Numerical Methods | Implementations in C++

Thanasis Mattas
Aristotle University of Thessaloniki, Greece | 2019
atmattas@physics.auth.gr

Fixed point iteration schemes

1. **Picard method** $[x_{n+1} = g(x_n)]$

Example: $f(x) = e^{2x} - 3x - 1 \Rightarrow x = \frac{e^{2x}-1}{3}$ with $g(x) = \frac{e^{2x}-1}{3} \Rightarrow g'(x) = \frac{2}{3}e^{2x}$

So, the regression formula is:

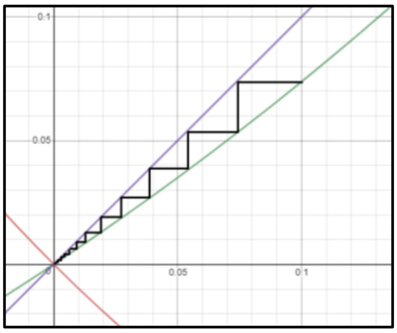
$$\boxed{x_{n+1} = \frac{e^{2x_n} - 1}{3}}$$

, the convergence rate of the error is:

$$\boxed{e_{n+1} = g'(r)e_n}$$

and the convergence condition is:

$$\boxed{|g'(x)| < 1 \Rightarrow x < 0.20273}$$



desmos.com

For the solution you can refer to **Picard_method.cpp** at the src directory ($x_0 = 0.1$).

2. **Newton-Raphson method** $\left[x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}\right]$

In order to compare the convergence speed against the Picard method, the same example is used:

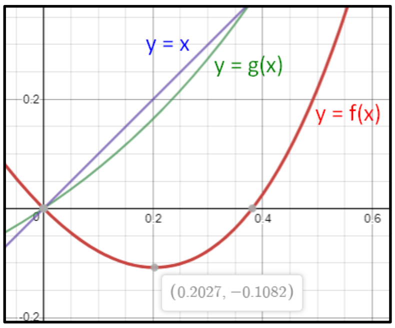
$$f(x) = e^{2x} - 3x - 1$$

Regression formula:

$$\boxed{x_{n+1} = x_n - \frac{e^{2x_n} - 3x - 1}{2e^{2x_n} - 3}}$$

Convergence rate of the error:

$$\boxed{e_{n+1} = -\frac{f''(r)}{2f'(r)}e_n^2}$$



desmos.com

2.b Selecting $x_0 = 0.203$

$x = 0.20273$ is the total extremum (min) of the equation $f(x)$. So, starting points that lie before the extremum lead to the root $x = 0$, and fixed points that lie after the exremum lead to the other root of the equation, $x = 0.3813$.

For the solution you can refer to **Newton-Raphson_method.cpp** at the src directory.

3. **Newton method with simultaneous equations** $\begin{bmatrix} x_{n+1} = x_n - \frac{fg_x - gf_y}{f_x g_y - g_x f_y} \\ y_{n+1} = y_n - \frac{gf_x - fg_y}{f_x g_y - g_x f_y} \end{bmatrix}$ $\left(f_i = \frac{\partial f}{\partial i}\right)$

Example: $\begin{matrix} x^2 + y^2 = 3 \rightarrow f(x) = x^2 + y^2 - 3 \\ 3x^2 - 9y^2 = 6 \rightarrow g(x) = 3x^2 - 9y^2 - 6 \end{matrix} \quad \Rightarrow \quad \begin{matrix} f_x = 2x,\ f_y = 2y \\ g_x = 6x,\ g_y = -18y \end{matrix}$

Regression formulas:

$$\boxed{\begin{matrix} x_{n+1} = x_n - \dfrac{4x^2 - 1}{8x} \\ y_{n+1} = y_n - \dfrac{4y^2 - 1}{8y} \end{matrix}}$$

For the solution you can refer to **Newton_method-Simultaneous_Equations.cpp** at the src directory.

4. **Picard method with simultaneous equations** $\begin{bmatrix} x_{n+1} = f_1(x_n, y_n) \\ y_{n+1} = f_2(x_n, y_n) \end{bmatrix}$

Using the same example as above:

$$\boxed{x_{n+1} = \sqrt{3 - y_n^2}} \qquad \boxed{y_{n+1} = \frac{\sqrt{x_n^2 - 2}}{3}}$$

Convergence conditions:

$$\boxed{\left|\frac{\partial f_1(y)}{\partial y}\right| < 1 \Rightarrow y_n > \sqrt{\frac{3}{2}}} \qquad \boxed{\left|\frac{\partial f_2(x)}{\partial x}\right| < 1 \Rightarrow x > \sqrt{3}}$$

The solutions estimated with the previous method are out of the limits of these conditions and, consequently, no starting point can lead to convergence. Note that the Picard method needs a good approximation of the solution (possibly acquired with another method), in order to meet convergence.

For the corresponding implementation you can refer to **Picard_method-Simultaneous_Equations.cpp** at the src directory.