ePG Pathshala

Subject: Information Technology
Paper : Numerical Methods
Module 6: Newton Raphson method for finding roots of an equation f(x) = 0

1.1 Introduction

In our endeavor of increasing the speed of convergence, we shall study now, yet another method, namely Newton Raphson Method for finding roots of an equation $f(x)=0$. It shall turn out to be the fastest in comparison to the methods: Bisection Method, Method of False Position and Secant Method studied so far. All these methods have linear convergence whereas Newton Raphson Method has second order convergence.

1.2 Iterative Process

One very big difference in Newton Raphson method from earlier methods is that only one initial guess is required as compared to two initial guesses required in other methods. Moreover, there is no constraint on the initial guess of bracketing the root. Automatically, it falls under the class of open methods. The iterative formula in Newton Raphson method for generating the sequence of approximations is:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \qquad \text{, for all n = 0, 1, 2, …} \qquad \text{… (1)}$$

where $f'(x_n)$ stands for value of derivative of function $f(x)$ at x_n. This can be treated as modification of Secant method as follows:

Iterative formula for Secant Method is $x_{n+1} = \dfrac{x_{n-1}f(x_n) - x_n f(x_{n-1})}{f(x_n) - f(x_{n-1})}$ … (2)

which is algebraically equivalent to $x_{n+1} = x_n - \dfrac{f(x_n)}{\dfrac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}}$ … (3)

Equation (3) can be derived from (2) as follows:

Subtracting and adding $x_n f(x_n)$ in numerator of equation (2) gives us,

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

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

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

$$= x_n - \frac{f(x_n)}{\dfrac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}}$$, which is same as equation (3)

When the sequence of approximations is close to the root, x_n and x_{n-1} are also quite near to each other. The ratio $\frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$ would be approximately equal to the derivative of function $f(x)$ at x_n. (We know $\lim_{a \to b} \frac{f(b) - f(a)}{b - a} = f'(b)$)

$$\therefore \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}} \cong f'(x_n) \quad \text{... (4)}$$

Substitution in (3) from (4) gives iterative formula (1): $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$

Thus, this method requires only one initial guess, but **does require two function evaluations,** value of function and that of its derivative**.**

Thus, the iterative process is:

Make an initial guess x_0 for the root of $f(x) = 0$. Compute $f(x_0)$ and $f'(x_0)$.

Calculate next approximation x_1 by the formula $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$ ... (5)

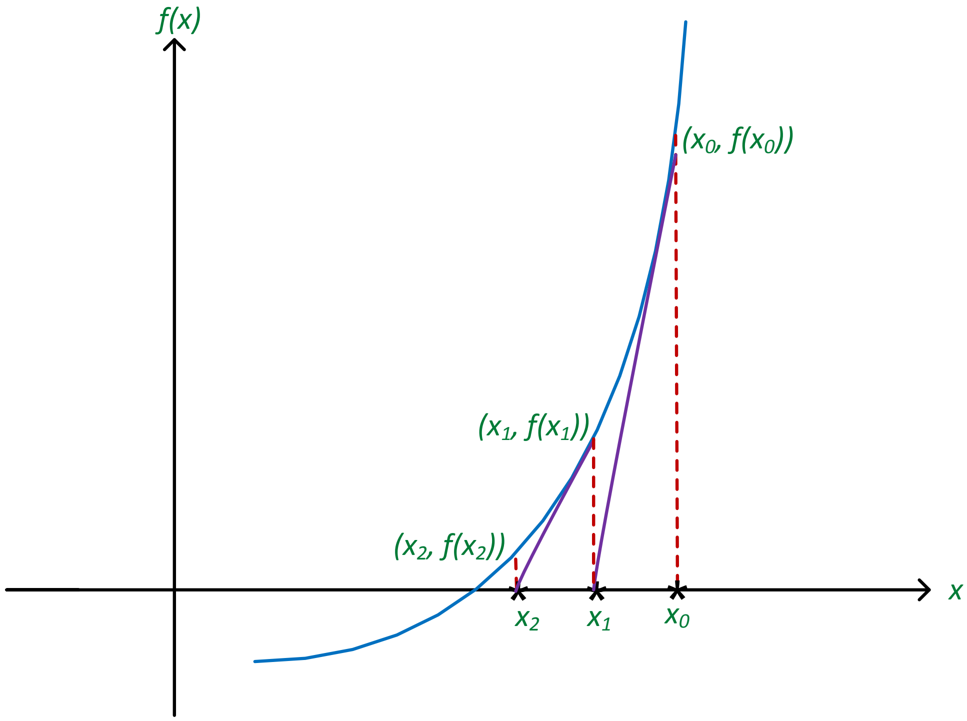
Once x_1 is obtained calculate $f(x_1)$ and $f'(x_1)$. Get next approximation x_2 by

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} \quad \text{... (6)}$$

and so on.

Let us understand this iterative process graphically.

1.3 Graphical Representation



[Figure – 1]

So, graphically, the process is

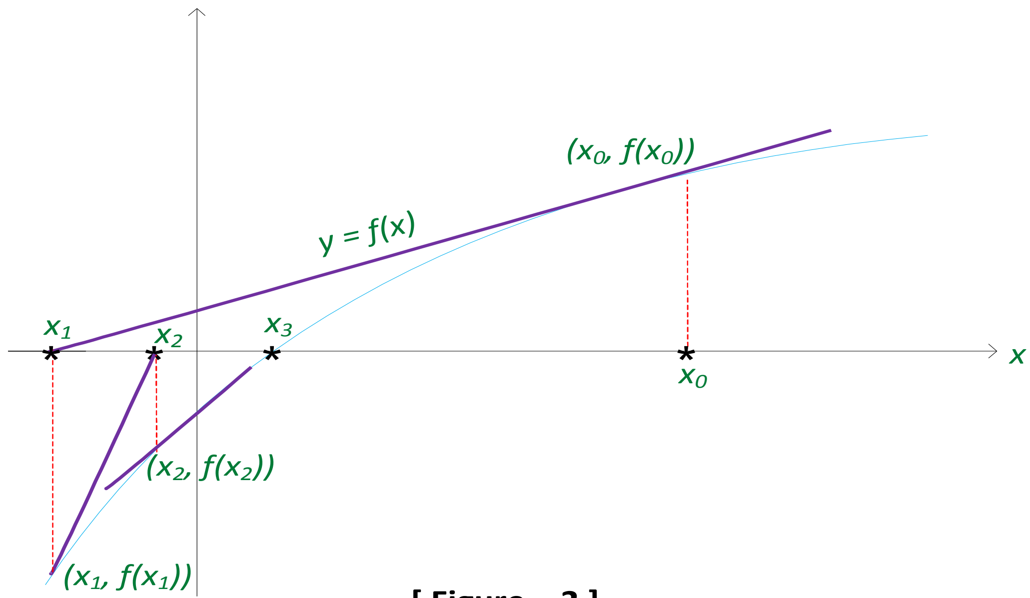
Step 1 Take initial guess x_0 .

Step 2 Locate the point $(x_0, f(x_0))$ on the curve $y = f(x)$

Step 3 Draw tangent to the curve $y = f(x)$ at $x = x_0$

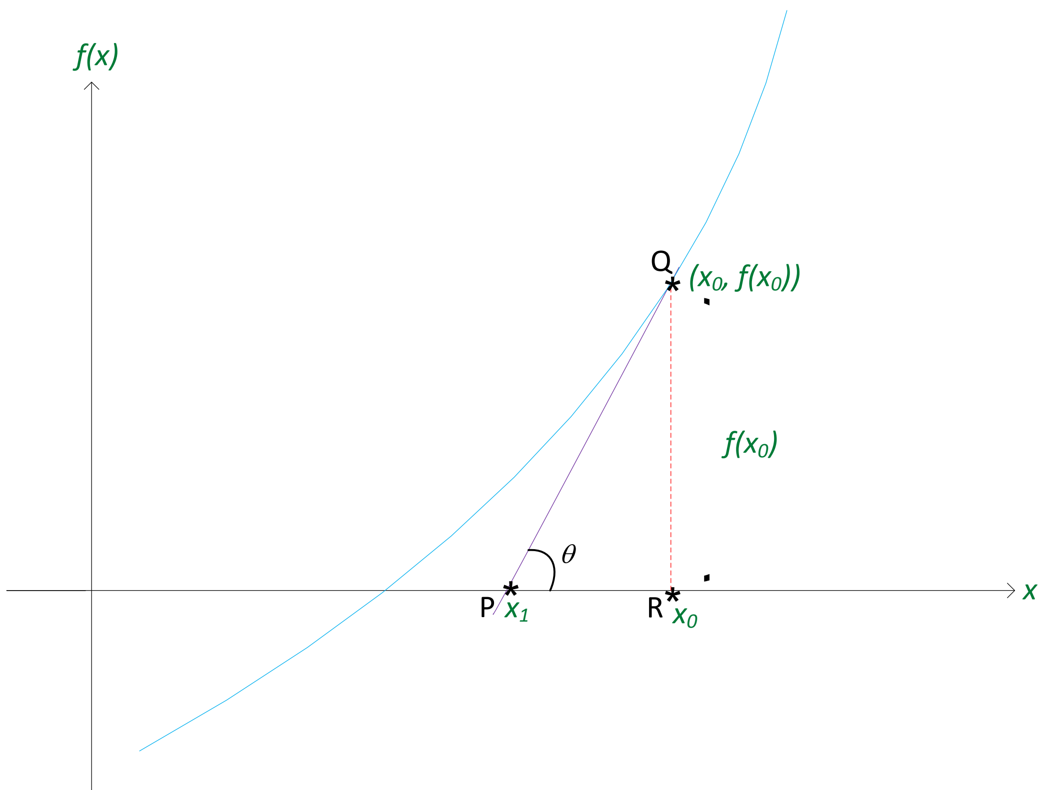
Step 4 The point at which this tangent meets X axis is x_1.

Step 5 Keep on repeating the steps with 2 – 4, till the root to desired accuracy is obtained.



[Figure – 2]

This graphical interpretation of formula (5) can be understood as follows:



[Figure – 3]

If we look at the triangle so formed namely PQR, if θ is the angle made by tangent to the curve $y = f(x)$ at $= x_0$, giving $\tan\theta = \dfrac{f(x_0)}{x_0 - x_1}$. But $\tan\theta = slope = f'(x_0)$

$$\therefore x_0 - x_1 = \frac{f(x_0)}{f'(x_0)},$$

$$\therefore x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

1.4 Stopping Criteria

Any of the stopping criteria can be applied. Usually whenever absolute error or relative error becomes smaller then preassigned upper limit of the error, process is stopped.

Though a quite fast method, this method also does not guarantee converge, being open method, hence it is always advisable to keep an upper limit on the number of iterations. Whenever desired accuracy is not achieved in a specified number of iterations, it is always desired and advised to exit and decide upon future course of action on looking at the iteration values. We shall be looking at different situations of divergence of this method in section 1.7

1.5 Algorithm for Newton Raphson Method

Step 1 Input: $f(x)$ the given function

x_0 : the initial guess for the root

$f'(x)$: the derivative of the function

ε : the error tolerance (X – TOL) ……

N: the maximum number of iterations.

Step 2 $k = 0$

Step 3 Compute $f(x_k)$ and $f'(x_k)$

If $f(x_k) = 0$,

output: estimate of the root is x_k, exit

Step 4 Compute next approximation

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Step 5 Test for convergence

If $|x_{k+1} - x_k| < \varepsilon$ or / and $\left|\frac{x_{k+1} - x_k}{x_k}\right| < \varepsilon$

Output: estimate of the root is x_{k+1}, exit

Step 6 $k = k + 1$ if $k \le N$, go to step 3

Step 7 Output: Does not converge in N iterations

1.6 Illustrations

(i) $f(x) = x^3 + x - 1$ $x_0 = 1$

No.	x_n	$f(x_n)$	$f'(x_n)$	$f(x_n)/f'(x_n)$	x_{n+1}	Error
1	0.000000	-1.000000	1.000000	-1.000000	1.000000	1.000000
2	1.000000	1.000000	4.000000	0.250000	0.750000	0.250000
3	0.750000	0.171875	2.687500	0.063953	0.686046	0.063954
4	0.686046	0.008941	2.411979	0.003707	0.682340	0.003707
5	0.682340	0.000028	2.396762	0.000012	0.682328	0.000012
6	0.682328	0.000000	2.396714	0.000000	0.682328	0.000000

The root of the equation = 0.682328

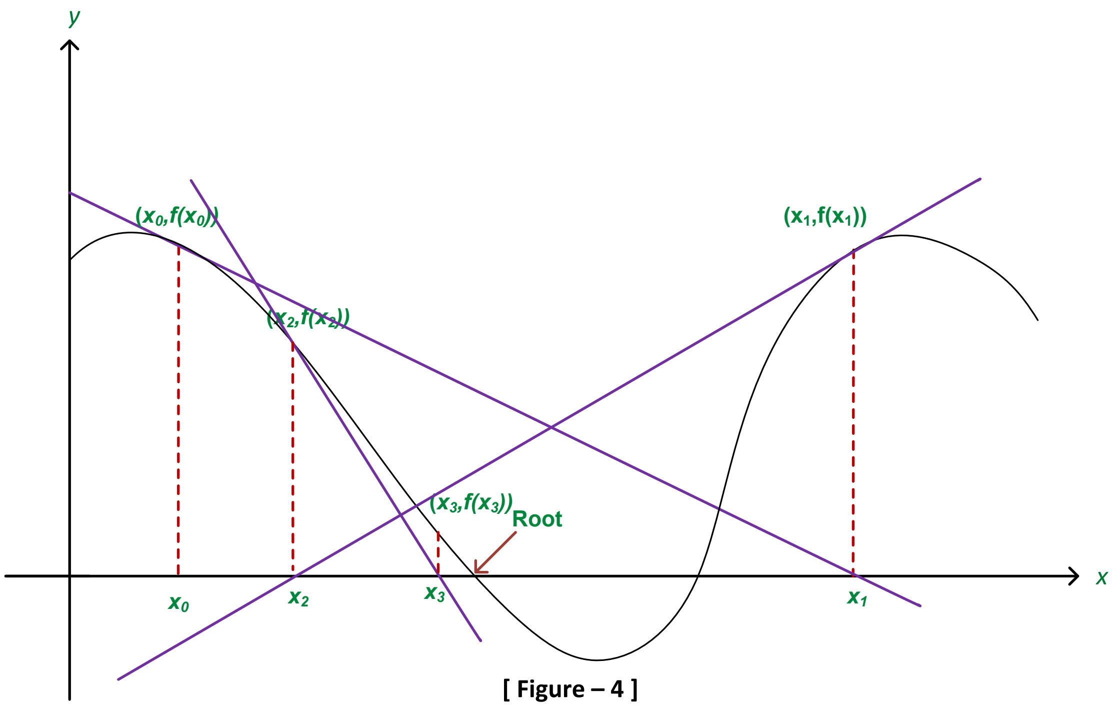
(ii) $f(x) = 3x - \cos(x) - 1$

No.	x_n	$f(x_n)$	$f'(x_n)$	$f(x_n)/f'(x_n)$	x_{n+1}	Error
1	0.000000	-2.000000	3.000000	-0.666667	0.666667	0.666667
2	0.666667	0.214113	3.618370	0.059174	0.607493	0.059174
3	0.607493	0.001397	3.570811	0.000391	0.607102	0.000391
4	0.607102	0.000000	3.570489	0.000000	0.607102	0.000000

The root of the equation = 0.607102

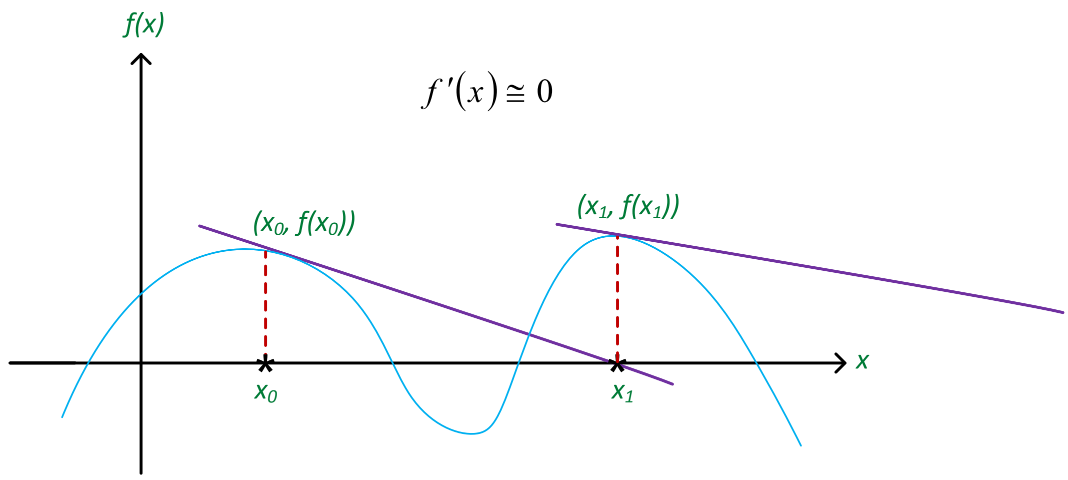
1.7 Pitfalls in Newton Raphson Method

Case – 1: Slow Convergence



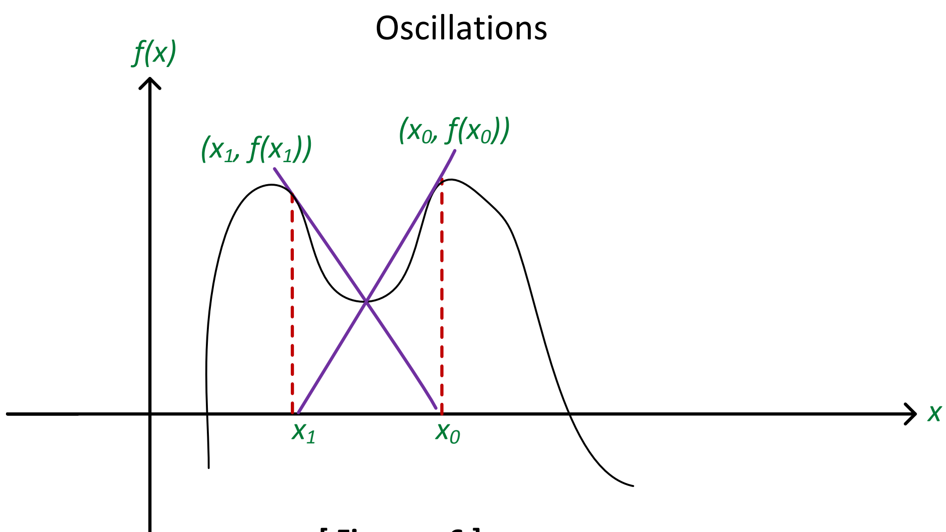
[Figure – 4]

Case – 2: Tangent is almost parallel ($f'(x) \cong 0$), leading to halting of the process



[Figure – 5]

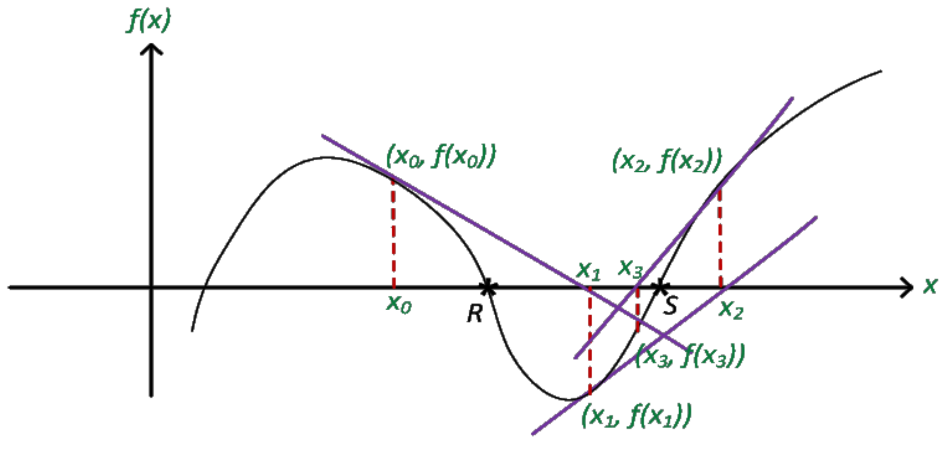
Case – 3: Oscillations



[Figure – 6]

if we look at the above figure tangent to the curve $y = f(x)$ at $x = x_0$ meets X axis at x_1, giving next approximation as x_1, and tangent to the curve at $x = x_1$ meets X axis at x_0. Thus the approximations shall keep on oscillating between two values x_0 and x_1 , setting an infinite loop, without leading anywhere.

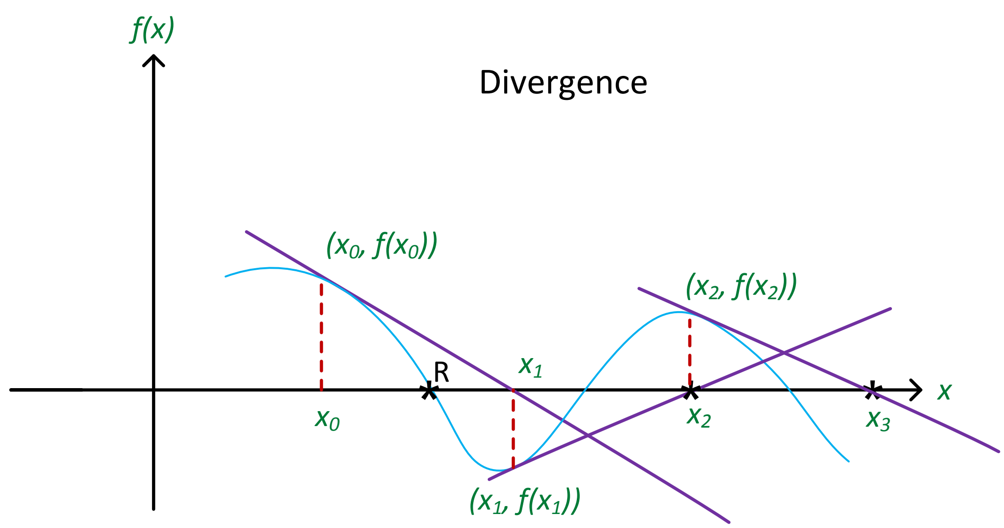
Case – 4: Convergence to another root (undesired root)



[Figure – 7]

If we begin with initial guess x_0 regarding it as sufficiently near the root, with an objective to estimate value of R with desired accuracy, but tangent to the curve at $x = x_0$, cuts X axis at x_1 and tangent to the curve at $x = x_1$, meets X axis at x_2, further away, process converging to the root S instead of R.

Case – 5: Divergence



[Figure – 8]

Here successive approximations keep on becoming larger and larger, thus the method diverges.

1.8 Advantages and Disadvantages of Newton Raphson Method

Advantages:

1. Only one initial guess is needed.
2. Very rapid convergence. It has quadratic convergence which implies, number of correct figures in the estimate is nearly doubled at each successive step.

   This can be understood as follows:

   Quadratic convergence means that if C is the root of $f(x)=0$, sequence of approximations follow an inequality given by $|C-x_{n+1}| \leq k|C-x_n|^2$ ... (6)

For convenience, let us take $k=1$. Also, let us suppose that the estimate x_n differs from exact root C by at most one unit in the pth decimal place. So, $|C-x_n| \leq 10^{-p}$

From inequality (6); $|C-x_{n+1}| \leq 10^{-2p}$

In other words, x_{n+1} differs from C by at most one unit in the 2pth decimal place, that is x_{n+1} has approximately twice as many correct digits as x_n ! Thus, significant digits get doubled.

Disadvantages:

1. If initial guess is not sufficiently near the root, it may diverge.
2. Function must be differentiable.
3. It requires two function evaluations per iteration, one of $f(x_n)$ and other that of $f'(x_n)$. In our iterative process, major chunk of time is expected to go in evaluation of function. All earlier methods required only one function evaluation per iteration. Thus, time consumed in performing one iteration of Newton Raphson Method would be double of that in any other method.
4. In case of multiple roots, pace of convergence slows down, if one knew the multiplicity m, then the formula $x_{n+1}=x_n-m\dfrac{f(x_n)}{f'(x_n)}$ needs to be worked with.

5. There can be many instances, when Newton Raphson may fail to converge, which are already explained and exemplified with diagram.

1.9 Extension of Newton Raphson Method for solving other problems

1) Systems of nonlinear equations:

In many physical and statistical problems, one needs to solve system of m nonlinear equations in n variables. A natural extension of this method is applied. Those who are interested can refer Cheney & kincail [3].

2) Complex roots:

All earlier methods are equipped to find real roots of an equation $f(x)=0$. The iterative formula of Newton Raphson method can be used to find complex roots as well, treating variable as complex number. That is, equation is $f(z)=0$ and iterative formula is given by $z_{n+1}=z_n-\dfrac{f(z_n)}{f'(z_n)}$, n = 0, 1, 2, …

Where $z_n = x_n + i\, y_n$

3) Fractals:

Fractals patterns can be easily created with the help of this method using complex variable. One can refer Cheney & kincail [13] for the same.

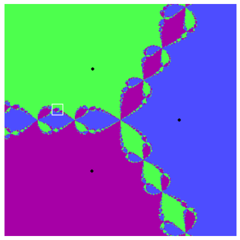
If we find nth roots of unity, that is try to solve the equation $f(z)=z^n-1=0$, then this equation has n roots. The basin of attraction is defined as the set of initial points z_0 which converge to the same root. It is found that these basins of attraction are disjoint from each other. For example, if we are finding roots of $z^3-1=0$, there would be three disjoint basins of attractions whereas if we are finding roots of an equation $z^4-1=0$, there would be four disjoint basins of attraction.

Usually points in each basin are assigned different color and points for which the method does not converage are left uncolored.

For n=3, n=4, following images are obtained respectively.

It may be worthwhile mentioning that fractals also arise from non-polynomials equations as well.

(i) n=3



(ii) n=4