

Практическое занятие 2

Вычислительные блоки. Элементы программирования. Решение нелинейных уравнений

2.1. Вычислительные блоки Given – Find, Given – Minerr

Функции *Find* и *Minerr* предназначены для решений уравнений, неравенств и систем уравнений. Их можно ввести с клавиатуры либо вставить из списка функций. Значок диалогового окна вставки функции находится на стандартной панели. Обе функции предваряются ключевым словом *Given*.

Функция *Find* решает систему так, чтобы левые и правые части входящих в нее уравнений отличались на величину, не превышающую значения TOL (по умолчанию системная переменная TOL=0.001, однако ее всегда можно переопределить). Функция *Minerr* (MINimal ERRor) ищет приближенное решение путем минимизации среднеквадратичной погрешности. Результат вычислений осуществляется по стрелке символьных вычислений. Если вычисляется несколько переменных, они выводятся в матричном виде.

К особенностям работы с данными блоками отнесем следующее:

- вычислительный блок охватывает все горизонтальное пространство рабочего листа, поэтому всякое написание параллельно блоку посторонних выражений будет считаться встроенным в блок и повлечет за собой ошибки;
- не допускается применение в одном блоке обеих функций;
- найденные значения не присваиваются искомым переменным, поэтому при необходимости присвоение результата следует выполнить самостоятельно (ниже вычислительного блока).

Пример решения системы уравнений с двумя неизвестными:

$$\begin{array}{l} \text{Given} \\ 2x - 5y = 12 \\ 7x + 4y = -1 \\ \text{Find}(x, y) \rightarrow \begin{pmatrix} 1 \\ -2 \end{pmatrix} \end{array}$$

2.2. Элементы программирования

Помимо встроенных функций MathCAD позволяет создавать собственные функции, в которых используются операторы программирования, расположенные на панели Programming. Создаваемый программный блок отделяется вертикальной чертой, которая образуется кнопкой **Add line**. Операторы программирования имеют свою конструкцию,

которая появляется в шаблонном виде при вводе их при помощи панели Programming (не рекомендуется вводить операторы самостоятельно с клавиатуры во избежание ошибок).

while - ввод цикла с предпроверкой с двумя пустыми шаблонами
while ■
■
■
справа и внизу. В правый шаблон записывается булево выражение (переменная), в нижний - тело цикла. Если в теле цикла более одного оператора, добавляется дополнительный шаблон кнопкой **Add line**.

if - ввод условия. Имеет формат
■ if ■

otherwise - совместно с условием **if** - блок условия с альтернативой

for - блок для ввода цикла с параметром. В строке **for** записываются границы значений параметра (как правило, в виде ранжированной переменной), ниже шаблон предназначен для ввода тела цикла.

for ■ ∈ ■

■

break - досрочный выход из программы или цикла.

continue - продолжение цикла;

return - возврат из программы;

on error - инструкция по обработке ошибок.

Порядок составления программы:

- задается имя функции и в скобках перечисляются все переменные параметры, от которых зависит ее значение (символы локальных переменных должны быть точно такие же, как в программном блоке);
- ставится знак присвоения и вводится программный блок.

Для вычисления значения функции необходимо ниже программного блока ввести ее имя, а в скобках перечислить значения переменных.

Рассмотрим пример составления функции изгибающего момента, действующего в балке с приложенной поперечной силой. В качестве искомых параметров принимаются: длина балки **L**, расстояние от точки приложения силы до левой опоры **a**, величина поперечной силы **F** и расстояние до искомого сечения **x**. Для расчета примем силу положительной. Реакции рассчитывает сама программа. Поскольку момент имеет две функциональные зависимости (до точки приложения силы и после точки приложения), в блоке используется условие оператором **if**. Учтены оператором **return** два ограничения для ввода параметров **x** и **a**: они не должны быть отрицательными или превышать длину балки (причем **x** может быть равен 0 или **L**, для параметра **a** равенство 0 или **L** приведет к нулевым значениям момента, поскольку сила будет приложена в опоре). Знак **∨** является логическим “или” и находится на панели Boolean.

$$M(x, F, a, L) := \begin{cases} \frac{-F \cdot (L - a)}{L} \cdot x & \text{if } 0 \leq x \leq a \\ \frac{-F \cdot (L - a)}{L} \cdot x + F \cdot (x - a) & \text{if } a \leq x \leq L \\ \text{return "Error input! x out of L!"} & \text{if } x > L \vee x < 0 \\ \text{return "Error input! a out of L!"} & \text{if } a \leq 0 \vee a \geq L \end{cases}$$

Полученная функция позволит вычислить значения момента в любом сечении балки, если вместо x задать конкретное значение. Если задать x в виде ранжированной переменной от 0 до L , то можно получить значения момента во всех существующих точках x , а также построить эпюру. Например, при $x = 0.5$ м, $a = 1$ м, $F = 4$ кН, $L = 2$ м, получим следующее решение: $M(0.5, 4, 1, 2) = -1$

Если задать значения для x с шагом в 0.5 м, то получим значения момента в 5 сечениях (на рис. 3 построена эпюра момента по 5 значениям $x := 0, 0.5 \dots L$).

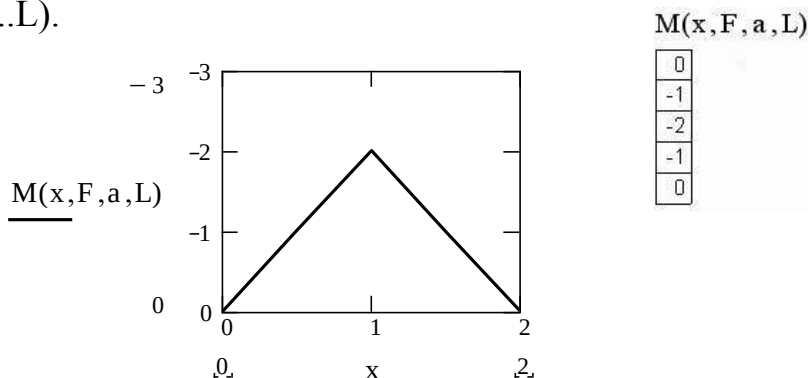


Рис. 3. Построение эпюры момента по функции-программе

2.3. Численное решение нелинейных уравнений

Многие уравнения не решаются аналитически, поэтому для их решения применяются численные методы. Это означает, что ищется приближенное решение $f(x) = 0$ с заданной погрешностью. Наиболее распространенными являются итерационные методы (т.е. методы последовательных приближений):

- простейший итерационный метод по начальному приближению корня x_0 ;
- метод половинного деления;
- метод секущих, или метод линейной интерполяции;
- метод касательных, или метод Ньютона.

Рассмотрим подробнее метод Ньютона. Пусть на отрезке $[a, b]$ содержится единственный корень $x = \alpha$ уравнения $f(x) = 0$, а функция имеет непрерывные производные $f'(x)$ и $f''(x)$, которые не обращаются в 0 на данном отрезке. Проведем касательную к функции в некоторой точке x_1 , y_1 внутри интервала $[a, b]$. Уравнение касательной: $y - y_1 = f'(x_1) \cdot (x - x_1)$

Поскольку $y_1 = f(x_1)$, получим для точки пересечения касательной с осью x ($y=0$):

$$x = x_1 - \frac{f(x_1)}{f'(x_1)} \quad (1)$$

Это означает, что существует такая окрестность точки $x = \alpha$, что последовательность

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})} \quad n = 1, 2, \dots \quad (2)$$

будет сходиться к $x = \alpha$.

Программа MathCAD предусматривает несколько способов и методов решения нелинейных уравнений. Поскольку все численные функции выдают результат с некоторой погрешностью, необходимо заранее задать ее величину. Погрешность регулируется системной переменной TOL, которая по умолчанию равна 0.001. Ее значение можно переопределить в рабочем листе, либо задать в диалоговом окне математических опций (меню *Tools* – пункт *Worksheet Options*).

Функция **root**($f(x), x, [a, b]$), где $f(x)$ – заданная функция, x – аргумент, a, b – интервал, в котором ищется корень функции $f(x)$. Функция **root** предполагает предварительное построение графика и определение интервала, внутри которого имеется одно пересечение с осью абсцисс.

Функция **polyroots**(v) для нахождения корней полинома. Решается относительно вектора V коэффициентов полинома (первый элемент вектора является коэффициентом при x^0 , далее по возрастанию порядка степени аргумента).

Вычислительные блоки с функциями **Find** или **Minerr** со следующей структурой:

- начальные условия (задается начальное значение искомой переменной – некоторая опорная точка);
- открытие блока ключевым словом **Given**;
- запись уравнения в виде приближенного равенства (используется знак жирного “=”)
- выражения с функциями **Find** или **Minerr**.

Пример блока **Given - Find** (блок **Given – Minerr** строится по тому же принципу):

$$\begin{aligned} x &:= 1 \\ \text{Given} \\ e^x - x^3 &= 0 \\ x0 &:= \text{Find}(x) \\ x0 &= 1.857 \end{aligned} \quad (3)$$

Кроме того, итерационный метод можно записать в виде программного блока. Функция вычисления корня нелинейного уравнения по методу Ньютона (см. формулы 1 и 2) может быть записана следующим образом:

$$\text{Newton}(f, x) := \left| \begin{array}{l} x_1 \leftarrow x + 2 \cdot \text{TOL} \\ \text{while } |x_1 - x| > \text{TOL} \\ \quad \left| \begin{array}{l} x \leftarrow x_1 \\ x_1 \leftarrow x - \frac{f(x)}{\frac{d}{dx}f(x)} \end{array} \right. \end{array} \right. \quad (4)$$

Для нахождения корня необходимо построить график функции, определить интервал поиска, задать выражение $f(x)$, после чего ввести стартовое значение переменной в качестве параметра x функции Newton.