify (Аналитические вычисления ➢ Упростить)**. При этом выполняются арифметические действия, сокращаются общие множители и приводятся подобные члены, применяются тригонометрические тождества, упрощаются выражения с радикалами, а также выражения, содержащие прямую и обратную функции. Некоторые действия по раскрытию скобок и упрощению сложных тригонометрических выражений требуют применения команды **Symbolics ➢ Expand (Аналитические вычисления ➢ Раскрыть)**.

Команду **Symbolics ➢ Simplify (Аналитические вычисления ➢ Упростить**) применяют и в более сложных случаях. Например, с ее помощью можно:

- вычислить предел числовой последовательности, заданной общим членом;
- найти общую формулу для суммы членов числовой последовательности, заданной общим членом;
- вычислить производную данной функции;
- найти первообразную данной функции или значение неопределенного интеграла.

Другие возможности меню **Symbolics** (**Аналитические вычисления)** состоят в выполнении аналитических операций, ориентированных на переменную, использованную в выражении. Для этого надо выделить в выражении переменную и выбрать команду из меню **Symbolics ➢ Variable** (**Аналитические вычисления ➢ Переменная)**. Команда **Solve (Решить)** ищет корни функции, заданной данным выражением, например, если выделить уголковым курсором переменную x в выражении $ax^2 + bx + c = 0$, то в результате применения команды **Symbolics ➢ Variable ➢ Solve (Аналитические вычисления ➢ Переменная ➢ Решить)** будут найдены все корни.

2. ИСПОЛЬЗОВАНИЕ ПРОГРАММЫ MATHCAD ДЛЯ РЕШЕНИЯ ИНЖЕНЕРНО-ЭКОНОМИЧЕСКИХ ЗАДАЧ

2.1. Решение линейных и нелинейных уравнений и их систем

Решение уравнений – алгебраических или трансцендентных – представляет собой одну из существенных задач прикладного анализа, потребность, которая возникает в многочисленных и самых разнообразных разделах физики, механики, техники и естествознания в широком смысле этого слова.

Методы решения уравнений делятся на две группы. К первой группе относятся такие методы, которые позволяют путем различных преобразований упростить уравнение и записать решение в виде формулы. Вторую группу составляют численные методы. Решение в этом случае ищется с помощью операций (при желании только арифметических) над числами, причем ответ получается сразу в виде числа (обычно приближенного). К первой группе относится, например, метод Кардано для решения уравнений третьей степени, способ понижения степени возвратных уравнений и т. д. Эти методы могут быть иногда полезны, однако практическая их ценность в общем невелика, так как между формулой и численным ответом, необходимым, как правило, в практических задачах, лежит еще большое количество вычислений. Кроме того, эти методы применимы лишь к сравнительно узкому кругу задач.

В настоящее время существует и применяется большое количество численных методов решения уравнений и их систем. Это методы итераций, Ньютона, Гаусса, Зейделя и др. [7]. Разнообразие методов свидетельствует о нетривиальности решения данной задачи и множестве практических сфер ее применения, кроме этого, связано со следующими обстоятельствами:

1) решение уравнения или системы уравнений редко осуществляется от начала до конца одним способом. Обычно вначале корни уравнений определяются каким-нибудь методом со сравнительно небольшой точностью, а затем «уточняются» с помощью других методов, достигая заданной точности;

2) не существует методов, одинаково эффективных для всех уравнений или систем уравнений. Естественно, что для одних уравнений более удобны одни методы, для других – другие. Это различие не всегда можно определить заранее. Поэтому в процессе решения

численными методами приходится иногда отказываться от одного метода и переходить к другому;

3) решение уравнений численными методами требует довольно значительной вычислительной работы, что вызывает необходимость разработки алгоритма и написания программы на одном из языков программирования.

Система Mathcad решает линейные уравнения и системы уравнений (кстати, как и нелинейные) итерационным методом. При этом может использоваться подстановка одних переменных в другие, нередко сводящая задачу к точному решению.

Главное достоинство системы Mathcad в том, что при решении таких задач не требуется никаких записей в программах, кроме записи исходного уравнения. Естественно, это единственная «неприятность», от которой уже нельзя избавиться, – «неприятность» начального приближения в случае использования численного метода решения и оператора для решения уравнения.

Пример 2.1. Найти корни уравнения $e^x - \cos x = 0$ численным методом.

В окне редактора Mathcad программа имеет вид

$$x := 0$$

$$\mathrm{root}(e^x - \cos(x), x) = 0$$

Зная, что это уравнение имеет не один корень, зададим другое начальное приближение, тогда имеем

$$x := -2$$

$$\mathrm{root}(e^x - \cos(x), x) = -1.292$$

Этим самым мы определим следующий корень исходного уравнения на интервале (–2, –1).

Используя блок *given-find*, в окне редактора будем иметь

Начальное приближение

$$x := 0$$

Булевское выражение

given

$$e^x - \cos(x) = 0$$

Получение решения

$$\mathrm{find}(x) = 0$$

Пример 2.2. Найти корни уравнения $e^x - \cos x = 0$ аналитическим методом.

Для решения этой задачи в Mathcad имеются три дублирующих друг друга способа.

1. Выделив уголковым курсором переменную x в выражении $\exp(x) - \cos(x) = 0$, применив команду **Symbolics ➢ Variable ➢ Solve (Символы ➢ Переменная ➢ Вычислить)**, будут найдены все корни.

2. Используя блок *given-find*, в окне редактора будем иметь

$$\text{given}$$

$$e^x - \cos(x) = 0$$

$$\text{find}(x) \rightarrow 0$$

3. Используя оператор *root*, в окне редактора будем иметь

$$\text{root}(e^x - \cos(x), x) \rightarrow 0$$

В MathCad есть эффективный способ вычисления корней уравнения для частного случая, когда функция $f(x)$ представляет собой полином (многочлен) n-го порядка:

$$f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n.$$

Известно, что уравнение, в котором левая часть представляет собой полином n-го порядка, имеет n корней. Среди них могут быть как вещественные, так и комплексные корни. Для решения уравнения $f(x) = 0$ с полиномиальной правой частью такого вида используется функция *polyroots* (v), где v – вектор коэффициентов полинома, записанных в последовательности от низшего (нулевого) коэффициента к высшим.

Пример 2.3. Найти все корни полинома $f(x) = 6 - 2x + 7x^2 - 8x^3$.

Решение. Используя функцию *polyroots*, имеем запись в окне редактора системы Mathcad

$$f(x) := 6 - 2 \cdot x + 7 \cdot x^2 - 8 \cdot x^3$$

$$v := (6\ {-2}\ 7\ {-8})^T$$

$$x := \text{polyroots}(v)$$

$$x = \begin{pmatrix} -0.159 + 0.777i \\ -0.159 - 0.77i \\ 1.193 \end{pmatrix}$$

При решении уравнения получен один вещественный и два комплексно-сопряженных корня. Применение функции *polyroots* по сравнению с функцией *root* для уравнений с полиномиальной правой частью предпочтительнее по двум причинам:

- не требуется выполнения этапа отделения корней;
- функция позволяет найти как вещественные, так и комплексные корни.

При решении систем линейных и нелинейных уравнений Mathcad использует итерационные методы. Причем часто решение систем линейных уравнений относят к некорректным задачам. Так случается, если задана система, у которой не хватает переменных (т. е. переопределена матрица A коэффициентов a_{ij}).

Итерационный метод, реализованный в Mathcad, позволяет легко решать такие задачи, обычно требующие специальных методов. Однако попытки решения другой ненормальной системы (с вырожденной матрицей, $det|A| = 0$) к успеху обычно не приводят.

На практике нет общих методов решения систем нелинейных уравнений, которые давали бы гарантию правильного решения или даже решения вообще. Нередки случаи, когда решение зацикливается и с заданной точностью получить его просто невозможно. Известны и случаи, когда из-за неудачного и произвольного выбора начальных значений переменных решение обречено на неудачу. Кроме того, заведомо неизвестна область существования корней, и некоторые корни при решении можно попросту проскочить. Система Mathcad это исключает. Причем она позволяет решать системы нелинейных уравнений, содержащие ограничения, обычно выражаемые неравенствами.

Ограничения – еще одно мощное средство, позволяющее отбросить нереальные или неинтересующие нас решения.

Пример 2.4. Определить точку пересечения окружности с центром в точке $O(0, 0)$ и радиусом $R = 4$ с параболой $y = x^2$ в первой четверти декартовой системы координат.

Решение. Эти кривые в силу симметрии параболы относительно оси Oy пересекаются в двух точках. Для определения одной из них нам необходимо решить систему уравнений

$$\begin{cases} x^2 + y^2 = 16, \\ y = x^2, \end{cases}$$

используя начальные приближения $x = 0$, $y = 0$, что удовлетворяет требованиям данной задачи. Листинг программы данной задачи имеет следующий вид:

$$x := 0$$

$$y := 0$$

Given

$$x^2 + y^2 = 16$$

$$y = x^2$$

$$\mathrm{Find}(x,y) = \begin{pmatrix} 1.879 \\ 3.531 \end{pmatrix}$$

К системам линейных уравнений сводится множество, если не сказать большинство, задач вычислительной математики, поэтому одним из основных вопросов вычислительной линейной алгебры является решение систем линейных алгебраических уравнений (СЛАУ), т. е. систем уравнений вида

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n = b_m$$

В матричной форме СЛАУ записывается в эквивалентном виде:

$$A \cdot x = b,$$

где A – матрица коэффициентов СЛАУ размерности $m \times n$; x – вектор неизвестных; b – вектор правых частей уравнений.

СЛАУ имеет единственное решение, если матрица A является невырожденной, или, по-другому, несингулярной, т. е. ее определитель не равен нулю.

С вычислительной точки зрения решение СЛАУ не представляет трудностей, если матрица A не очень велика. С большой матрицей проблем также не возникнет, если она не очень плохо обусловлена. В Mathcad СЛАУ можно решить как в более наглядной форме, используя вычислительный блок *given-find*, так и в более удобной для записи формах, используя встроенную функцию *lsolve* или матричную запись для решения системы линейных уравнений.

Пример 2.5. Решить систему уравнений:

$$\begin{cases} 2x + 5y + 3z = 14; \\ 7x + 4y + 9z = 24; \\ 3x + 8y + 6z = 34. \end{cases}$$

Решение. Представим решение этой СЛАУ, используя различные способы, и представим листинги решений в системе Mathcad:

Листинг 1. Решение СЛАУ с помощью блока *given-find*

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

Given

$$2 \cdot x + 5 \cdot y + 3 \cdot z = 14$$

$$7 \cdot x + 4 \cdot y + 9 \cdot z = 24$$

$$3 \cdot x + 8 \cdot y + 6 \cdot z = 34$$

$$\mathrm{Find}(x, y, z) = \begin{pmatrix} 1.977 \\ 1.816 \\ 0.322 \end{pmatrix}$$

Листинг 2. Решение СЛАУ с помощью функции *lsolve*

$$A := \begin{pmatrix} 2 & 5 & 3 \\ 7 & 4 & 9 \\ 3 & 8 & 6 \end{pmatrix} \qquad b := \begin{pmatrix} 14 \\ 24 \\ 34 \end{pmatrix}$$

$$\mathrm{Lsolve}(A, b) = \begin{pmatrix} 1.977 \\ 1.816 \\ 0.322 \end{pmatrix}$$

Листинг 3. Решение СЛАУ матричным методом

$$A := \begin{pmatrix} 2 & 5 & 3 \\ 7 & 4 & 9 \\ 3 & 8 & 6 \end{pmatrix} \qquad b := \begin{pmatrix} 14 \\ 24 \\ 34 \end{pmatrix}$$

$$x := A^{-1} \cdot b$$

$$x = \begin{pmatrix} 1.977 \\ 1.816 \\ 0.322 \end{pmatrix}$$

Встроенную функцию *lsolve* допускается применять и при символьном решении СЛАУ (*листинг 4*).

Листинг 4. Символьное решение СЛАУ

$$A := \begin{pmatrix} 2 & 5 & 3 \\ 7 & 4 & 9 \\ 3 & 8 & 6 \end{pmatrix} \qquad b := \begin{pmatrix} 14 \\ 24 \\ 34 \end{pmatrix}$$

$$\mathrm{Lsolve}(A, b) \rightarrow \begin{pmatrix} 1.977 \\ 1.816 \\ 0.322 \end{pmatrix}$$

2.2. Поиск экстремумов функций

Функция $y = f(x)$ называется возрастающей (убывающей) в некотором промежутке, если для любых точек x_1 и x_2, принадлежащих данному промежутку, из неравенства $x_1 < x_2$ следует неравенство $f(x_1) < f(x_2)$.

Знак первой производной указывает на возрастание или убывание функции на заданном промежутке. Если в некотором промежутке производная данной функции положительна, то функция возрастает в этом промежутке, если отрицательна – убывает на данном промежутке.

Максимумом (минимумом) функции $y = f(x)$ называется такое ее значение $y_i = f(x_i)$, которое больше (меньше) всех других ее значений, принимаемых в точках x, достаточно близких к точке x_i и отличных от нее.

Максимум и минимум называются экстремумом функции. Значения аргумента и функции, при которых достигается экстремум, – точки экстремума.

Достаточными условиями экстремума являются следующие. Если в точке $x = x_0$ первая производная функции $y = f(x)$ равна нулю, а вторая производная отлична от нуля, то x_0 – точка экстремума, причем:

1) x_0 – точка максимума, если $f''(x_0) < 0$;

2) x_0 – точка минимума, если $f''(x_0) > 0$.

Чтобы найти наибольшее значение функции $y = f(x)$ на отрезке $[a, b]$, необходимо вычислить значения ее максимумов на этом отрезке, значения функции на ее концах, т. е. $f(a)$, $f(b)$, и из полученных чисел выбрать самое большое. Аналогично находится наименьшее значение функции.

На практике часто необходимо найти экстремум (или экстремумы) некоторой целевой функции $F(x_1, x_2, ..., x_n)$ n переменных x_i (проектных параметров). Такая функция описывает $(n + 1)$-мерную поверхность. Соответственно функция $F(x)$ одного параметра $x_i = x$ описывает некоторую кривую на плоскости.

Поиск экстремумов функции одной переменной является самостоятельной и часто встречаемой задачей. Кроме того, к нему сводится гораздо более сложная задача поиска экстремумов функций многих переменных.

В общем случае функция $F(x)$ может иметь несколько экстремумов (максимумов или минимумов). Из них главный (оптимальное решение для пространства проектирования) называется глобальным. Задача поиска экстремумов сводится к их локализации и уточнению значений x и $F(x)$ в точке экстремума. Будем также полагать, что на значения параметра x (если это особо не оговорено) накладываются ограничения в виде неравенств $a \leq x \leq b$, где a и b – границы интервала поиска. В пределах отрезка $[a, b]$ функцию считаем унимодальной, т. е. содержащей один экстремум.

Поиск экстремумов численными методами, как правило, разделяется на два этапа.

На первом этапе выделяются интервалы аргумента x, в которых существует единственная точка x^*, где функция $F(x)$ принимает экстремальное значение. Первый этап поиска решения задачи нахождения экстремума близок по идеологии к задаче отделения корней уравнений и не поддается строгой алгоритмизации. Обычно интервалы унимодальности находят на основе анализа упрощенных математических моделей исследуемых процессов.

Графическое представление исследуемой функции также помогает на первом этапе, хотя и требует значительных временных затрат, т. к. шаг изменения аргумента должен быть достаточно малым, чтобы не пропустить возможные экстремумы.

На втором этапе осуществляется уточнение местоположений экстремумов на интервалах унимодальности функций, используя различные методы (метод Фибоначчи, дихотомии и др.).

Для организации решения поиска максимума (минимума) функции одной переменной, используя МП Mathcad, достаточно задать саму функцию и определить так называемую целевую функцию с конкретным именем-параметром. Кроме того, необходимо задать установку *maximize* (имя-параметр) на поиск максимума целевой функции. Это указание запускает встроенную в систему стандартную программу поиска одного из экстремумов численным методом.

Для указания отрезка $[a, b]$ используется зарезервированное слово *given*.

Пример 2.6. Определить максимум функции $y(x) = e^{0.4x}\sin x$. Для реализации данной задачи программа на рабочем листе имеет вид

$$f(x) := e^{(0.4 \cdot x)} \cdot \sin(x)$$

$$x := 2$$

$$z := \mathrm{maximize}(f, x)$$

$$z = 1.951$$

2.3. Решение задач линейного и нелинейного программирования

Математическое программирование представляет собой математическую дисциплину, занимающуюся изучением экстремальных задач и разработкой методов их решения.

В общем виде математическая постановка экстремальной задачи состоит в определении наибольшего или наименьшего значения целевой функции $f(x_1, x_2, \ldots, x_n)$ при условиях $g_i(x_1, x_2, \ldots, x_n) \le b_i$ $(i = 1, \ldots, m)$, где f и g_i – заданные функции, а b_i – некоторые действительные числа.

Задачи математического программирования делятся на задачи линейного и нелинейного программирования. При этом если все функции f и g_i линейные, то соответствующая задача является задачей линейного программирования. Если же хотя бы одна из указанных функций нелинейная, то соответствующая задача является задачей нелинейного программирования.

Использование МП Mathcad позволяет последовательно находить решение задач, получающихся из исходной с помощью внесения изменений в исходные данные, а также построения различных целевых функций. Наряду с этим использование указанного пакета дает возможность получать отчеты, необходимые для проведения широкого послеоптимизационного анализа решения задач линейного и нелинейного программирования, а также проводить другие различные исследования, обусловленные рациональным подходом к решению задач математического программирования.

2.3.1. Задачи оптимизации. Найдем решение задачи, состоящей в определении максимального значения функции

$$F = c_1x_1 + c_2x_2 \qquad (1)$$

при условиях:

$$a_{i1}x_1+a_{i2}x_2 \le b_i, i = 1, \ldots, m), \qquad (2)$$

$$x_j \ge 0, j = (1, 2). \qquad (3)$$

Каждое из неравенств (2), (3) системы ограничений задачи геометрически определяет полуплоскость соответственно с граничными прямыми $a_{i1}x_1 + a_{i2}x_2 = b_i$ $(i = 1, m)$, $x_1 = 0$, $x_2 = 0$. Так как множество точек пересечения данных полуплоскостей выпуклое, то областью допустимых решений задачи (1) – (3) является выпуклое множество, которое называется многоугольником решений.

Таким образом, исходная задача линейного программирования состоит в нахождении такой точки многоугольника решений, в которой целевая функция F принимает максимальное значение. Эта точка существует тогда, когда многоугольник решений не пуст и на нем целевая функция ограничена сверху.

При указанных условиях в одной из вершин многоугольника решений целевая функция принимает максимальное значение. Для определения данной вершины построим линию уровня $c_1x_1 + c_2x_2 = h$, где h – некоторая постоянная, проходящая через многоугольник решений, и передвигать ее в направлении вектора $C = (c_1, c_2)$ до тех пор, пока она не пройдет через последнюю ее общую точку с многоугольником решений. Координаты указанной точки и определяют оптимальный план данной задачи.

Итак, нахождение решения задачи линейного программирования (1)– (3) на основе ее геометрической интерпретации включает следующие этапы:

- строят прямые, уравнения которых получаются в результате замены в ограничениях (2) и (3) знаков неравенств на знаки точных равенств;
- находят полуплоскости, определяемые каждым из ограничений задачи;
- находят многоугольник решений;
- строят вектор $C = (c_1, c_2)$;
- строят прямую $c_1x_1 + c_2x_2 = h$, проходящую через многоугольник решений;
- передвигают прямую $c_1x_1 + c_2x_2 = h$ в направлении вектора C, в результате чего либо находят точку (точки), в которой целевая функция принимает максимальное значение, либо устанавливают неограниченность сверху функции на множестве планов;
- определяют координаты точки максимума функции и вычисляют значение целевой функции в этой точке.

Пример 2.7. Для производства двух видов изделий *A* и *B* лесхоз использует три вида сырья (доска, брус, бревно). Нормы расхода сырья каждого вида на изготовление единицы продукции данного вида приведены в табл. 1. В ней же указаны прибыль от реализации одного изделия каждого вида и общее количество сырья данного вида, которое может быть использовано лесхозом.

Таблица 1

Виды сырья	Нормы расхода сырья, м^3, на одно изделие		Общее количество сырья, м^3
	A	*B*	
Доска	12	4	300
Брус	4	4	120
Бревно	3	12	252
Прибыль от реализации одного изделия (усл. ед)	30	40	–

Учитывая, что изделия *A* и *B* могут производиться в любых соотношениях (сбыт обеспечен), требуется составить такой план их выпуска, при котором прибыль лесхоза от реализации всех изделий является максимальной.

Решение. Предположим, что лесхоз изготовит x_1 изделий вида A и x_2 изделий вида B. Поскольку производство продукции ограничено имеющимся в распоряжении лесхоза сырьем каждого вида и количество изготовляемых изделий не может быть отрицательным, должны выполняться неравенства:

$$12x_1 + 4x_2 \leq 300,$$
$$4x_1 + 4x_2 \leq 120,$$
$$3x_1 + 12x_2 \leq 252,$$
$$x_1, x_2 > 0.$$

Общая прибыль от реализации x_1 изделий вида A и x_2 изделий вида B составит $F = 30x_1 + 40x_2$.

Таким образом, мы приходим к следующей математической задаче: среди всех неотрицательных решений данной системы линейных неравенств требуется найти такое, при котором функция F принимает максимальное значение.

Найдем решение сформулированной задачи, используя ее геометрическую интерпретацию. Сначала определим многоугольник решений. Для этого в неравенствах системы ограничений и условиях неотрицательности переменных знаки неравенств заменим на знаки точных равенств и найдем соответствующие прямые:

$$12x_1 + 4x_2 = 300,$$
$$4x_1 + 4x_2 = 120,$$
$$3x_1 + 12x_2 = 252,$$
$$x_1 = 0,$$
$$x_2 = 0.$$

Эти прямые изображены на рис. 5. Каждая из построенных прямых делит плоскость на две полуплоскости. Координаты точек одной полуплоскости удовлетворяют исходному неравенству, а другой – нет. Чтобы определить искомую полуплоскость, нужно взять какую-нибудь точку, принадлежащую одной из полуплоскостей, и проверить, удовлетворяют ли ее координаты данному неравенству. Если координаты взятой точки удовлетворяют данному неравенству, то искомой является та полуплоскость, которой принадлежит эта точка, в противном случае – другая полуплоскость.

Найдем, например, полуплоскость, определяемую неравенством $12x_1 + 4x_2 \leq 300$. Для этого построим прямую $12x_1 + 4x_2 = 300$, возьмем какую-нибудь точку, принадлежащую одной из двух полученных полуплоскостей, например точку $O(0; 0)$. Координаты этой точки удовлетворяют неравенству $12 \cdot 0 + 4 \cdot 0 < 300$; значит, полуплоскость, которой принадлежит точка $O(0, 0)$, определяется неравенством $12x_1 + 4x_2 \leq 300$. Это и показано стрелками на рис. 5.

Пересечение полученных полуплоскостей и определяет многоугольник решений данной задачи.

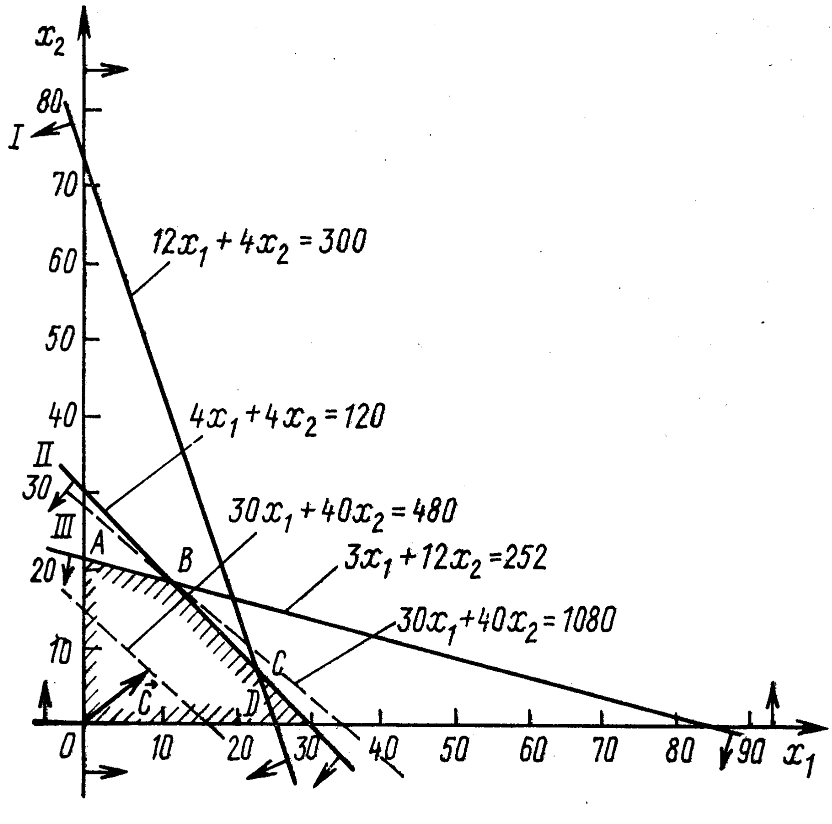


Рис. 5. Многоугольник решений

Как видно на рис. 5, многоугольником решений является пятиугольник *OABCD*. Координаты любой точки, принадлежащей этому пятиугольнику, удовлетворяют данной системе неравенств и условию неотрицательности переменных. Поэтому сформулированная задача будет решена, если мы сможем найти точку, принадлежащую пятиугольнику $OABCD$, в которой функция F принимает максимальное значение. Чтобы найти указанную точку, построим вектор $C = (30, 40)$ и прямую $30x_1 + 40x_2 = h$, где h – некоторая постоянная, такая, что прямая $30x_1 + 40x_2 = h$ имеет общие точки с построенным многоугольником решений. Положим, например, $A = 480$ и построим прямую $30x_1 + 40x_2 = 480$ (рис. 5).

Если теперь взять какую-нибудь точку, принадлежащую построенной прямой и многоугольнику решений, то ее координаты определяют такой план производства изделий A и B, при котором прибыль от их реализации равна 480 (усл. ед.). Далее, полагая h равным некоторому числу, большему, чем 480, мы будем получать различные параллельные прямые. Если они имеют общие точки с многоугольником решений, то эти точки определяют планы производства изделий A и B, при которых прибыль от их реализации превзойдет 480 (усл. ед.).

Перемещая построенную прямую $30x_1 + 40x_2 = 480$ в направлении вектора C, видим, что последней общей точкой ее с многоугольником решений задачи служит точка B. Координаты этой точки и определяют план выпуска изделий A и B, при котором прибыль от их реализации является максимальной.

Найдем координаты точки B как точки пересечения двух прямых. Следовательно, ее координаты удовлетворяют уравнениям этих прямых:

$$4x_1 + 4x_2 = 120,$$

$$3x_1 + 12x_2 = 252.$$

Решив эту систему уравнений, получим $x_1^* = 12$, $x_2^* = 18$.

Следовательно, если лесхоз изготовит 12 изделий вида A и 18 изделий вида B, то он получит максимальную прибыль, равную $F_{\max} = 30 \cdot 12 + 40 \cdot 18 = 1080$ усл. ед.

В настоящее время решение этих задач становится актуальным для признания рентабельности производства при изготовлении тех или иных изделий. Поэтому, если целевая функция зависит от двух переменных, то, как видно из примера, эту задачу можно решить графическим путем, в противном случае – задача линейного программирования, когда целевая функция F зависит от многих факторов и имеется множество ограничений, в общем случае требует применения трудоемких вычислений.

Покажем на данном примере, сколь просто справляется Mathcad с решением этой довольно головоломной задачи. В рабочей области окна системы Mathcad программа имеет вид

$$f(x1, x2) := 30 \cdot x1 + 40 \cdot x2$$

$$x1 := 10$$

$$x2 := 15$$

Given

$$x1 \geq 0$$

$$x2 \geq 0$$

$$12 \cdot x1 + 4 \cdot x2 \leq 300$$

$$4 \cdot x1 + 4 \cdot x2 \leq 120$$

$$3 \cdot x1 + 12 \cdot x2 \leq 252$$

$$R := \mathrm{Maximize}(f, x1, x2)$$

$$R = \begin{pmatrix} 12 \\ 18 \end{pmatrix}$$

Рассмотрим другие задачи линейного программирования, используемые в экономических и производственных расчетах.

Пример 2.8. Цех предприятия должен изготовить 100 изделий трех типов. Каждого изделия нужно не менее 20 шт. На изделия уходит соответственно 4, 5, и 2 кг однородного металла при его общем запасе 340 кг, а также по 5, 9 и 2 кг пластмассы при ее общем запасе 700 кг.

Сколько изделий каждого типа надо выпустить для получения максимального объема выпуска в денежном выражении, если цена изделий составляет по калькуляции 4, 3 и 2 усл. ед.?

Для наглядности листинга программы в окне редактора Mathcad введем комментарий в виде объекта текст. Для ввода текста следует выполнить команду **Вставка ➤ Текстовая область** или нажать клавишу двойной кавычки «"» и установить вид и размер шрифта, так же как и в текстовом редакторе Word.

Данная производственная задача явно сводится к задаче вычисления максимума функции с ограничениями. Используя функцию *maximize* системы Mathcad, программа в окне редактора Mathcad имеет следующий вид:

Листинг.

Обозначим x1, x2 и x3 – количество изделий 1, 2 и 3 вида.

Целевая функция:

$$f(x1, x2, x3) := 4 \cdot x1 + 3 \cdot x2 + 2 \cdot x3$$

Произвольные начальные приближения:

$$x1 := 1 \quad x2 := 1 \quad x3 := 1$$

Блок решений и ограничений:

Given

$$x1 \geq 0$$

$$x2 \geq 0$$

$$x3 \geq 0$$

$$4x1 + 5x2 + 2x3 \leq 340$$

$$5x1 + 9x2 + 2x3 \leq 700$$

$$x1 + x2 + x3 = 100$$

$$R := \mathrm{maximize}(f, x1, x2, x3)$$

$$R = \begin{pmatrix} 70 \\ 0 \\ 30 \end{pmatrix}$$

Максимальная прибыль:

$$f(R_0, R_1, R_2) = 340$$

Полученное решение $x1 = 70$, $x2 = 0$, $x3 = 30$ позволит получить максимальную прибыль в денежном эквиваленте $f = 340$ усл. ед.

Пример 2.9. Продукцией городского молочного завода являются молоко, кефир и сметана, расфасованные в бутылки. На производство 1 т молока, кефира и сметаны требуется соответственно 1010, 1010 и 9450 кг молока. При этом затраты рабочего времени при разливе 1 т молока и кефира составляют 0,18 и 0,19 машино-ч. На расфасовке 1 т сметаны заняты специальные автоматы в течение 3,25 ч. Всего для производства цельномолочной продукции завод может использовать 136 000 кг молока. Основное оборудование может быть занято в течение 21,4 машино-ч, а автоматы по расфасовке сметаны –

в течение 16,25 ч. Прибыль от реализации 1 т молока, кефира и сметаны соответственно равна 30, 22 и 136 руб. Завод должен ежедневно производить не менее 100 т молока, расфасованного в бутылки. На производство другой продукции не имеется никаких ограничений. Требуется определить, какую продукцию и в каком количестве следует ежедневно изготовлять заводу, чтобы прибыль от ее реализации была максимальной.

Данная производственная задача также сводится к задаче вычисления максимума функции с ограничениями. Используя функцию *maximize* системы Mathcad, программа в окне редактора Mathcad имеет следующий вид:

Листинг.

Предположим, что молочный завод будет ежедневно производить x1 т молока, x2 т кефира и x3 т сметаны.

Тогда целевая функция:

$$f(x1, x2, x3) := 1010 \cdot x1 + 1010 \cdot x2 + 9450 \cdot x3$$

Произвольные начальные приближения:

$$x1 := 1 \quad x2 := 1 \quad x3 := 1$$

Блок решений и ограничений:

Given

$$x1 \geq 0$$

$$x2 \geq 0$$

$$x3 \geq 0$$

$$1010 \cdot x1 + 1010 \cdot x2 + 9450 \cdot x3 \leq 136\,000$$

$$0.18 \cdot x1 + 0.19 \cdot x2 \leq 21.4$$

$$x1 \leq 100$$

$$R := \mathrm{maximize}(f, x1, x2, x3)$$

$$R = \begin{pmatrix} 100 \\ 5 \\ 3.169 \end{pmatrix}$$

Максимальная прибыль:

$$f(R_0, R_1, R_2) = 1.36 \times 10^5$$

Полученное решение $x1 = 100$, $x2 = 5$ и $x3 = 3.169$ позволит получить максимальную прибыль $f = 136\,000$ усл. ед.

Пример 2.10. Лесхозу требуется не более 10 трехтонных и не более 8 пятитонных автомашин. Отпускная цена автомашин: первой марки – 3000 усл. ед., второй марки – 6000 усл. ед. Лесхоз может выделить для приобретения автомашин от 18 до 72 тыс. усл. ед.

Сколько следует приобрести автомашин каждой марки в отдельности, чтобы их суммарная грузоподъемность была максимальной. Определить максимальную суммарную грузоподъемность.

Данная производственная задача также сводится к задаче вычисления максимума функции с ограничениями. Используя функцию *maximize* системы Mathcad, программа в окне редактора Mathcad имеет следующий вид.

Листинг.

Обозначим x1, x2 – количество трехтонных и пятитонных автомашин.

Целевая функция:

$$f(x1, x2) := 3\cdot x1 + 5\cdot x2$$

Произвольные начальные приближения:

$$x1 := 1 \quad x2 := 1$$

Блок решений и ограничений:

Given

$$10 \ge x1 \ge 0$$

$$8 \ge x2 \ge 0$$

$$3000x1 + 6000x2 \ge 18\,000$$

$$3000x1 + 6000x2 \le 72\,000$$

$$R := \text{maximize}(f, x1, x2)$$

$$R = \begin{pmatrix} 10 \\ 7 \end{pmatrix}$$

Максимальная грузоподъемность:

$$f(R_0, R_1) = 65$$

Системе совершенно безразличны новые трудности, возникающие вследствие нелинейности целевой функции или ограничений.

Итерационные процедуры оптимизации системы Mathcad универсальны и позволяют решать как линейные, так и нелинейные задачи оптимизации.

2.3.2 Транспортная задача. Общая постановка транспортной задачи состоит в определении оптимального плана перевозок некоторого однородного груза из m пунктов отправления $A_1, A_2, ..., A_m$ в n пунктов назначения $B_1, B_2, \ldots, B_n$. При этом в качестве критерия оптимальности обычно берется либо минимальная стоимость перевозок всего груза, либо минимальное время его доставки. Рассмотрим транспортную задачу, в качестве критерия оптимальности которой взята минимальная стоимость перевозок всего груза. Обозначим через c_{ij} тарифы перевозки единицы груза из i-го пункта отправления в j-й пункт назначения, через $a_{i,}$ – запасы груза в i-м пункте отправления, через b_j – потребности в грузе в j-м пункте назначения, а через x_{ij} – количество единиц груза, перевозимого из i-го пункта отправления в j-й пункт назначения. Тогда математическая постановка задачи состоит в определении минимального значения функции:

$$F = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

при условиях:

$$\sum_{i=1}^{m} x_{ij} = b_j \ \ (j = 1 \ldots n),$$

$$\sum_{i=1}^{n} x_{ij} = a_i \ \ (i = 1 \ldots m),$$

$$x_{\text{ij}} \geq 0.$$

Поскольку переменные x_{ij} удовлетворяют записанным системам линейных уравнений и условию неотрицательности, обеспечиваются доставка необходимого количества груза в каждый из пунктов назначения, вывоз имеющегося груза из всех пунктов отправления, а также исключаются обратные перевозки. Всякое неотрицательное решение этих систем линейных уравнений, определяемое матрицей $X = x_{ij}$, называется планом транспортной задачи.

План $X^* = x_{ij}^*$, при котором целевая функция F принимает свое минимальное значение, называется оптимальным планом транспортной задачи.

Обычно исходные данные транспортной задачи записывают в виде табл. 2.

Таблица 2

Пункты отправления	Пункты назначения					Запасы
	B_1	…	B_j	…	B_n	
A_1	c_{11}	…	c_{1j}	…	c_{1n}	a_1
…	…	…	…	…	…	…
A_i	c_{i1}	…	c_{ij}	…	c_{in}	a_i
…	…	…	…	…	…	…
A_m	c_{m1}	…	c_{mj}	…	c_{mn}	a_m
Потребности	b_1	…	b_j	…	b_n	

Очевидно, общее наличие груза у поставщиков равно $\sum_{i=1}^{m} a_i$, а общая потребность в грузе в пунктах назначения равна $\sum_{j=1}^{n} b_j$.

Если общая потребность в грузе в пунктах назначения равна запасу груза в пунктах отправления, т. е.

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j,$$

то модель такой транспортной задачи называется закрытой. Если же указанное условие не выполняется, то модель транспортной задачи называется открытой.

Число переменных x_{ij} в транспортной задаче с m пунктами отправления и n пунктами назначениями равно $n \cdot m$, а число уравнений в ограничениях равно $n + m$. Следовательно, опорный план транспортной задачи может иметь не более $n + m - 1$ отличных от нуля неизвестных.

Если в опорном плане число отличных от нуля компонент равно в точности $n + m - 1$, то план является невырожденным, а если меньше – то вырожденным.

Рассмотрим простую закрытую транспортную задачу с не-большим числом неизвестных: $m = 2$, $n = 3$. Исходные данные закрытой транспортной задачи представим в виде табл. 3.

Таблица 3

Пункты отправления	Пункты назначения			Запасы
	B_1	B_2	B_3	
A_1	45	46	76	400
A_2	43	57	54	500
Потребности	400	300	200	

.Для набора листинга решения исходной задачи необходимы две палитры **Булево** и **Вычисления**, показанные на рис. 6.

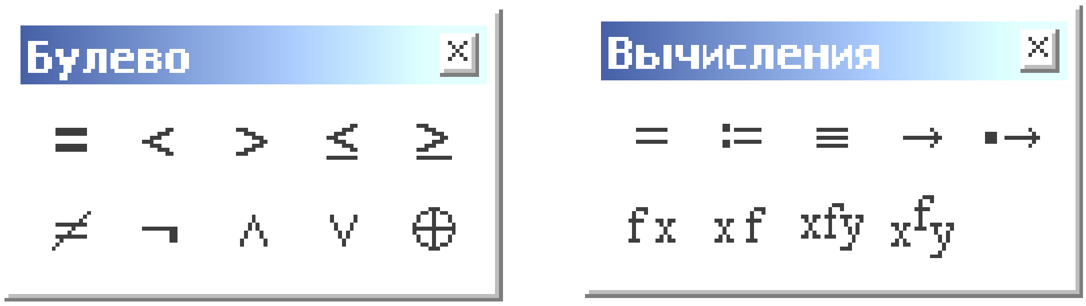


Рис. 6. Палитры Булево и Вычисления

Запись транспортной задачи в окне редактора Mathcad имеет следующий вид:

Листинг.

Тарифы – стоимости перевозки единицы продукта от первого поставщика (A_1) потребителям 1, 2, 3 (B_1, B_2, B_3):

c11 = 45 c12 = 46 c13 := 76

Тарифы – стоимости перевозки единицы продукта от второго поставщика (A_2) потребителям 1, 2, 3 (B_1, B_2, B_3):

c21 := 43 c22 := 57 c23 := 54

Количество продукта у 1-го и 2-го поставщика:

a1 := 400 a2 := 500

Потребности потребителей 1-го, 2-го, 3-го:

$b1 := 400 \quad b2 := 300 \quad b3 := 200$

Общие затраты на перевозку (целевая функция):

$$f(x11, x12, x13, x21, x22, x23) := c11 \cdot x11 + c12 \cdot x12 + c13 \cdot x13 + c21 \cdot x21 + c22 \cdot x22 + c23 \cdot x23$$

Начальные приближения:

$x11 := 400 \; x12 := 0 \; x13 := 0 \; x21 := 0 \; x22 := 300 \; x23 := 200$

Блок ограничений:

Given

Условие равенства вывозимого продукта запасам у 1 и 2 поставщиков:

$x11 + x12 + x13 = a1$

$x21 + x22 + x23 = a2$

Условие равенства доставленного продукта потребностям 1, 2 и 3 потребителей:

$x11 + x21 = b1$

$x12 + x22 = b2$

$x13 + x23 = b3$

Физический смысл транспортной задачи:

$x11 \geq 0 \quad x12 \geq 0 \quad x13 \geq 0$

$x21 \geq 0 \quad x22 \geq 0 \quad x23 \geq 0$

Минимизация:

$c := \mathrm{Minimize}(f, x11, x12, x13, x21, x22, x23)$

$x11 := c_0 \quad x12 := c_1 \quad x13 := c_2 \quad x21 := c_3 \quad x22 := c_4 \quad x23 := c_5$

Минимальные затраты на перевозку:

$D := f(c_0, c_1, c_2, c_3, c_4, c_5)$

$D = 42\,000$

Имеющихся текстовых пояснений в документах Mathcad достаточно, чтобы можно было легко решить и другие подобные задачи. Для решения достаточно изменить числовые значения в данных документах.

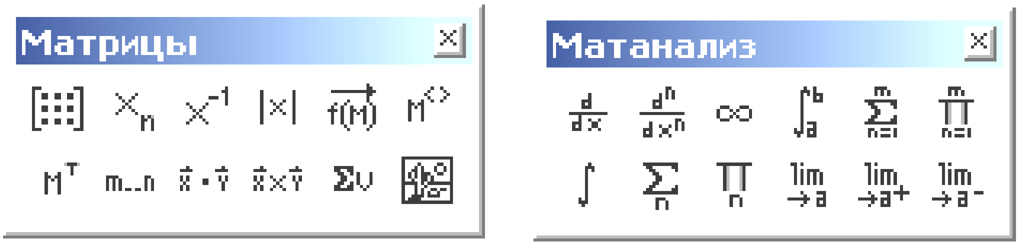


Рис. 7. Панели Матрицы и Матанализ

При решении транспортных задач с большим числом переменных целесообразно использовать матрицы. В них с помощью палитры матричных операций, показанной на рис. 7, размещаются тарифы перевозок, начальные приближения для объемов перевозок, запасы поставщиков и потребности потребителей. Целевая функция записывается с помощью шаблона суммы (панель **Матанализ**). В качестве примера рассмотрим задачу, исходные данные которой представлены в виде табл. 4.

Таблица 4

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2	3	5	4	30
A_2	3	2	4	1	40
A_3	4	3	2	6	20
Потребности	20	25	35	10	–

Решение закрытой транспортной задачи с тремя поставщиками и четырьмя потребителями имеет следующий вид.

Листинг.

Тарифы – стоимости перевозки единицы продукта от трех поставщиков (A_i) четырем потребителям (B_j):

$$c := \begin{pmatrix} 2 & 3 & 5 & 4 \\ 3 & 2 & 4 & 1 \\ 4 & 3 & 2 & 6 \end{pmatrix}$$

Количество продукта у поставщиков:

$$a := \begin{pmatrix} 30 \\ 40 \\ 20 \end{pmatrix}$$

Потребности потребителей:

$$b := \begin{pmatrix} 20 \\ 25 \\ 35 \\ 10 \end{pmatrix}$$

Общие затраты на перевозку (целевая функция):

$$f(x) := \sum_{i=0}^{2} \sum_{k=0}^{3} c_{i,k} \cdot x_{i,k}$$

Начальное приближение:

$$x := \begin{pmatrix} 20 & 10 & 0 & 0 \\ 0 & 15 & 25 & 0 \\ 0 & 0 & 10 & 10 \end{pmatrix}$$

Вспомогательная матрица:

$i := 0\,..2 \quad k := 0\,..\,3 \quad e_{k,j} := 1$

$$e := \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Given

Ограничения:

равенство вывозимого продукта запасам поставщиков:

$$(x \cdot e)^{\langle 0 \rangle} = a;$$

равенство вывозимого продукта равно потребностям потребителей

$$\left((e \cdot x)^{T}\right)^{\langle 0 \rangle} = b$$

Условие неотрицательности объема поставок:

$$x \geq 0$$

Минимизация:

y:= Minimize(f, x)

Ответ:

$$y = \begin{pmatrix} 20 & 0 & 10 & 0 \\ 0 & 25 & 5 & 10 \\ 0 & 0 & 20 & 0 \end{pmatrix}$$

Минимальные затраты на перевозку:

$$f(y) = 210$$

Для перехода к решению открытых транспортных задач с объемом запасов у поставщиков, не равным потребностям потребителей, в документе делаются соответствующие изменения, как показано на примерах в приложении.

Пример 2.11. Для строительства четырех дорог используется гравий из трех карьеров. Запасы гравия в каждом из карьеров соответственно равны 120, 280 и 160 усл. ед. Потребности в гравии для строительства каждой из дорог соответственно равны 130, 220, 60 и 70 усл. ед. Известны также тарифы перевозок 1 усл. ед. гравия из каждого карьера к каждой строящейся дороге, которые задаются матрицей. Составить такой план перевозок гравия, при котором потребности в нем каждой из строящихся дорог были бы удовлетворены при наименьшей общей стоимости перевозок.

Решение. Исходные данные задачи сведем в таблицу (табл. 5).

Таблица 5

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	1	7	9	5	120
A_2	4	2	6	8	280
A_3	3	8	1	2	160
Потребности	130	220	60	70	–

Как видно из табл. 5, запасы гравия в карьерах A_1, A_2 и A_3 (120 + 280 + 160 = 560) больше, чем потребности в нем на строящихся дорогах (130 + 220 + 60 + 70 = 480). Следовательно, модель исходной транспортной задачи является открытой.

Решение открытой транспортной задачи с тремя поставщиками и четырьмя потребителями имеет следующий вид.

Листинг.

Тарифы – стоимости перевозки единицы продукта от трех поставщиков (A_i) четырем потребителям (B_j):

$$c := \begin{pmatrix} 1 & 7 & 9 & 5 \\ 4 & 2 & 6 & 8 \\ 3 & 8 & 1 & 2 \end{pmatrix}$$

Количество продукта у поставщиков:

$$a := \begin{pmatrix} 120 \\ 280 \\ 160 \end{pmatrix}$$

Потребности потребителей:

$$b := \begin{pmatrix} 130 \\ 220 \\ 60 \\ 70 \end{pmatrix}$$

Общие затраты на перевозку (целевая функция):

$$f(x) := \sum_{i=0}^{2} \sum_{k=0}^{3} c_{i,k} \cdot x_{i,k}$$

Начальное приближение:

$$x := \begin{pmatrix} 100 & 0 & 0 & 0 \\ 0 & 200 & 0 & 0 \\ 0 & 0 & 40 & 20 \end{pmatrix}$$

Вспомогательная матрица:

$$i := 0..2 \quad k := 0..3 \quad e_{k,j} := 1$$

$$e := \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Given

Ограничения:

неравенство вывозимого продукта запасам поставщиков:

$$(x \cdot e)^{<0>} \le a;$$

неравенство вывозимого продукта потребностям потребителей

$$((e \cdot x)^{T})^{<0>} \ge b$$

Условие неотрицательности объема поставок:

$$x \ge 0$$

Минимизация:

$$y := \text{Minimize}(f, x)$$

Ответ:

$$y = \begin{pmatrix} 120 & 0 & 0 & 0 \\ 0 & 220 & 0 & 0 \\ 10 & 0 & 60 & 70 \end{pmatrix}$$

Минимальные затраты на перевозку:

$$f(y) = 790$$

Как видно из листинга программы, исходная задача имеет оптимальный план, при котором остается неиспользованным 60 усл. ед. гравия во втором карьере и 20 усл. ед. – в третьем, а общая минимальная стоимость перевозок составляет 790 усл. ед.

2.4. Решение задач интерполирования и регрессионного анализа

Одной из важнейших задач в процессе математического моделирования является вычисление значений функций, входящих в математическое описание модели. Для сложных моделей подобные вычисления могут быть трудоемкими даже при использовании ПК.

Используемые в математических моделях функции задаются как аналитическим способом, так и табличным, при котором функции известны только при дискретных значениях аргументов.

Пусть известные значения некоторой функции $f(x)$ образуют табл. 6.

Таблица 6

x	x_0	x_1	…	x_n
$f(x)$	y_0	y_1	…	y_n

При этом требуется получить значение функции $f(x)$ для такого значения аргумента x, которое входит в отрезок $[x_0, x_n]$, но не совпадает ни с одним из значений x_i ($i = 0, 1, ..., n$).

Классический подход к решению задачи построения аппроксимирующей (от латинского *approximo* – приближаюсь) функции $\varphi(x)$ основывается на требовании строгого совпадения значений $f(x)$ и $\varphi(x)$ в точках x_i. В этом случае нахождение приближенной функции называют интерполяцией, а точки x_i – узлами интерполяции.

2.4.1. Линейная интерполяция. Самый простой вид интерполяции – линейная, которая представляет искомую зависимость $\varphi(x)$ в виде ломаной линии. Интерполирующая функция $\varphi(x)$ состоит из отрезков прямых, соединяющих точки (рис. 8).

Для построения линейной интерполяции служит встроенная функция *linterp*.

Пример 2.12. Пусть имеются точки $A_1(1; 2)$, $A_2(2; 2.5)$, $A_3(3; 4.5)$, $A_4(4; 5)$, $A_5(5; 6.8)$. Надо вычислить значения этой функции в требуемых точках, например $\varphi(2.5)$ и $\varphi(4.58)$.

Листинг программы по решению этой задачи следующий:

$$x := (1\ 2\ 3\ 4\ 5)^T$$

$$y := (2\ 2.5\ 4.5\ 5\ 6.8)^T$$

$$f(t) := \mathrm{linterp}(x, y, t)$$

Как видно из листинга, чтобы осуществить линейную интерполяцию, надо выполнить следующие действия:

- ввести векторы данных x и y (первые две строки листинга);
- определить функцию *linterp*(x, y, t);
- вычислить значения этой функции в требуемых точках $f(2.5) = 3.5$ и $f(4.58) = 6.044$;

- построить график, как показано на рис. 8.

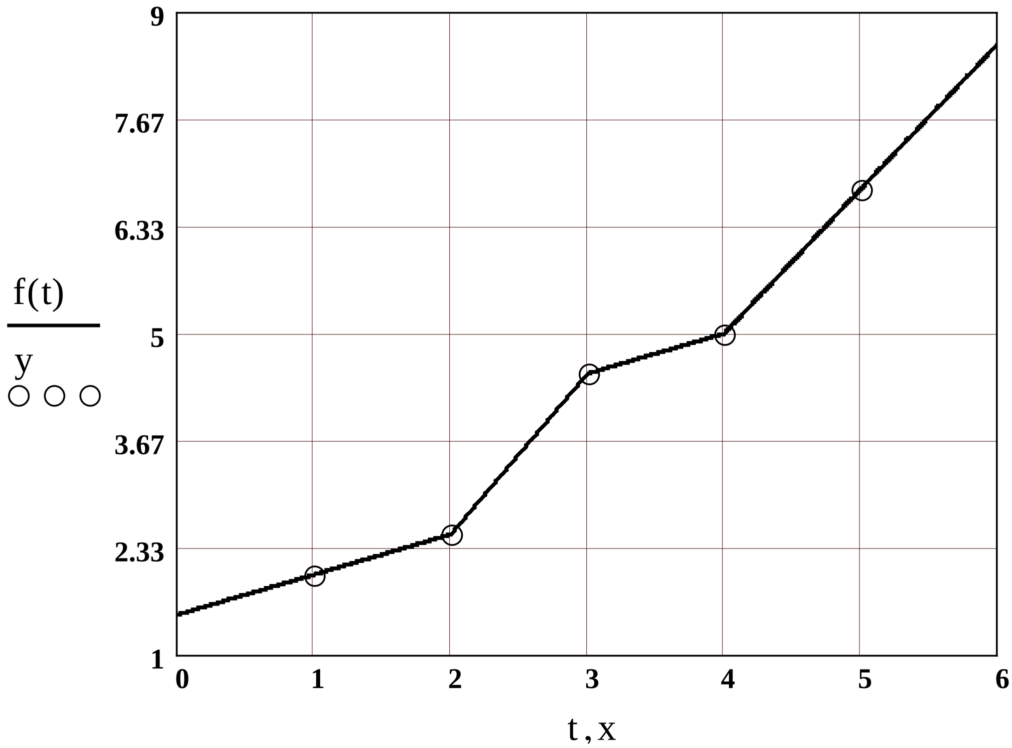


Рис. 8. Линейная интерполяция

2.4.2. Сплайн-интерполяция. В большинстве практических приложений желательно соединить экспериментальные точки не ломаной линией, а гладкой кривой. Лучше всего для этих целей подходит интерполяция сплайнами, т. е. отрезками парабол. Для построения сплайн-интерполяции в Mathcad служит встроенная функция *interp*(*s*, *x*, *y*, *t*) – функция, аппроксимирующая данные векторов *x* и *y* сплайнами, где *s* – вектор вторых производных, созданный одной из сопутствующих функций *cspline*, *pspline* или *lspline*; *x* – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания; *y* – вектор действительных данных значений того же размера; *t* – значение аргумента, при котором вычисляется интерполирующая функция.

Сплайн-интерполяция в MathCAD реализована чуть сложнее линейной. Перед применением функции *interp* необходимо предварительно определить первый из ее аргументов – векторную переменную *s*. Делается это при помощи одной из трех встроенных функций тех же аргументов (*x*, *y*):

- *lspline*(*x*, *y*) – вектор значений коэффициентов линейного сплайна;
- *pspline*(*x*, *y*) – вектор значений коэффициентов квадратичного сплайна;
- *cspline*(*x*, *y*) – вектор значений коэффициентов кубического сплайна.

Выбор конкретной функции сплайновых коэффициентов влияет на интерполяцию вблизи конечных точек интервала.

Смысл сплайн-интерполяции заключается в том, что в промежутках между точками осуществляется аппроксимация в виде зависимости $f(t) = a*t^3 + b*t^2 + c*t + d$. Коэффициенты a, b, c, d рассчитываются независимо для каждого промежутка, исходя из значений y_i в соседних точках. Этот процесс скрыт от пользователя, поскольку смысл задачи интерполяции состоит в выдаче значения $f(t)$ в любой точке t.

Чтобы подчеркнуть различия, соответствующие разным вспомогательным функциям *cspline*, *pspline*, *lspline*, покажем результат действия листинга кубического сплайна (рис. 9):

$$x := (0\ 1\ 2\ 3\ 4\ 5\ 6)^T$$

$$y := (4.1\ 2.4\ 3\ 4.3\ 3.6\ 5.2\ 5.9)^T$$

$$s := \mathrm{cspline}(x, y)$$

$$f(t) := \mathrm{interp}(s, x, y, t)$$

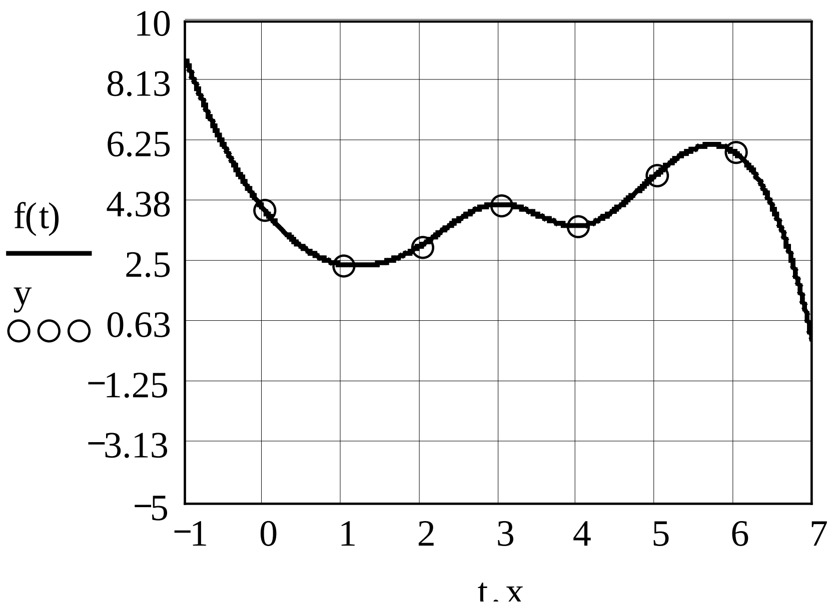


Рис. 9. Кубическая сплайн-интерполяция

При замене функции *cspline* в предпоследней строке предыдущего листинга на линейную *lspline* (рис. 10) :

$$x := (0\ 1\ 2\ 3\ 4\ 5\ 6)^T$$

$$y := (4.1\ 2.4\ 3\ 4.3\ 3.6\ 5.2\ 5.9)^T$$

$$s := \text{lspline}(x, y)$$

$$f(t) := \text{interp}(s, x, y, t)$$

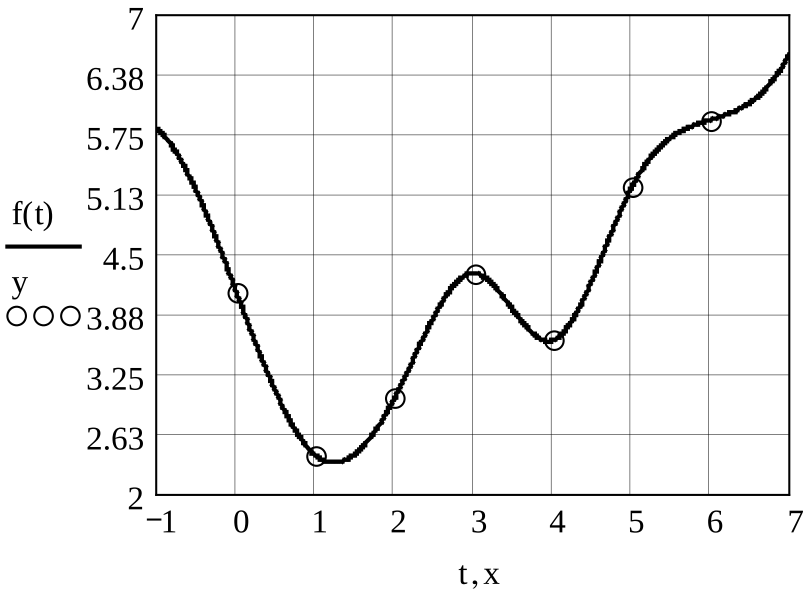


Рис. 10. Линейная сплайн-интерполяция

Как видно, выбор вспомогательных функций существенно влияет на поведение $f(t)$ вблизи граничных точек рассматриваемого интервала (0,6) и особенно разительно меняет результат экстраполяции данных за его пределами ($0 > t$, $t > 6$).

2.4.3. Линейная регрессия. **С**овпадение значений в узлах иногда может вовсе не означать совпадения характеров поведения исходной и интерполирующей функций. Требование неукоснительного совпадения значений в узлах выглядит тем более не оправданным, если значения функции $f(x)$ получены в результате измерений и являются сомнительными.

Поставим сейчас задачу так, чтобы с самого начала обязательно учитывался характер исходной функции: найти функцию заданного вида $y = \varphi(x)$, которая в точках $x_1, x_2, \ldots, x_n$ принимает значения как можно более близкие к табличным значениям $y_1, y_2, \ldots, y_n$.

Практически вид приближенной функции $\varphi(x)$ можно определить следующим образом. По табл. 6 строится точечный график функции $f(x)$, а затем проводится плавная кривая, по возможности, наилучшим образом отражающая характер расположения точек. По полученной таким образом кривой устанавливается вид приближающей функции.

Следует заметить, строгая функциональная зависимость для экспериментально полученной табл. 7 наблюдается редко, ибо каждая из участвующих в ней величин $y = \varphi(x)$ (ее называют эмпирической зависимостью или уравнением регрессии y на x) интересна тем, что позволяет находить значение функции $f(x)$ для нетабличных значений x, «сглаживая» результаты измерений величины y. Воспользовавшись метрикой евклидова пространства, приходим к требованию что сумма

$$s = (f(x_1) - \varphi(x_1))^2 + (f(x_2) - \varphi(x_2))^2 + \ldots + (f(x_n) - \varphi(x_n))^2$$

должна быть наименьшей.

Итак, задача приближения функции f теперь формулируется следующим образом: для функции f, заданной табл. 7, найти функцию $\varphi(x)$ определенного вида так, чтобы сумма квадратов была наименьшей. Эта задача носит название аппроксимация функции методом наименьших квадратов (МНК). Не вдаваясь в его тонкости (см. [7]), отметим главное – этот метод позволяет так подобрать некоторые параметры аппроксимирующей зависимости, что она описывает исходные данные с минимальной среднеквадратической погрешностью. Система Mathcad хороша тем, что при работе с ней вполне достаточно помнить лишь об этом фундаментальном определении МНК, а все остальное Mathcad автоматически проделает сама, используя соответствующие процедуры.

Рассмотрим простую задачу на линейную регрессию.

Пример 2.13. Пусть имеется ряд точек $A_1(1; 2)$, $A_2(2; 2.9)$, $A_3(3; 4.05)$, $A_4(4; 5)$, $A_5(5; 6.1)$. Надо подобрать коэффициенты $a0$ и $a1$ линейной зависимости $y(x) = a0 + a1 \cdot x$ такими, чтобы прямая прошла в «облаке» точек с наименьшим общим среднеквадратическим отклонением от них. Для решения этой задачи (линейной регрессии) в Mathcad имеются два дублирующих друг друга способа. Правила их применения представлены в следующих листингах.

Листинг 1. Линейная регрессия

$$x := (1\ 2\ 3\ 4\ 5)^T$$

$$y := (2\ 2.9\ 4.05\ 5\ 6.1)^T$$

$$a0 := \text{intercept}(x, y)$$
$$a1 := \text{slope}(x, y)$$
$$a0 = 0.92$$
$$a1 := 1.03$$
$$z(x) := a0 + a1 \cdot x$$

Листинг 2. Линейная регрессия

$$x := (1\ 2\ 3\ 4\ 5)^T$$
$$y := (2\ 2.9\ 4.05\ 5\ 6.1)^T$$
$$\text{Line}(x,y) = \begin{pmatrix} 0.92 \\ 1.03 \end{pmatrix}$$
$$z(x) := \text{Line}(x,y)_0 + \text{Line}(x,y)_1 \cdot x$$

Представим эту задачу графическим способом, задав полученную эмпирическую зависимость $z(x)$ в виде линии, а координаты вектора y соответствующими точками (рис. 11).

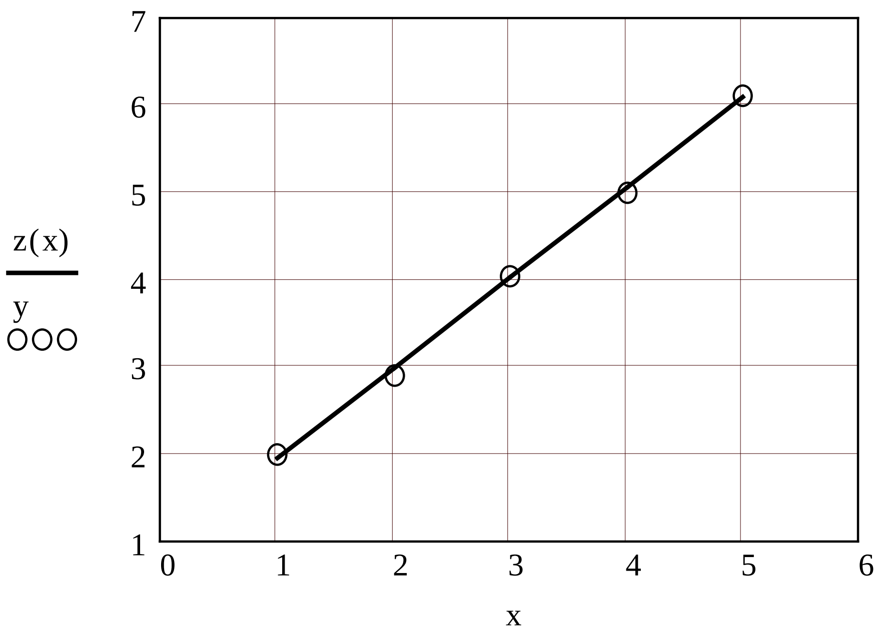


Рис. 11. Графики $z(x)$ и точки вектора y

Анализируя представленные графики, можно утверждать о правильно выбранной математической модели (уравнения регрессии) исследуемого процесса.

2.4.4. Полиномиальная регрессия. Еще один важный вид нелинейной регрессии – это полиномиальная регрессия, при которой аппроксимирующая функция есть полином заданной степени *n*:

$$P_n(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n.$$

При $n = 1$ полином является прямой линией, при $n = 2$ – параболой, при $n = 3$ – кубической параболой и т. д. Как правило, на практике применяются $n < 5$.

Для построения регрессии полиномом *n*-й степени необходимо наличие, по крайней мере ($n + 1$) точек данных.

Полиномы могут описывать весьма сложные и замысловатые зависимости, поэтому они и широко применяются в задачах аппроксимации. В Mathcad полиномиальная регрессия осуществляется комбинацией встроенной функции *regress* и полиномиальной интерполяции *interp*.

Для построения полиномиальной регрессии после функции *regress* необходимо использовать функцию *interp*.

Пример 2.14. Пусть имеется ряд точек полученных на основе эксперимента исследуемого процесса $A_0(0; 4.1)$, $A_1(1; 2.4)$, $A_2(2; 3)$, $A_3(3; 4.3)$, $A_4(4; 3.6)$, $A_5(5; 5.2)$, $A_6(6; 5.9)$. Построить полиномиальную регрессию 4-й степени.

Листинг программы в окне редактора Mathcad по решению этой задачи имеет следующий вид.

$$x := (0\ 1\ 2\ 3\ 4\ 5\ 6)^T$$

$$y := (4.1\ 2.4\ 3\ 4.3\ 3.6\ 5.2\ 5.9)^T$$

$$k := 4$$

$$s := \mathrm{regress}(x, y, k)$$

$$f(t) := \mathrm{interp}(s, x, y, t)$$

График полученной полиномиальной регрессии и координаты точек эксперимента представлены на рис. 12.

Как показывает данный пример, встроенные функции системы Mathcad, специализированные на научно-технические и статистические расчеты, обеспечивают микропрограммное (т. е. без ввода внешних программ) вычисление важнейших характеристик двумерных массивов $x_i y_i$, а также проведение корреляционного и линейного регрессионного анализа.

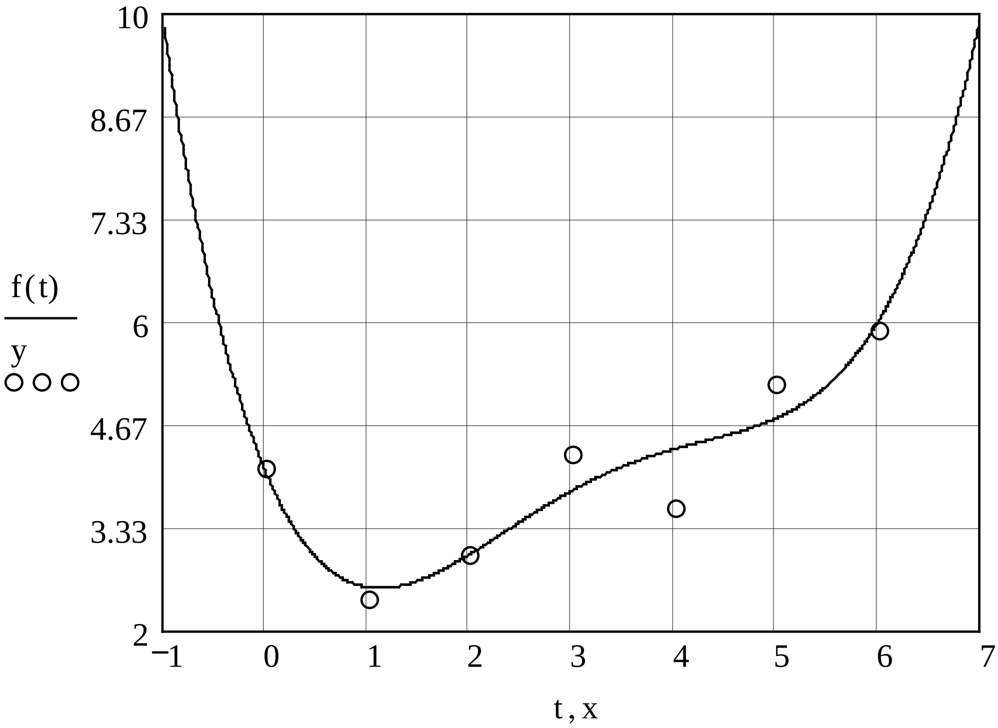


Рис. 12. Регрессия полиномом 4-й степени

Таким образом, чрезвычайно простыми и однообразными средствами Mathcad позволяет решать многочисленные задачи регрессионного анализа и аппроксимации.

2.4.5. Регрессия специального вида. Кроме рассмотренных, в Mathcad встроено еще несколько видов трехпараметрической регрессии. Их реализация несколько отличается от приведенных выше вариантов регрессии тем, что для них, помимо массива данных, требуется задать некоторые начальные значения коэффициентов *a, b, c*. Использовать соответствующий вид регрессии можно, если заранее известно, какой зависимостью описывается массив данных. Когда тип регрессии плохо отражает массив данных, то ее результат часто бывает неудовлетворительным и даже сильно различающимся в зависимости от выбора начальных значений.

Каждая из регрессионных функций выдает вектор уточненных параметров *a, b, c* эмпирических зависимостей:

- *expfit*(*x*, *y*, *g*) – регрессия экспонентой $f(x) = a \cdot e^{bx} + c$;
- *igsfit*(*x*, *y*, *g*) – регрессия функцией $f(x) = a / (1 + b \cdot e^{-cx})$;

- *sinfit* (x, y, g) – регрессия синусоидой $f(x) = a \cdot sin(x + b) + c$;
- *pwfit*(x, y, g) – регрессия степенной функцией $f(x) = a \cdot x^b + c$;
- *logfit* (x, y, g) – регрессия натуральной логарифмической функцией $f(x) = a \cdot ln(x + b) + c$;
- *infit*(x,y) – регрессия двухпараметрической логарифмической функцией $f(x) = a \cdot ln(x) + b$;

x – вектор действительных данных аргумента;

y – вектор действительных значений того же размера;

g – вектор, задающий начальные значения a, b, c.

Многие задачи регрессии данных различными двухпараметрическими зависимостями $f(x)$ можно свести к более надежной, с вычислительной точки зрения, линейной регрессии. Делается это с помощью соответствующей замены переменных.

Пример 2.15. Пусть имеется ряд экспериментальных точек $A_0(0; 4.1)$, $A_1(1; 2.4)$, $A_2(2; 3)$, $A_3(3; 4.3)$, $A_4(4; 3.6)$, $A_5(5; 5.2)$ $A_6(6; 5.9)$. Надо определить коэффициенты a, b и c эмпирической зависимости $f(x) = a \cdot e^{bx} + c$ и построить график полученной зависимости.

Листинг программы в окне редактора Mathcad по решению этой задачи имеет следующий вид:

Ввод координат векторов

$$x := (0\ 1\ 2\ 3\ 4\ 5\ 6)^T$$

$$y := (4.1\ 2.4\ 3\ 4.3\ 3.6\ 5.2\ 5.9)^T$$

Ввод единичного вектора

$$g = (1\ 1\ 1)^T$$

Определение параметров экспоненциальной регрессии

$$c := \mathrm{expfit}(x, y, g)$$

Координаты найденного вектора c

$$c = \begin{pmatrix} 0.111 \\ 0.544 \\ 3.099 \end{pmatrix}$$

Уравнение экспоненциальной регрессии

$$f(t) := c_0 \cdot \exp(c_1 \cdot t) + c_2$$

График полученной эмпирической зависимости (экспоненциальной регрессии) и координаты точек эксперимента исследуемого процесса представлены на рис. 13.

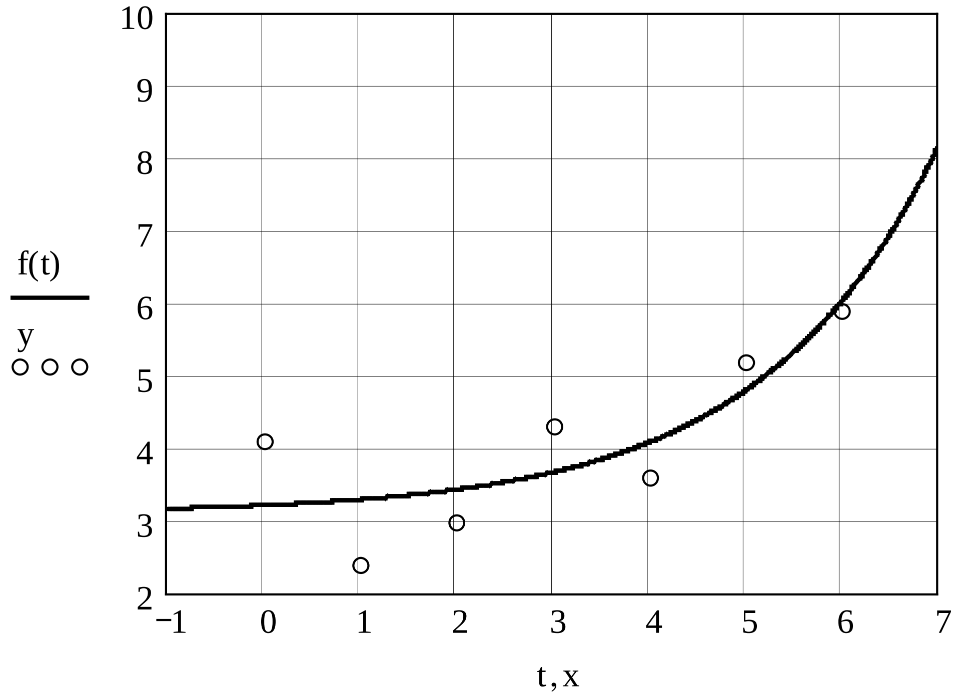


Рис. 13. Экспоненциальная регрессия

Правильность выбора начальных значений можно оценить по результату регрессии – если функция, выданная Mathcad, хорошо приближает зависимость *f*(*x*), значит они были подобраны удачно.

2.5. Решение финансово-экономических задач

В последние версии системы Mathcad введен пакет функций для выполнения финансово-экономических расчетов. Любая из финансово-экономических функций предназначена для вычисления значения одного из параметров названных расчетов по заданным значениям других параметров. К основным параметрам относятся:

- *pv* – сумма вклада в настоящее время;
- *fv* – сумма вклада в будущем;
- *rate* – процент начислений;
- *crate* – фиксированный процент начислений на период;
- *nper* – число периодов;
- *cnper* – число составных периодов;
- *start* – начало периода;

- *end* – конец периода.

Для решения задач, используемых в финансово-экономической деятельности предприятий и банковских услуг населению, система Mathcad имеет следующие функции:

- *cnper*(*rate*, *pv*, *fv*) – число составных периодов для получения будущего значения вклада при заданных текущем значении вклада и проценте начислений;
- *crate*(*nper*, *pv*, *fv*) – фиксированный процент начислений на период, необходимый для прироста вклада от настоящего значения до будущего значения при заданном числе составных периодов;
- *cumint*(*rate*, *nper*, *pv*, *start*, *end*) – совокупный процент, который платится по займу;
- *cumprn*(*rate*, *nper*, *pv*, *stall*, *end*) – совокупная сумма, которая платится по займу;
- *eff(rate, nper)* – эффективная ежегодная сумма, которая платится по займу;
- *fv*(*rate*, *nper*, *pmt*) – будущее значение вклада;
- *fvadj*(*prin*, *v*) – будущее значение вклада при ряде составных процентных ставок;
- *fvc*(*rate*, *v*) – будущее значение вклада при фиксированных процентных ставках;
- *ipmt*(*rate*, *per*, *nper*, *pv*) – процентная ставка на период;
- *irr*(*v*) – внутренний процент возврата для ряда денежных вливаний;
- *mirr*(*v*, *finrate*, *reinrate*) – модифицированная внутренняя ставка возврата для ряда денежных взносов;
- *nom*(*rate*, *npег*) – номинальная процентная ставка;
- *npег*(*rate*, *pmt*, *pv*) – число периодов;
- *npv*(*rate*, *v*) – нынешнее значение вклада;
- *pmt*(*rate*, *npег*, *pv*) – плата за период займа;
- *ppmt*(*rate*, *per*, *npег*, *pv*) – плата на общую сумму данного периода;
- *pv*(*rate*, *npег*, *pmt*) – вклад, который должен быть внесен в настоящее время при устойчивой плате и фиксированном проценте ставки;
- *rate*(*nper*, *pmt*, *pv*) – процентная ставка за период ренты.

К сожалению, определения финансово-экономических величин и расчетные соотношения между ними заметно различаются у нас и за рубежом [2], что чревато серьезными ошибками в выполнении таких расчетов.

Рассмотрим на практических примерах использование встроенных функций системы Mathcad при выполнении финансово-экономических расчетов:

Пример 2.16. Сколько должно пройти годовых периодов, чтобы при проценте годовых 10% начальный вклад в 1000 усл. ед. возрос до конечного значения в 2000 усл. ед.?

Листинг.

$$rate := 10\cdot\% \quad pv := 1000 \quad fv := 2000$$

$$cnper(rate, pv, fv) = 7.273$$

Пример 2.17. Какой процент годовых нужен, чтобы через семь годовых периодов вклад в 1000 усл. ед. вырос до 2000 усл. ед.?

Листинг.

$$nper := 7 \quad pv := 1000 \quad fv := 2000$$

$$crate(nper, pv, fv) = 0.104$$

Пример 2.18. Некто приобрел в рассрочку дом, стоимостью 150 000 усл. ед и намерен выкупить его за 30 лет. Сколько ему надо внести в первый год при проценте годовых 8%, чтобы с учетом накоплений вклада дом был выкуплен.

Листинг.

$$rate := 8.3\cdot\% \quad pv := 130\,000 \quad nper := 30\cdot 12 \quad start := 1 \quad end := 12$$

$$cumint\left(\frac{rate}{12}, nper, pv, start, end\right) = 12\,403.768$$

Пример 2.19. Каковы должны быть ежемесячные отчисления для выкупа дома при тех же условиях?

Листинг.

$$cumpm\left(\frac{rate}{12}, nper, pv, start, end\right) = 1180.354$$

Пример 2.20. Процент накопления равен 10%, а число периодов накопления – 24. Чему равен эффективный процент накопления?

Листинг.

$$rate := 10 \cdot \% \quad nper := 24$$

$$\frac{eff(rate, nper)}{\%} = 10.494$$

Пример 2.21. Вкладчик решил вкладывать в банк ежемесячно по 100 усл. ед. Сколько долларов окажется на вкладе через год, если процент годовых составляет 6%?

Листинг.

$$fv\left(\frac{0.06}{12}, 12, -100\right) = 1233.556$$

Пример 2.22. Сколько денег будет на вкладе при тех же условиях, если сделан начальный вклад в 1000 усл. ед.?

Листинг.

$$fv\left(\frac{0.06}{12}, 12, -100, -1000\right) = 2295.234$$

Пример 2.23. Вкладчик решил сделать в течение трех лет вклады в 1000, 1200 и 1500 усл. ед. Каково будет будущее значение вклада при проценте годовых 8%?

Листинг.

$$rate := 8 \cdot \% \quad V := \begin{pmatrix} 1000 \\ 1200 \\ 1500 \end{pmatrix}$$

$$fvc(rate, V) = 3962.4$$

Пример 2.24. Вкладчик положил в банк начальную сумму 1000 усл. ед. на три года. Если проценты годовых равны 6%, 7% и 8%, то какая сумма будет на вкладе через три года?

Листинг.

$$prin := 1000 \quad sheed := \begin{pmatrix} 0.06 \\ 0.07 \\ 0.08 \end{pmatrix}$$

$$\text{fvadj(prin, sheed)} = 1224.936$$

Пример 2.25. Вкладчик внес в конце первого периода сумму в 10 000 усл. ед., а затем снимал в конце последующих периодов суммы в 3000 усл. ед., 4200 усл. ед. и 6800 усл. ед. Если процент равен 10%, то какая сумма окажется на данный момент?

Листинг.

$$\text{prate} := 10\cdot\% \quad \text{CF} := (0\ -10\,000\ 3000\ 4200\ 6800)^{\mathrm{T}}$$

$$\text{npv(prate, CF)} = 1080.403$$

Пример 2.26. Вкладчик хочет получать 200 усл. ед. в конце каждого месяца при проценте годовых 7,5% на протяжении 15 лет. Какой начальный вклад для этого нужно внести?

Листинг.

$$\text{pv}\left(\frac{0.075}{12}, 12\cdot 15, 200\right) = -21\,574.685$$

Пример 2.27. Вкладчик внес вклад в 10 000 усл. ед. и намерен погашать его на протяжении 48 периодов суммами в размере 250 усл. ед. Какой процент начислений за каждый период должен обеспечить банк, если в конце всех периодов на счету должна остаться сумма в 1000 усл. ед.?

Листинг.

$$\text{pv} := 10\,000 \quad \text{nper} := 48 \quad \text{fv} := 1000 \quad \text{pmt} := -250$$

$$\text{r} := \text{rate(nper, pmt, pv, fv)}$$

$$\text{r} = 0.004$$

$$\text{ry} := \text{r}\cdot 12$$

$$\text{ry} = 0.052$$

Пример 2.28. Вкладчик имеет в банке сумму 5000 усл. ед. Процент годовых начислений равен 8.75. В конце каждого из 10 периодов он намерен снимать некоторую сумму до полной ликвидации вклада. Какую сумму он должен снимать?

Листинг.

$$\text{Nper} := 10 \quad \text{ratel} := 8.75\cdot\% \quad \text{pv} := 15\,000 \quad \text{fv} := 0$$

$$\text{pmt(rate1, nper, pv, fv)} = -2311.64$$

Приведенные расчеты можно использовать на практике только после тщательной проверки. Зачастую гораздо проще составить документы, в которых такие расчеты выполняются по принятым у нас формулам и алгоритмам.

2.6. Задача межотраслевого баланса

Простейшая макроэкономическая модель экономики предполагает наличие в ней двух секторов – производителей и потребителей. От производителей идет поток товаров и услуг, от потребителей – поток денег, при этом для проведения расчетов объем товаров и услуг удобно выражать в денежных единицах. Всю производимую продукцию (товары и услуги) можно разделить на две части: промежуточный продукт и конечный.

Промежуточный продукт – эта та часть совокупного продукта, которой производители обмениваются между собой или используют для собственных нужд, например, электроэнергия. Конечный продукт – это вся продукция, предназначенная для потребителя. Если в выпуске конечного продукта участвуют несколько фирм-производителей, то взаимодействие между ними описывается с помощью модели межотраслевого баланса, или модели «затраты – выпуск». На производство единицы своей продукции каждая фирма несет определенные затраты, связанные, например, с приобретением оборудования, материалов, запасных частей, оплатой услуг, оплатой труда и т. п., которые называются прямыми затратами. В модели в простейшем случае предполагаем линейность связей. Это означает, что увеличение выпуска продукции вдвое повлечет за собой двукратное увеличение затрат.

Пусть имеется n отраслей производства. Взаимосвязь между ними можно описать с помощью матрицы коэффициентов прямых затрат размера $n \times n$, каждый элемент a_{ij} ее означает количество продукции в стоимостном выражении, которое отрасль i поставляет отрасли j для производства единицы продукции:

$$\mathrm{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}.$$

Векторы-столбцы совокупного продукта X и конечного продукта Y можно определить соответственно как

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} \text{ и } Y = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix},$$

где x_i и y_i – совокупный и конечный продукты отрасли i.

Тогда уравнение Леонтьева описывает связь между матрицей коэффициентов прямых затрат, совокупным и конечным продуктом:

$$X = A - X + Y.$$

Для определения конечного продукта каждой отрасли при известных объемах совокупного продукта и матрице коэффициентов прямых затрат используем соотношение

$$Y = X - A - X,$$

а для определения совокупного продукта по известному конечному продукту (конечному спросу) – соотношение

$$X = (I - A)^{-1} - Y,$$

где I – единичная матрица, в которой $a_{ii} = 1$, $a_{ij} = 0$.

Задача. Пусть имеются две фирмы, производящие некоторую продукцию. Совокупный продукт, выпускаемый фирмой 1 за год составляет в стоимостных объемах 400 усл. ед. Для этого производства используется продукция собственных подразделений фирмы 1 в объеме 40 усл. ед. и закупленная у фирмы 2 в объеме 320 усл. ед. Фирма 2 выпускает продукцию на сумму 500 усл. ед., при этом поставки собственных подразделений определяются суммой 100 усл. ед., закупки у фирмы 1 – суммой 200 усл. ед Определить возможный объем выпуска для каждой фирмы, исходя из имеющихся денежных ресурсов.

Листинг программы в окне редактора Mathcad по решению этой задачи имеет следующий вид:

Origin = 1

Совокупный продукт фирмы 1:

$$x_1 := 400$$

Совокупный продукт фирмы 2:

$$x_2 := 500$$

Собственная продукция фирмы 1:

$$a_1 := 40$$

Собственная продукция фирмы 2:

$$a_2 := 100$$

Закупки у фирмы 2

$$b_1 := 320$$

Закупки у фирмы 1

$$b_1 := 200$$

Определяем матрицу коэффициентов собственных затрат на производство продукции фирмами 1 и 2:

$$a_{11} := \frac{a_1}{x_1} \quad a_{12} := \frac{b_2}{x_2} \quad a_{21} := \frac{b_1}{x_1} \quad a_{22} := \frac{a_{21}}{x_2}$$

Совокупный продукт фирм 1 и 2:

$$X := \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Конечный продукт фирм 1 и 2:

$$Y := X - A \cdot X$$

$$Y = \begin{pmatrix} 160 \\ 80 \end{pmatrix}$$

ПРИЛОЖЕНИЕ

Таблица

Закрытая задача	Открытые задачи	
Запасы равны потребностям $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$	Запасы больше потребностей $\sum_{i=1}^{m} a_i > \sum_{j=1}^{n} b_j$	Запасы меньше потребностей $\sum_{i=1}^{m} a_i < \sum_{j=1}^{n} b_j$
Запасы продукта у поставщиков 1, 2 и 3 $a \equiv \begin{bmatrix} 30 \\ 40 \\ 20 \end{bmatrix}$ в сумме равны потребностям потребителей $b \equiv \begin{bmatrix} 20 \\ 25 \\ 35 \\ 10 \end{bmatrix}$	Запасы продукта у поставщиков $a \equiv \begin{bmatrix} 35 \\ 40 \\ 25 \end{bmatrix}$ в сумме превышают потребности потребителей $b \equiv \begin{bmatrix} 20 \\ 25 \\ 35 \\ 10 \end{bmatrix}$	Запасы продукта у поставщиков $a \equiv \begin{bmatrix} 30 \\ 40 \\ 20 \end{bmatrix}$ в сумме меньше потребности потребителей $b \equiv \begin{bmatrix} 25 \\ 30 \\ 40 \\ 10 \end{bmatrix}$
Ограничения: равенство вывозимого продукта запасам у поставщиков $(x\cdot e)^{\langle 0\rangle} = a$; количество вывозимого продукта равно потребностям потребителей $((e\cdot x)^{T})^{\langle 0\rangle} = b$	Ограничения: количество вывозимого продукта меньше запасов у поставщиков $(x\cdot e)^{\langle 0\rangle} \le a$; количество вывозимого продукта больше потребностей потребителей $((e\cdot x)^{T})^{\langle 0\rangle} \ge b$	Ограничения: количество вывозимого продукта больше запасов у поставщиков $(x\cdot e)^{\langle 0\rangle} \ge a$; количество вывозимого продукта меньше потребностей потребителей $((e\cdot x)^{T})^{\langle 0\rangle} \le b$
Ответ: $y = \begin{bmatrix} 20 & 0 & 10 & 0 \\ 0 & 25 & 5 & 10 \\ 0 & 0 & 20 & 0 \end{bmatrix}$ Минимальные затраты на перевозку $f(y) = 210$	Ответ: $y = \begin{bmatrix} 20 & 5 & 0 & 0 \\ 0 & 20 & 10 & 10 \\ 0 & 0 & 25 & 0 \end{bmatrix}$ Минимальные затраты на перевозку $f(y) = 195$	Ответ: $y = \begin{bmatrix} 25 & 0 & 5 & 0 \\ 0 & 30 & 0 & 10 \\ 0 & 0 & 20 & 0 \end{bmatrix}$ Минимальные затраты на перевозку $f(y) = 185$

ЛИТЕРАТУРА

1. Леонтьев В. П. Новейшая энциклопедия персонального компьютера 2003. – 3-е изд., перераб. и доп. – М.: ОЛМА-ПРЕСС, 2002. – 847 с.

2. Дьяконов В. П. Mathcad: Учеб. курс. – СПб.: Питер, 2001. – 640 с.

3 Акулич И. Л. Математическое программирование в примерах и задачах. – М.: Высшая школа,1986. – 320 с.

4. Лащенко А. П. Информатика и компьютерная графика: Учеб.-метод. пособие для студентов спец. 1-75 01 01 «Лесное хозяйство». – Мн.: БГТУ, 2004. – 66 с.

5. Компьютерные информационные технологии: Учеб. пособие / Лащенко А. П. и др. – Мн.: БГТУ, 2004. – 70 с.

6. Кирьянов Д. В. Самоучитель Mathcad 2001. – СПб.: БХВ-Петербург, 2002. – 544 с.

7. Черняк А. А. и др. Математика для экономистов на базе Mathcad. – СПб.: БХВ-Петербург, 2003. – 496 с.

8. Демидович Б. П., Марон И. А. Основы вычислительной математики. – М.: Наука, 1966, – 684с.

9. Вержбицкий В. М. Основы численных методов. – М.: Высшая школа, 2002. – 840 с.

10. Практическое руководство по методам вычислений с приложением программ для персональных компьютеров: Учеб. пособие / Ракитин В. И. и др. – М.: Высшая школа, 1998. – 383 с.

СОДЕРЖАНИЕ

Лащенко Анатолий Павлович

Учебное издание

ИНЖЕНЕРНО-ЭКОНОМИЧЕСКИЕ ЗАДАЧИ НА БАЗЕ MATHCAD

Практикум

Редактор *Е. И. Гоман*

Подписано в печать 16.06.2006. Формат $60\times84^{1}/_{16}$.
Бумага офсетная. Гарнитура Таймс. Печать офсетная.
Усл. печ. л. 4,0. Уч.-изд. л. 4,0.
Тираж 300 экз. Заказ .

Учреждение образования
«Белорусский государственный технологический университет».
220050. Минск, Свердлова, 13а.
ЛИ № 02330/0133255 от 30.04.2004.

Отпечатано в лаборатории полиграфии учреждения образования
«Белорусский государственный технологический университет».
220050. Минск, Свердлова, 13.
ЛП № 02330/0056739 от 22.01.2004.