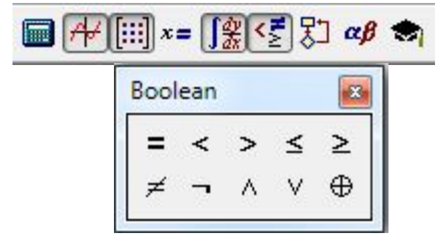
Тема 4. Решение систем нелинейных уравнений

Решение системы нелинейных уравнений в Mathcad

$$\sin(x-0.6)-y=1.6$$

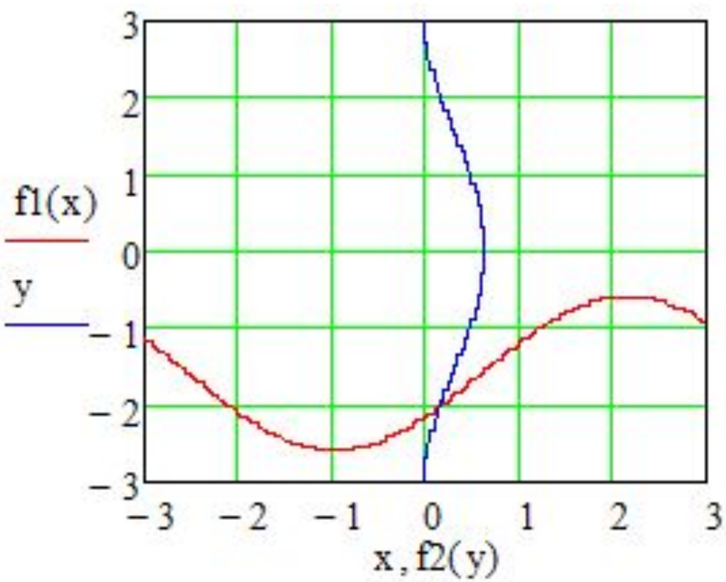
$$3x-\cos y=0.9$$



Решение системы нелинейных уравнений

$$f1(x) := \sin(x-0.6)-1.6$$

$$f2(y) := 0.3+\frac{\cos(y)}{3}$$



Начальные приближения

$x := 0 \qquad y := -2$

Given

$$\sin(x-0.6)-y = 1.6$$

$$3x-\cos(y) = 0.9$$

$$\text{Find}(x,y) = \begin{pmatrix} 0.151057 \\ -2.034013 \end{pmatrix}$$

Решение систем нелинейных уравнений

Основное отличие методов решения систем нелинейных уравнений:

- **используются только итерационные методы.**

Итерационные методы:

- **метод простой итерации;**
- **метод Ньютона.**

Алгоритм метода простой итерации

1. Приведение системы уравнений к виду:

$$x_1 = \varphi_1(x_1, x_2, x_3, \dots, x_n),$$
$$x_2 = \varphi_2(x_1, x_2, x_3, \dots, x_n),$$
$$\cdots\cdots\cdots\cdots$$
$$x_n = \varphi_n(x_1, x_2, x_3, \dots, x_n).$$

2. Задание начального приближения: $x_i^{(0)}$

Алгоритм метода простой итерации

3. Уточнение решения:

$$x_1^{(s+1)} = \varphi_1(x_1^{(s)}, x_2^{(s)}, x_3^{(s)}, \dots, x_n^{(s)}),$$
$$x_2^{(s+1)} = \varphi_2(x_1^{(s+1)}, x_2^{(s)}, x_3^{(s)}, \dots, x_n^{(s)}),$$
$$x_3^{(s+1)} = \varphi_2(x_1^{(s+1)}, x_2^{(s+1)}, x_3^{(s)}, \dots, x_n^{(s)}),$$
$$\cdots\cdots\cdots\cdots$$
$$x_n^{(s+1)} = \varphi_n(x_1^{(s+1)}, x_2^{(s+1)}, x_3^{(s+1)}, \dots, x_n^{(s)}).$$

4. Проверка окончания итерационного процесса:

$$\left| x_i^{(s+1)} - x_i^{(s)} \right| \le \varepsilon \qquad i = 1,2,\dots,n$$

Эта разновидность метода простой итерации построена по аналогии с методом Зейделя, используемым для систем линейных уравнений.

Алгоритм метода простой итерации

Достоинства метода:

- **простота.**

Недостатки:

- **проблема сходимости, если исходные значения лежат за пределами этой области, то решение получить не удается;**
- **с увеличением числа уравнений область сходимости уменьшается;**
- **в случае очень больших систем сходимость обеспечивается лишь при условии, что исходные значения переменных очень близки к истинному решению.**

Область, в которой заданные исходные значения сходятся к решению, называется областью сходимости.

Метод Ньютона

Это наиболее распространенный метод решения системы нелинейных уравнений.
Его популярность обусловлена тем, что по сравнению с другими методами, он обеспечивает более быструю сходимость.
При использовании метода Ньютона система уравнений приводится к виду:

$$f_i(x_1,\ x_2,\ \ldots,\ x_n) = 0$$

Метод Ньютона

В основе метода Ньютона лежит представление всех *n* уравнений в виде рядов Тейлора:

$$f_i(x_1+\Delta x_1, x_2+\Delta x_2, \ldots, x_n+\Delta x_n) =$$

$$= f_i(x_1, x_2, \ldots, x_n) + \frac{\partial f_i}{\partial x_1}\Delta x_1 + \frac{\partial f_i}{\partial x_2}\Delta x_2 + \ldots + \frac{\partial f_i}{\partial x_n}\Delta x_n + \ldots$$

Если приращения переменных Δx_i таковы, что неизвестные x_i принимают значения, близкие к корню, то будем считать, что левые части этих уравнений обращаются в нули.

Метод Ньютона

Система уравнений:

$$\frac{\partial f_1}{\partial x_1}\Delta x_1 + \frac{\partial f_1}{\partial x_2}\Delta x_2 + \ldots + \frac{\partial f_1}{\partial x_n}\Delta x_n = -f_1$$

$$\frac{\partial f_2}{\partial x_1}\Delta x_1 + \frac{\partial f_2}{\partial x_2}\Delta x_2 + \ldots + \frac{\partial f_2}{\partial x_n}\Delta x_n = -f_2$$

$$\cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot$$

$$\frac{\partial f_n}{\partial x_1}\Delta x_1 + \frac{\partial f_n}{\partial x_2}\Delta x_2 + \ldots + \frac{\partial f_n}{\partial x_n}\Delta x_n = -f_n$$

Найденные значения Δx_i в дальнейшем используются как поправки к исходному приближенному решению

$$x_1 = x_1 + \Delta x_1$$

$$x_2 = x_2 + \Delta x_2$$

$$\cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot$$

$$x_n = x_n + \Delta x_n$$

Алгоритм метода Ньютона

1. Приведение системы уравнений к виду:

$$f_i(x_1,\ x_2,\ \ldots,\ x_n) = 0$$

2. Задание начального приближения: $x_i^{(0)}$

3. Решение системы линейных алгебраических уравнений:

$$\frac{\partial f_i}{\partial x_1}\Delta x_1 + \frac{\partial f_i}{\partial x_2}\Delta x_2 + \ldots + \frac{\partial f_i}{\partial x_n}\Delta x_n = -f_i$$

4. Уточнение решения:

$$x_i^{(s+1)} = x_i^{(s)} + \Delta x_i^{(s)}$$

5. Проверка окончания итерационного процесса:

$$\Delta x_i^{(s)} \le \varepsilon \quad \textbf{или} \quad \frac{\Delta x_i^{(s)}}{x_i^{(s)}} \le \varepsilon \qquad i = 1,2,\ldots,n$$

Пример

С использованием метода Ньютона решить систему уравнений с точностью ε=0,001:

$$\begin{cases} \sin(x-0{,}6)-y=1{,}6 \\ 3x-\cos y=0{,}9 \end{cases}$$

1. Приведение системы уравнений к виду:

$$f_1(x,y)=\sin(x-0{,}6)-y-1{,}6=0$$

$$f_2(x,y)=3x-\cos y-0{,}9=0$$

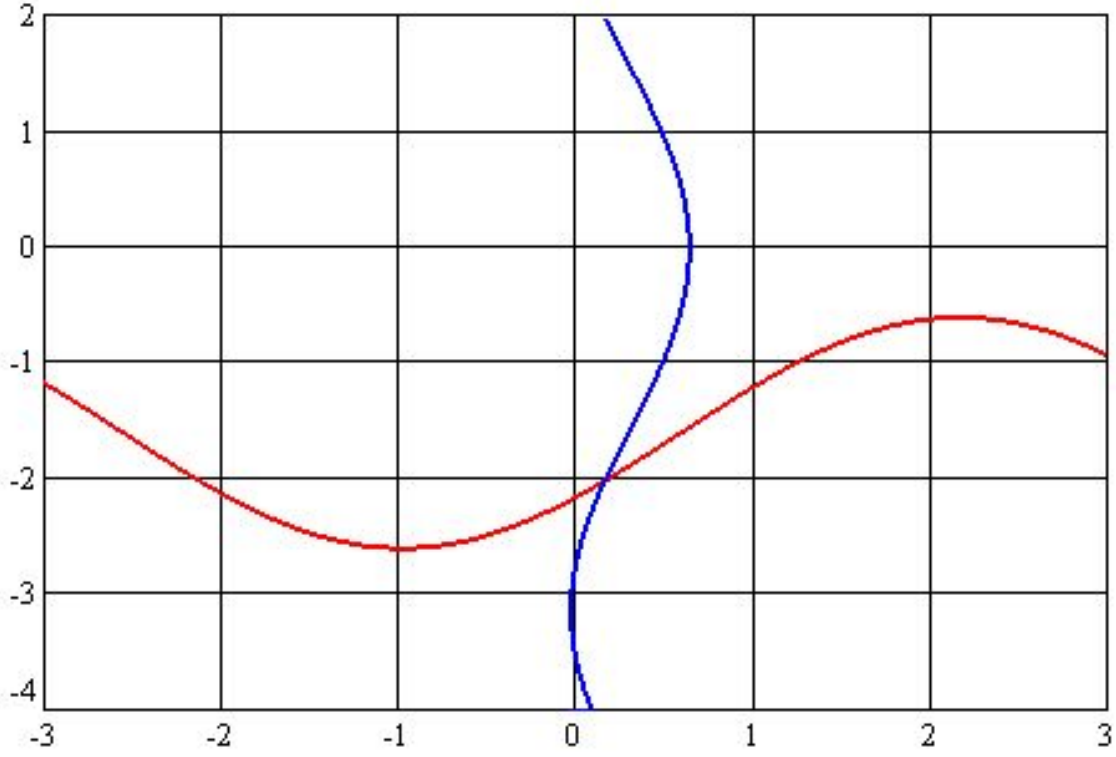
2. Частные производные:

$$\frac{\partial f_1}{\partial x}=\cos(x-0{,}6) \qquad \frac{\partial f_1}{\partial y}=-1$$

$$\frac{\partial f_2}{\partial x}=3 \qquad \frac{\partial f_2}{\partial y}=\sin y$$

Пример

3. Начальные приближения:



$$x_0 = 0 \qquad y_0 = -2$$

Пример

1-я итерация:

Система уравнений:

$$0{,}825336\Delta x - \Delta y = 0{,}164463$$

$$3\Delta x - 0{,}909297\Delta y = 0{,}483853$$

Решение системы уравнений:

$$\Delta x = 0{,}148540 \qquad \Delta y = -0{,}042047$$

Уточнение решения:

$$x_1 = x_0 + \Delta x = 0 + 0{,}148540 = 0{,}148540$$

$$y_1 = y_0 + \Delta y = -2 - 0{,}042047 = -2{,}042047$$

Пример

2-я итерация:

Система уравнений:

$$0{,}899811\Delta x - \Delta y = -0{,}005767$$

$$3\Delta x - 0{,}891001\Delta y = 0{,}000379$$

Решение системы уравнений:

$$\Delta x = 0{,}002510 \qquad \Delta y = 0{,}008026$$

Уточнение решения:

$$x_2 = x_1 + \Delta x = 0{,}148540 + 0{,}002510 = 0{,}151050$$

$$y_2 = y_1 + \Delta y = -2{,}042047 + 0{,}008026 = -2{,}034021$$

Пример

3-я итерация:

Система уравнений:

$$0{,}900903\Delta x - \Delta y = -0{,}0000013$$

$$3\Delta x - 0{,}894616\Delta y = 0{,}0000146$$

Решение системы уравнений:

$$\Delta x = 0{,}000007 \qquad \Delta y = 0{,}000008$$

Уточнение решения:

$$x_3 = x_2 + \Delta x = 0{,}151050 + 0{,}000007 = 0{,}151057$$

$$y_3 = y_2 + \Delta y = -2{,}034021 + 0{,}000008 = -2{,}034013$$

$$|\Delta x| < 0{,}001 \qquad |\Delta y| < 0{,}001$$

Решение системы уравнений: $x = 0{,}151057$

$y = -2{,}034013$

Пример программы

С использованием метода Ньютона решить систему уравнений с точностью ε=10⁻⁶:

$$\begin{cases} \sin(x-0{,}6)-y=1{,}6 \\ 3x-\cos y=0{,}9 \end{cases}$$

1. Приведение системы уравнений к виду:

$$f_1(x,y)=\sin(x-0{,}6)-y-1{,}6=0$$

$$f_2(x,y)=3x-\cos y-0{,}9=0$$

2. Частные производные:

$$\frac{\partial f_1}{\partial x}=\cos(x-0{,}6) \qquad \frac{\partial f_1}{\partial y}=-1$$

$$\frac{\partial f_2}{\partial x}=3 \qquad \frac{\partial f_2}{\partial y}=\sin y$$

Пример программы

```
void func(double *x, double **df, double *f)
{
  f[0] = sin(x[0]-0.6) - x[1] - 1.6;
  f[1] = 3*x[0] - cos(x[1]) - 0.9;
  df[0][0] = cos(x[0]-0.6);
  df[0][1] = -1;
  df[1][0] = 3;
  df[1][1] = sin(x[1]);
}
void rsly_Gauss(double **a, double *x, int n)
{
  ...
}
```

Пример программы

```
int rsny_Newton(double **df, double *x, int n,
                double eps, int itr)
{
  int i, k, error;
  double *f;
  f = new double[n];
  for (k = 0; k < itr; k++)
  {func(x, df, f);
   rsly_Gauss(df, f, n);
   //  Уточнение корней
   for (i = 0; i < n; i++)
     x[i] -= f[i];
   error = 0;
   for (i = 0; i < n && error == 0; i++)
     if (fabs(f[i]) > eps) error = 1;
   if (!error) break;
  }
  delete[] f;
  return error;
}
```

Пример программы

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
  . . .
  error = rsny_Newton(a, x, n, eps, itr);

  if (!error)  // if (error == 0)
   for (i=0; i<n; i++)
     StringGrid2->Cells[i][0] =
            FloatToStrF(x[i], ffFixed, 10, 6);
  else
   ShowMessage("Решение системы не найдено");
 . . .
}
```

Контрольные вопросы

1. **Решение системы нелинейных уравнений в MathCAD.**
2. **Метод простой итерации.**
3. **Метод Ньютона.**

Задание

1. **Решить систему нелинейных уравнений в MathCAD.**
2. **Решить систему нелинейных уравнений с использованием метода Ньютона с точностью 10^{-6}.**
3. **Решить систему нелинейных уравнений с использованием метода Ньютона (C++Builder).**
4. **Решить систему нелинейных уравнений с использованием метода простой итерации с точностью 10^{-3}.**
5. **Решить систему нелинейных уравнений с использованием метода простой итерации (C++Builder).**

Спасибо
за внимание!