Отчёт по теме: “Метод Ньютона”

Постановка задачи

Задана система, состоящая из n нелинейных уравнений с n неизвестными:

$$
\begin{cases}
f_1(x_1,\ x_2, \dots ,\ x_n) \ = \ 0; \\
f_2(x_1,\ x_2, \dots ,\ x_n) \ = \ 0; \\
. \\
. \\
. \\
f_k(x_1,\ x_2, \dots ,\ x_n) \ = \ 0; \\
. \\
. \\
. \\
f_n(x_1,\ x_2, \dots ,\ x_n) \ = \ 0.
\end{cases}
$$

$\forall\ i \ = \ 1, \dots, n \quad f_i : R^n \to R$ − нелинейные функции, определенные и достаточно гладкие в некоторой области $\subset\ R^n$.

Перепишем систему в векторном виде:

$\overline{x} = (x_1,\ x_2, \dots ,\ x_n)^{\mathrm{T}},\ \overline{F}(x) \ = \ [\, f_1(\overline{x}), f_2(\overline{x}), \dots, f_n(\overline{x})]^{\mathrm{T}} \ = \ \overline{0};$

Требуется найти вектор $\overline{x^*} = (x_1{}^*,\ x_2{}^*, \dots ,\ x_n{}^*)^{\mathrm{T}} -$ решение данной системы.

Реализация

Будем искать решение с помощью итерационного метода Ньютона.

В многомерном случае формула для нахождения решения является обобщением формулы одномерного итеративного метода и имеет вид:

$$x^{(k+1)} \ = \ x^{(k)} - J^{-1}(\,\widehat{x^{(k)}}\,) * \ \overline{F}(\,x^{(k)}\,),\ k \ = \ 0, 1, 2\dots ,$$

где $J(\overline{x}) \ = \left[\frac{\partial f_i(x_j)}{x_j}\right]$ − матрица Якоби первых производных, взятая в точке

$\widehat{x^{(k)}} \ = \ (x_k^0,\ x_2^k,\ x_3^k,\ \dots,\ x_n^k),\ k \ = \ 1, 2, \dots,\ \widehat{x^{(0)}} = \ x^{(0)}$.

Для реализации метода Ньютона необходимо, чтобы матрица Якоби $J(\overline{x})$ была невырождена в некоторой окрестности точки $\overline{x} \ = \ \overline{x^*}$.

Тогда обратная матрица $J^{-1}(\overline{x})$ существует в данной окрестности.

На каждом шаге будем вычислять $J_k^{-1}(\,x^{(k)}\,)$, используя метод отражений.

Для получения компонентов матрицы Якоби будем вычислять производные по определению как отношение приращения функции к приращению аргумента.

Возьмём приращение аргумента $dx_j = [\,\widehat{x^{(k)}} - x^{(k)}\,]_j = \frac{1}{\varepsilon_1}$ по $j-$ой координате, остальные фиксированы. Тогда приращение функции имеет следующий вид: $df_{ij} = f_i(\,[\,\widehat{x^{(k)}}]_j\,) - f_i(\,[x^{(k)}]_j\,)$, а $(ij)-$ая компонента матрицы вычисляется как $df_{ij} * \varepsilon_1$.

Для сходимости последовательности $\{\,x^{(k)}\,\}$ к решению необходимо выбрать начальное приближение $x^{(0)}$ из достаточно малой окрестности $\overline{x^*}$.
При предположении $F \in C^2$ метод имеет квадратичную сходимость.

Алгоритм поиска решения завершается на шаге k, когда $\|\,F(x_1^k, x_2^k, ..., x_n^k)\| < \varepsilon_2$.

В программе положим $\varepsilon_1 = 10^{10}$, $\varepsilon_2 = 10^{-15}$.

Результаты работы программы

Для системы нелинейных уравнений с n = 2 неизвестными

$$\begin{cases} f_1(x_1,\ x_2)\ =\ e^{x_1} * sin(\pi * x_2)\ -\ 1; \\ f_2(x_1,\ x_2)\ =\ sin(x_1)\ +\ cos(\pi * x_2); \end{cases}$$

и вектора начального приближения $x^{(0)} = (1,\ 0.2)$ был получен результат:

Итерация: 1
1.00000000000000 0.20000000000000

Итерация: 2
-0.87134990337327 0.54625650858499

Итерация: 3
0.54446560289322 0.54673472362455

Итерация: 4
0.13399966919180 0.55334793557945

Итерация: 5
-0.00899743918670 0.49687466081420

Итерация: 6
-0.00000709842489 0.49999776486407

Итерация: 7
-0.00000000000099 0.50000000000055

Итерация: 8
0.00000000000000 0.50000000000000

Решение =
0.00000000000000 0.50000000000000 .

Для вектора начального приближения $x^{(0)} = (0.4,\ 0.4)$ был получен результат:

Итерация: 1
0.40000000000000   0.40000000000000

Итерация: 2
-0.00603119573797   0.50859234812279

Итерация: 3
-0.00035369782081   0.49988526130544

Итерация: 4
-0.00000000229660   0.49999999923280

Итерация: 5
-0.00000000000000   0.50000000000000

Решение =
-0.00000000000000   0.50000000000000

Для системы нелинейных уравнений с n = 3 неизвестными

$$\begin{cases} f_1(x_1,\ x_2,\ x_3)\ =\ \frac{x_1^2}{8}\ -\ sin(\frac{x_2}{6}); \\ f_2(x_1,\ x_2,\ x_3)\ =\ sin(\frac{x_1 * x_2}{4})\ +\ sin(x_2)\ -\ e^{x_3}; \\ f_3(x_1,\ x_2,\ x_3)\ =\ sin(x_1 * x_2)\ +\ cos(x_2)\ +\ e^{x_3}; \end{cases}$$

и вектора начального приближения $x^{(0)} = (2.3,\ 2.5,\ 1)$ был получен результат:

Итерация: 1
2.30000000000000 2.50000000000000 1.00000000000000

Итерация: 2
2.00595187032576 3.07384346978777 0.42280359191285

Итерация: 3
2.00033590117168 3.14250518226883 0.07777919388799

Итерация: 4
2.00000009044273 3.14159282600081 0.00294798310335

Итерация: 5
1.99999999999996 3.14159265359001 0.00000434380375

Итерация: 6
2.00000000000000 3.14159265358979 0.00000000001022

Итерация: 7
2.00000000000000 3.14159265358979 -0.00000000000000

Решение =
2.00000000000000 3.14159265358979 -0.00000000000000 .