# Fixed-point method for Nonlinear equation and Non-linear System

Gattadi Vivek (2019MCB1217)
Sophomore, B.Tech. Mathematics and Computing, IIT Ropar

A project under MA205 course submitted to Dr. Manoranjan Mishra, Faculty at Department of Mathematics, IIT Ropar.

# Contents

# Introduction

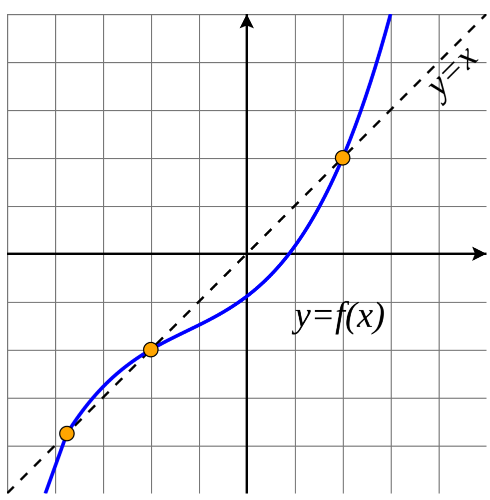
Over the years, we have been taught on how to solve equations using various algebraic methods. These methods include the substitution method and the elimination method. Other algebraic methods that can be executed include the quadratic formula and factorization. In Linear Algebra, we learned that solving systems of linear equations can be implemented by using row reduction as an algorithm. However, when these methods are not successful, we use the concept of numerical methods.

Numerical methods are used to approximate solutions of equations when exact solutions can not be determined via algebraic methods. They construct successive approximations that converge to the exact solution of an equation or system of equations

## Motivation behind designing Fixed-point method:

### 1.1 What is a fixed point?

A fixed point of a function is an element of function's domain that is mapped to itself by the function. i.e. Let $\varphi:[a,b]\to[a,b]$ be a function, then $\exists\ \boldsymbol{\alpha} \in [a,b]$ is called fixed point of $\varphi$ iff $\varphi(\boldsymbol{\alpha}) = \boldsymbol{\alpha}$.




**A function f(x) with 3 fixed points**

- If $\varphi(x)$ is any continuous function in [a,b]. Such that $\varphi(x) \in [a,b]$ and $|\varphi'(x)| < 1$ for all $x \in [a,b]$ then there exists a unique $\boldsymbol{\alpha} \in [a,b]$ such that $\varphi(\boldsymbol{\alpha}) = \boldsymbol{\alpha}$.

## 1.2 What is "Fixed point Iteration Method" and its use?

In Numerical Analysis, it is a method of finding roots of a function by computing fixed points by doing some iterations to the functions. Let us see the following two theorems and understand the connectivity between them.

- If $\varphi(x)$ is a differentiable function in [a,b] such that $\max\{|\varphi'(x)|\} = k$, where $k < 1$ and $\varphi(x) \in [a,b]$ for all $x \in [a,b]$. Then the sequence defined by $\mathbf{x_{n+1} = \varphi(x_n)}$ for n = 0,1,2,3,... where $x_o \in [a,b]$ converges to the fixed point '$\boldsymbol{\alpha}$' of $\varphi(x)$ in the interval [a,b].
- If $\varphi(x)$ is an iterative function of the equation $f(x) = 0$ in the interval [a,b] then the iterative procedure for the root as of the equation $f(x) = 0$ is given by $\mathbf{x_{n+1} = \varphi(x_n)}$ for n = 0,1,2,3,... where $x_o \in [a,b]$ .

So, from the above two theorems, we can say that if want to find the approximate root of a function $f(x) = 0$, and if we are able to write $f(x)$ as $f(x) = x - \varphi(x)$, where $\varphi(x)$ satisfies the above conditions, then the root of $f(x) = 0$ is the root of $x = \varphi(x)$ i.e the fixed points of $\varphi(x)$ in [a,b], which can be found by performing iterations on $\mathbf{x_{n+1} = \varphi(x_n)}$ for n = 0,1,2,3,... where $x_o \in [a,b]$ is an arbitrary initial value. Hence the above theorems tell us the necessity of fixed points in finding the roots of a function.

# Theory

## General Algorithm for Fixed-Point Method:

### Algorithm:

Let us consider the following equation which we interested to find an approximate root:

$$f(x) = 0 \qquad (1)$$

In this method, we first rewrite the equation $f(x) = 0$ in the form

$$x = \varphi(x) \qquad (2)$$

in such a way that any solution of the equation (2), which is a fixed point of $\varphi$, is a solution of equation (1). Then consider the following algorithm:

1. Start from any point $x_o$ and consider the recursive process
$$x_{n+1} = \varphi(x_n),\ n = 0, 1, 2, \dots \qquad (3)$$
2. If f is continuous and $(x_n)$ converges to some $l_o$ then it is clear that $l_o$ is a fixed point of $\varphi$ and hence it is a solution of the equation (1). Moreover, $x_n$ (for a large n) can be considered as an approximate solution of the equation (1).

## Application of fixed-point iteration to a function f (x):

Consider $f(x) = \ln(x) + x - 2.4$

The iteration formula is $x_{k+1} = 2.4 - \ln(x_k)$ (In the form $x_{n+1} = \varphi(x_n)$ )

Starting with $x_o = 2$ , the values in the below table are obtained:

| k | $x_k$ | k | $x_k$ | k | $x_k$ |
|---|---|---|---|---|---|
| 0 | 2.000 000 000 00 | 7 | 1.804 911 890 15 | 14 | 1.807 904 231 22 |
| 1 | 1.706 852 819 44 | 8 | 1.809 488 223 73 | 15 | 1.807 831 708 89 |
| 2 | 1.865 348 781 67 | 9 | 1.806 955 944 02 | 16 | 1.807 871 823 73 |
| 3 | 1.776 551 950 09 | 10 | 1.808 356 369 42 | 17 | 1.807 849 634 50 |

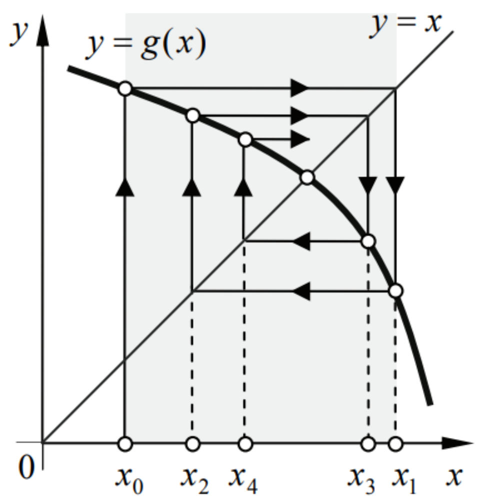
| | | | | | |
|---|---|---|---|---|---|
| 4 | 1.825 325 621 01 | 11 | 1.807 581 650 48 | 18 | 1.807 861 908 25 |
| 5 | 1.798 241 606 41 | 12 | 1.808 010 152 82 | 19 | 1.807 855 119 13 |
| 6 | 1.813 190 697 89 | 13 | 1.807 773 122 55 | 20 | 1.807 857 537 |

## Path of Convergence:

From the earlier sessions, we have understood that if max{|φ'(x)|} = k, where k < 1 and φ(x) ∈ [a,b] for all x ∈ [a,b] then sequence defined by $\mathbf{x_{n+1} = \varphi(x_n)}$ for n = 0,1,2,3,... where $x_o$ ∈ [a,b] converges to the fixed point '$\boldsymbol{\alpha}$' of φ(x) in the interval [a,b].

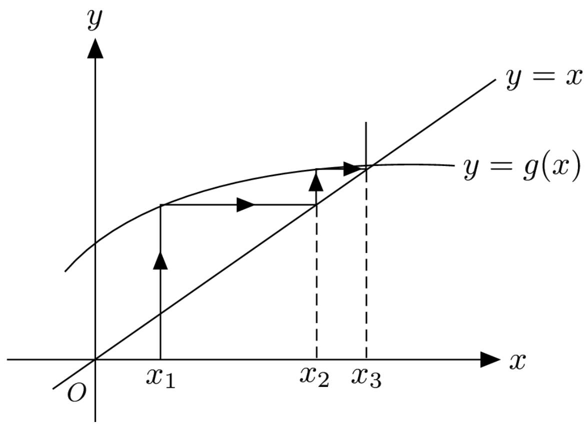
Depending upon weather sign of φ'(x), there exists two paths for convergence:

1. Spiral path to the root:




Let g(x) = φ(x). If -1 < φ'(x) < 0, then the convergence has a 'spiral' path to the root.

2. Zig-Zag path to the root:




Let g(x) = φ(x). If 0 < φ'(x) < 1, then the convergence has a 'zig-zag' path to the root.

## Newton Raphson's Method (1 variable function):

The following iterative method used for solving the equation f(x) = 0 is called Newton's method.

### Algorithm:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad n = 0, 1, 2, \ldots.$$

It is understood that here we assume all the necessary conditions so that $x_n$ is well defined. If we take g(x) = x −(f(x)/ f '(x)) then the Newton Raphson algorithm is a particular case of Fixed point algorithm.

## Extension of idea of Fixed-point to higher dimensions:

### System of Nonlinear Equations:

Let us consider the system of Nonlinear equations:

$$f_1(x_1, x_2, \ldots, x_n) = 0,$$
$$f_2(x_1, x_2, \ldots, x_n) = 0,$$
$$\vdots$$

$$f_n(x_1, x_2, ..., x_n) = 0,$$

where $(x_1, x_2, ..., x_n) \in R^n$ and each $f_i$ is a nonlinear real function $\in R$, i = 1, 2,..., n

Let us consider the Jacobian Matrix 'J(x)' of the above system:

$$J(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x) & \frac{\partial f_1}{\partial x_2}(x) & \cdots & \frac{\partial f_1}{\partial x_n}(x) \\ \frac{\partial f_2}{\partial x_1}(x) & \frac{\partial f_2}{\partial x_2}(x) & \cdots & \frac{\partial f_2}{\partial x_n}(x) \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_n}{\partial x_1}(x) & \frac{\partial f_n}{\partial x_2}(x) & \cdots & \frac{\partial f_n}{\partial x_n}(x) \end{bmatrix}$$

where $x = (x_1, x_2, ..., x_n)$.
Let $J_K$ be defined as the value Jacobian at a particular value of $x = (x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})$ i.e $J_K = J(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})$ .

Let **F**: $R^n \to R^n$ be defined as follows:

$$\mathbf{F}(x_1, x_2, ..., x_n) = \begin{bmatrix} f_1(x_1, x_2, ..., x_n) \\ f_2(x_1, x_2, ..., x_n) \\ \vdots \\ f_n(x_1, x_2, ..., x_n) \end{bmatrix}$$

Let $\mathbf{x} \in R^n$ .Then x can be represented as:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Let **F** be a real function from $D \subset R^n$ to $R^n$ . If $\mathbf{F}(p) = p$, for some $p \in D$, then p is said to be a 'fixed point' of **F**.

Let $\mathbf{x}^{(0)}$ be chosen arbitrarily:

$$x^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix}$$

We know from Linear Algebra that we can take systems of equations and express those systems in the form of matrices and vectors. With this in mind and using we can express the nonlinear system as a matrix with a corresponding vector. Thus, the following equation is derived:

$$x_{(k+1)} = x_{(k)} - ((J_K)^{-1})*(F(x_{(k)}))$$

where $x_{(k+1)}$ ,$x_{(k)}$, $x_{(k-1)} \in R^n$ and $(J)^{-1}$ is defined as follows:

$$J(\mathbf{x})^{-1} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\mathbf{x}) & \frac{\partial f_1}{\partial x_2}(\mathbf{x}) & \cdots & \frac{\partial f_1}{\partial x_n}(\mathbf{x}) \\ \frac{\partial f_2}{\partial x_1}(\mathbf{x}) & \frac{\partial f_2}{\partial x_2}(\mathbf{x}) & \cdots & \frac{\partial f_2}{\partial x_n}(\mathbf{x}) \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_n}{\partial x_1}(\mathbf{x}) & \frac{\partial f_n}{\partial x_2}(\mathbf{x}) & \cdots & \frac{\partial f_n}{\partial x_n}(\mathbf{x}) \end{bmatrix}^{-1}$$

This equation represents the procedure of Newton's method for solving nonlinear algebraic systems. However, instead of solving the equation f(x) = 0, we are now solving the system **F**(x) = 0 (having $x^{(0)}$ as initial approximation) with the similar analogy in the single variable case.

# Computational Aspects

## Convergence Criteria:

Till now we have seen the methods to find the approximate root of a given function. Now we are interested in finding the fact that under what conditions the method converges to the root of the function.

The necessary and sufficient condition for the above iteration to converge is, for each iteration $\rho((J_K)^{-1}) < 1$ where 'ρ' is the spectral radius.

## Stopping point of Iteration:

Now the question arises when to stop the iteration. For this we choose to fix our tolerance 'ε' (> 0) in the beginning of the iteration. So when the error which is defined below is less than tolerance, then we stop the iteration.

$$|| x_{(k+1)} - x_{(k)} || < \varepsilon$$

## Error Order:

- The iteration method $\mathbf{x_{n+1} = \varphi(x_n)}$ n = 0,1,2,3... ,is said to be of the order 'p' if $\varphi'(e) = \varphi''(e) = \ldots\ldots = \varphi^{(p-1)}(e) = 0$ and $\varphi^{(p)}(e) != 0$ where e is the solution of $\varphi(x) = x$.

- The error order depends upon how we modify our given function f(x) in the form of $x = \varphi(x)$.

## Computational Cost:

Computational cost is the execution time per time step during simulation. To estimate the time that it takes for your model to execute on real-time hardware, estimate the simulation execution-time budget for your real-time target machine.

For each iteration, we have to compute jacobi, matrix multiplication in the iterative equation, norm which takes a higher computational cost. If the initial guess is closer to the actual root, then it takes less time to get our desired approximation to the root. If the initial guess is not closer to the root, then it takes a relatively longer time to get our desired approximation to the root.

## Newton Raphson Vs Secant Method:

Let us briefly go through the Secant method:

## Secant Method:

$$x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$

In this method we choose two initial values $x_0$ and $x_1$ and iterate to get our root. It is understood that here we assume all the necessary conditions so that $x_n$ is well defined. Here we cannot write in g(x) function as both $x_n$ and $x_{n-1}$ are involved.

### Advantages of Newton Raphson over Secant Method:

- Newton Raphson has order of convergence of '2' whereas Secant Method has order of convergence of '1.618'. This implies that Newton Raphson converges faster.
- In a given interval, in Newton Raphson we can test whether the iteration converges to the root or not by doing the derivative condition test (jacobian test for system) whereas there is no such condition for the Secant Method.

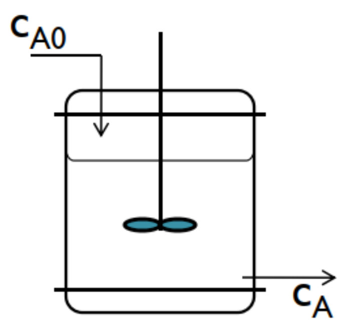
### Disadvantages of Newton Raphson over Secant Method:

- Secant method doesn't require to compute the derivative of the function. If the function is not differentiable, the Newton Raphson method fails. Since there is no derivative term in the Secant Method, this problem won't arise.

# Real Life Model Application

## Application:

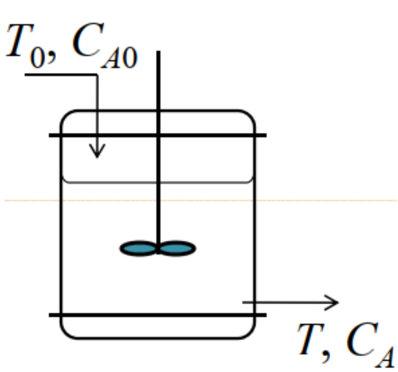
### Catalytic reaction in a CSTR with L-H Kinetics:




$$\underbrace{\frac{C_{A0}}{\tau}-\frac{C_A}{\tau}-\frac{kC_A}{(1+KC_A)^2}}_{f(C_A)}=0$$

In the Catalytic reaction in a CSTR, we have to find the roots of $f(C_A)$. So we can rearrange the above equation in $x = \varphi(x)$ form where $x = C_A$ .Hence we solve it by using the above method and simultaneously checking the conditions.

### Adiabatic CSTR:



$$\underbrace{\frac{C_{A0}}{\tau} - \frac{C_A}{\tau} - k_0 e^{-E/RT)} C_A^{1.65}}_{f_1(C_A, T)} = 0$$

$$\underbrace{\frac{T_0}{\tau} - \frac{T}{\tau} + \frac{-\Delta H}{\rho c_p}\left[k_0 e^{\left(-E/RT\right)} C_A^{1.65}\right]}_{f_2(C_A, T)} = 0$$

The Adiabatic CSTR has two parameters to compute 'T' and '$C_A$' and has two nonlinear equations to solve. So to find the root of the system, we follow and apply the methods that we have learnt in this project.

## Conclusion:

Hence, we can see the importance of iterative methods in solving the non-linear equations and non-linear system of equations. The fixed point method helps us to conclude in advance whether the iterative equation converges to the root in the given interval by some conditions. If the conditions are satisfied, then the method converges to the root of the equation.

------Thank You------