

On the Newton-Raphson Method and Its Modifications

Sobre el método de Newton-Raphson y sus modificaciones

Juan Gabriel Triana Laverde¹

Abstract

The Newton-Raphson method, also known as Newton's method, is a method for finding successively better approximations to the roots of a real-valued function, starting with an initial guess, being useful even for generating fractals when we consider complex functions. It is a fast method, but convergence is not guaranteed, which is the reason why several modifications of that method have been proposed. Here we present some modifications of the Newton-Raphson method, and we study the convergence of those methods through cases.

Keywords: convergence, fractal, newton-Raphson method.

Resumen

El método de Newton-Raphson, también conocido como método de Newton, permite la aproximación sucesiva de raíces de funciones de valor real, partiendo de un valor inicial, siendo útil incluso para generar fractales al considerar funciones complejas. Es un método de convergencia rápida, pero no garantizada, razón por la cual se han propuesto diversas modificaciones del método. En este trabajo se presentan algunas modificaciones del método de Newton-Raphson y se estudia las falencias de convergencia de estos métodos a través de ejemplos.

Palabras clave: convergencia, fractal, método de Newton-Raphson.

Recepción: 26-oct-2022

Aceptación: 03-ene-2023

¹Universitaria Agustiniiana. Correo electrónico: juang.triana@uniagustiniana.edu.co

1 Introduction

Nonlinear equations and its applications have been widely studied; currently, we know that if the equation is not soluble via analytical methods, we can use numerical methods to solve it [10]. Thus, numerical methods for finding roots are continually being developed cf. [14].

The Newton-Raphson method, named after Isaac Newton (1643-1727) and Joseph Raphson (1648-1715), is a well-known method for finding roots given by the following recurrence relation

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

Hence considering an initial guess x_0 , the formula above can be used to calculate x_1 , x_2 and so on, we would stop once $|x_{k+1} - x_k| < \varepsilon$, for a given ε value cf. [4]. Due to the geometrical meaning the Newton-Raphson method is also known as the method of tangents [18]. Thomas Simpson (1710-1761) introduces an extension of this method for system of equations [5].

A Matlab code such that the function is typed by the user via keyboard and the derivative is calculated by Matlab (requires symbolic variables), considering as parameters: x_0 the initial guess, n the number of iterations and tol the tolerance, or the maximum error ε , is given as follows.

```
syms x;
f=input('Enter the function ');
%as x as variable
x0=input('Enter x0 ');
n=input('Enter the max.iterations');
tol=input('Enter the tolerance ');
fp=diff(f);
for k=1:n
    x1=x0-subs(f,x0)/subs(fp,x0);
    if(abs(x0-x1)< tol)
        fprintf('The root is x=%f ',x1)
        fprintf('with accuracy \n',tol)
        return
        fprintf('x%d = %f\n',k,x1);
    end
    x0=x1;
end
```

The above code is slower than a code with anonymous functions instead of symbolic variables [12]. Other programming languages can be used to implement the method, among them Python [11]. There are examples such that Newton-Raphson method, also known as Newton's method [20], jumps between two values; for instance, $f(x) = x^3 - 2x + 2$ with $x_0 = 0$ cf. [9]. Since $f'(x) = 3x^2 - 2$, we have

$$x_1 = 0 - \frac{(0)^3 - 2(0) + 2}{3(0)^2 - 2} = 1$$

$$x_2 = 1 - \frac{(1)^3 - 2(1) + 2}{3(1)^2 - 2} = 0$$

Thus, for any $n \geq 0$ we have $x_{2n} = 0$ and $x_{2n+1} = 1$. High sensitivity of the Newton-Raphson method on starting values is verified by $f(x) = \sin(x)$ in the following table, where x_0 is the initial guess and x^* is the value to which the method converges starting from x_0 .

x_0	1.4	1.5	1.6	1.7	1.8
x^*	π	-4π	10π	3π	2π

Table 1. Convergence of Newton-Raphson method for $f(x) = \sin(x)$ with several initial guesses.

Under certain conditions the convergence of Newton-Raphson method is quadratic [6], it can be verified that the Newton-Raphson method converges linearly near zeros having multiplicity greater than 1 cf. [3]; however, we note that the recurrence can become numerically unstable, and may not converge, if $f'(x_k)$ is close to zero. The following theorem is a global convergence theorem for Newton-Raphson method, proof can be found in [1].

Theorem 1. Let $f \in C^2([a, b])$ satisfy the following conditions:

1. $f(a)f(b) < 0$ and $f'(x) \neq 0$, for all $x \in [a, b]$.
2. Either $f''(x) \geq 0$ for all $x \in [a, b]$ or else $f''(x) \leq 0$ for all $x \in [a, b]$.
3. $\left| \frac{f(a)}{f'(a)} \right| < b - a$ and $\left| \frac{f(b)}{f'(b)} \right| < b - a$.

Then f has a unique zero $x^* \in (a, b)$ and the Newton-Raphson method converges to x^* for any initial guess $x_0 \in (a, b)$.

It is not easy to verify the hypotheses of the global convergence theorem, mainly due to the difficulty of determining a and b , thus in some cases for the function $f(x)$ there are a and b such that satisfies the conditions for convergence but $x_0 \notin (a, b)$.

To improve students' numerical and analytical thinking abilities, we present some limitations of the Newton-Raphson method. We aim to show several modified Newton-Raphson methods, generate a code for each method, and even compare them to study the numerical stability and convergence of these methods via examples.

2 Methods

Multiple roots bring some difficulties to Newton-Raphson method, thus Anthony Ralston and Philip Rabinowitz (1926-2006) introduce the following modification cf. [16]

$$x_{k+1} = x_k - \frac{f(x_k)f'(x_k)}{[f'(x_k)]^2 - f(x_k)f''(x_k)}$$

The formula above is known as modified Newton-Raphson method for multiple roots [2], in this way we avoid the singularity due to $f'(x_k) = 0$. Although it is a good option for multiple roots, this method requires more computational effort than the classical method. Ralston and Rabinowitz also introduce the modified method, also known as relaxed Newton's method [7], given by the following recurrence

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

Where m is the multiplicity of the root. It is easy to check that the classical Newton-Raphson method is the case $m = 1$. we may verify that $f(x) = x^3 - 2x + 2$ have just a real root, $r = -1.769292$ with multiplicity 1; however, by applying the modified Newton-Raphson method for multiple roots on $f(x) = x^3 - 2x + 2$ with initial guess $x_0 = 0$, we obtain $x_{10} = -1.769292$. On the other hand, by the relaxed Newton's method with $m = 2$ we have

a slow convergent method, with $m = 3$ we get $x_{2n} = 0.2416943$ and $x_{2n+1} = 2.7583057$ for any $n \geq 2$.

Recently, predictor-corrector strategies were used to improve Newton-Raphson method, among them we find the recurrence relation

$$x_{k+1} = x_k - \frac{2f(x_k)}{f'(x_k) + f'(x_{k+1}^*)} \quad \text{with}$$

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

The formula above is a third order of convergence method [23]. A modified method with convergence of order $1 + \sqrt{2}$ can be found in [13], that method is given by the recurrence relation

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(\frac{1}{2}[x_k + x_k^*])} \quad \text{with}$$

$$x_{k+1}^* = x_k - \frac{f(x_k)}{f'(\frac{1}{2}[x_{k-1} + x_{k-1}^*])}$$

where $x_0^* = x_0 - \frac{f(x_0)}{f'(x_0)}$. It is easy to verify that if $x_0^* = x_0$ we get the classical method.

By applying the predictor-corrector methods to $f(x) = x^3 - 2x + 2$ with initial guess $x_0 = 0$ we find the root $r = -1.769292$; but for this case, both predictor-corrector methods are slower than the modified method for multiple roots.

3 Results

In this section we test the presented methods with the following problem, use the Newton-Raphson method and its modifications to estimate a root of $f(x) = -x^4 + 6x^2 + 11$ employing an initial guess of $x_0 = 1$. On the other hand, we consider the family of complex polynomials $p_n(z) = z^n + 1$ to generate fractals as visual representations of the sensitivity on starting values.

By the classical method we have $x_{2n} = 1$ and $x_{2n+1} = -1$ for any $n \geq 0$. We obtain the same result with the modified Newton-Raphson method for multiple roots. For the relaxed Newton's method, given by the recurrence $x_{k+1} = x_k - m \frac{f(x_k)}{f'(x_k)}$, in the

following table we present the result obtained by applying several iterations taking $m = 2, 3$ and 4 .

Table 2. Relaxed Newton-Raphson method with $f(x) = -x^4 + 6x^2 + 11$ and initial guess $x_0 = 1$, considering $m = 2, 3$ and 4 .

	$m = 2$	$m = 3$	$m = 4$
x_1	-3	-5	-7
x_2	-2.55555	-1.83636	-0.490683
x_3	-2.97297	-23.62853	8.658129
\vdots	\vdots	\vdots	\vdots
x_{99997}	-2.73589	-3.00601	-123032917.3
x_{99998}	-2.73116	-2.32670	-2.98×10^{-8}
x_{99999}	-2.73589	-4.21986	123032917.3
x_{100000}	-2.73116	-1.82822	2.98×10^{-8}

For the relaxed Newton's method with $m = 2$ we have a slow convergent method. For $m = 3$ we have an unstable non-convergent method and for $m = 4$ we get a method such that, for large values n the result jumps between

$$x_{4n} = -2.98 \times 10^{-8} = -x_{4n+2}$$

$$x_{4n+1} = -1.23 \times 10^8 = -x_{4n+3}$$

The predictor-corrector methods diverge, in both methods the denominator of the recurrence formula in the calculation of x_1 is 0. Hence, for $f(x) = -x^4 + 6x^2 + 11$ with initial guess $x_0 = 1$, the Newton-Raphson method and the modifications considered in this paper do not have a good performance. However, we can use a modification of Newton-Raphson method with cubic convergence cf. [8], given by the recurrence relation

$$x_{k+1} = x_k - \frac{4f(x_k)}{f'(x_k) + 3f'\left(\frac{x_k + 2\rho_k}{3}\right)} \quad \text{with}$$

$$\rho_k = x_k - \frac{f(x_k)}{f'(x_k)}$$

It is easy to verify that if $\rho_k = x_k$ we get the classical Newton-Raphson method. As an interesting fact, by applying the above recurrence to $f(x) = -x^4 + 6x^2 + 11$ with initial guess $x_0 = 1$ we obtain the root $x_8 = 2.733521$.

In 1789, Arthur Cayley noted the difficulties in generalizing Newton's method to compute complex

roots of polynomials with degree $n > 2$ cf. [17]. Since the complex polynomial $p_n(z) = z^n + 1$ have n different roots [22], we assign a color to each root and we take a region of the complex plane to apply the Newton-Raphson method for each point, with the point as initial value; thus each point in the region takes the color of the root to which it converges. Hence, we get an interesting visual representation known as Newton's fractal [19].

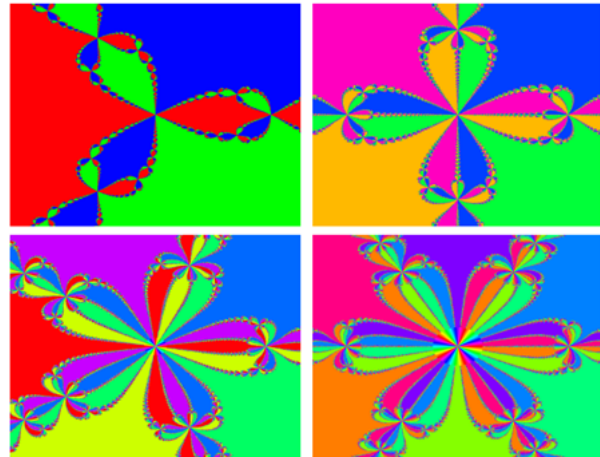


Figure 1. Newton's fractals for $n=3,4,5$ and 6 .

This idea can be extended to another kind of polynomials cf. [21] and methods. Here we use the relaxed Newton's method for generating fractals with family of polynomials $p_n(z) = z^n + 1$.

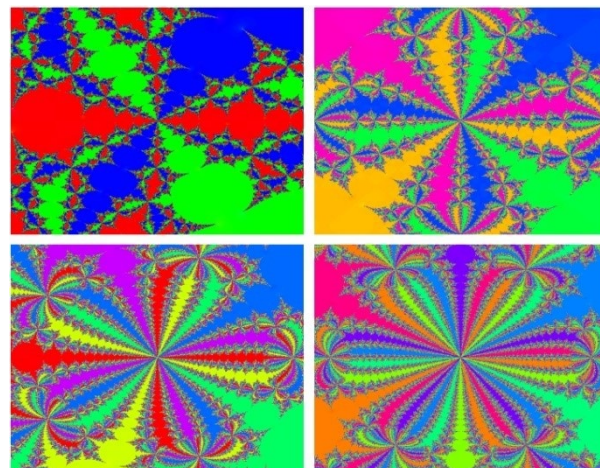


Figure 2. Fractal obtained by the relaxed Newton's method with $m = 2$, for $n = 3, 4, 5$ and 6 .

Fractals can be generated via the relaxed Newton's method with another families of polynomial cf. [15]. Since fractals are obtained through the instability of

the Newton-Raphson method, we may ignore the meaning of m in the relaxed Newton's method to consider non-integers values for generating fractals.

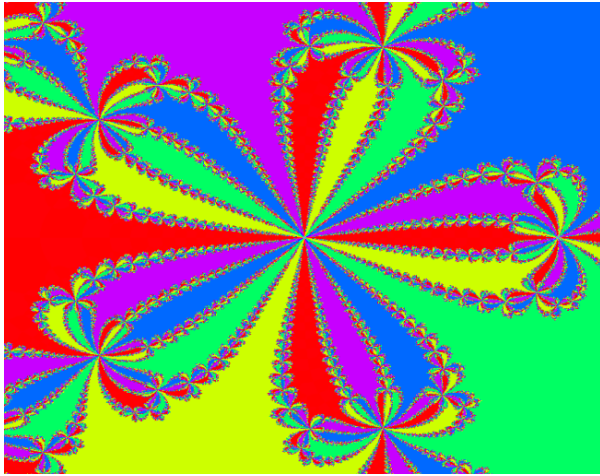


Figure 3. Fractal obtained by the relaxed Newton's method with $m = 1.5$ and $n = 5$.

4 Conclusions

To improve the student's skills in algorithmic thinking in numerical analysis courses, the implementation of several modified Newton-Raphson methods starting with a code of the classical Newton-Raphson method can be considered as an activity with the motivation of performing a comparison among them.

Given a function, to find a and b such that satisfies the global convergence theorem for Newton-Raphson method is a challenge that can be considered in a course for students of mathematics or students interested in theoretical numerical analysis.

Special cases such as the one considered in the section Results are helpful for preparing numerical analysis exams in platforms like Moodle since we can create several kinds of questions. For instance: multiple choice, we may ask which of the following modified Newton-Raphson methods converges; computed numeric, in which case we can ask for the result after performing a given number of iterations; justify a response, since we can ask to explain why the obtained result after performing some of the modified methods or compare the results obtained by several of those methods. Even programming questions, to implement one of the

modified Newton-Raphson methods from the code of the classical Newton-Raphson method.

The Newton fractal is a visual representation of the high sensitivity of the Newton-Raphson method on starting values, similar ideas through modified methods can be an interesting activity to consider in a numerical analysis course, here the teacher's role will be to guide the students on the use of programming languages for generating visual representations.

References

- [1] M. Allen and E. Isaacson. *Numerical analysis for applied science*. John Wiley and Sons, 2nd edition, 2019.
- [2] R. Burden and J. Faires. *Numerical analysis*, 9th edition, Cengage learning, 2011.
- [3] S. Chapra and R. Canale. *Numerical methods for engineers*, Mc Graw Hill, 7th edition, 2015.
- [4] W. Cheney and D. Kincaid. *Numerical mathematics and computing*, Cengage learning, 7th edition, 2013.
- [5] P. Deuffhard. A short history of Newton's method. *Doc. Math.*, extra volume ISMP, pp. 25-30, 2012.
- [6] A. Faul. *A concise introduction to numerical analysis*. CRC Press, 2016.
- [7] W. Gilbert. Generalizations of Newton's method. *Fractals*, vol. 9, no. 3, pp. 251-262, 2001. <https://doi.org/10.1142/S0218348X01000737>
- [8] A. Goudjo and L. Kouye. A new modification of Newton method with cubic convergence. *Adv. Pure Math*, vol. 11, no. 1, pp. 1-11, 2021. <https://doi.org/10.4236/apm.2021.111001>
- [9] T. Heister, L. Rebholz and F. Xue. *Numerical analysis*. De Gruyter, 2019.
- [10] M. King and N. Mody. *Numerical and statistical methods for bioengineering*. Cambridge University Press, 2010.

- [11] Q. Kong, T. Siau and A. Bayen. *Python programming and numerical methods*. Academic Press, 2021.
- [12] H. Lee. *Programming with Matlab*. SDC Publications, 2016.
- [13] T. Mc Dougall and S. Wotherspoon. A simple modification of Newton's method to achieve convergence order $1 + \sqrt{2}$. *Appl. Math. Lett*, vol. 29, pp. 20-25, 2014. <https://doi.org/10.1016/j.aml.2013.10.008>
- [14] A. Özyapici, Z. Sensoy and T. Karanfiller. Effective root-finding methods for nonlinear equations based on multiplicative calculi. *J. Math*, Article ID 8174610, pp. 1-7, 2016. <https://doi.org/10.1155/2016/8174610>
- [15] S. Plaza and N. Romero. Attracting cycles for the relaxed Newton's method. *J. Comput. Appl. Math*, vol. 235, pp. 3238-3244, 2011.
- [16] A. Ralston and P. Rabinowitz. *A first course in numerical analysis*. Mc Graw-Hill, 2nd edition, 1978.
- [17] G. Rubiano. *Iteración y fractales*. Universidad Nacional de Colombia, 2009.
- [18] S. Saha. *Numerical analysis with algorithms and programming*. CRC Press, 2016.
- [19] M. Sahari and I. Djellit. Fractal Newton basins. *Discrete Dyn. Nat.Soc*, pp. 1-16, 2006. <https://doi.org/10.1155/DDNS/2006/28756>
- [20] T. Sauer. *Numerical analysis*. Pearson, 3rd edition, 2018.
- [21] F. Vieira and R. Machado. The Newton fractal's Leonardo sequence study with Google Colab. *Int. Electron. J. Math. Educ*, vol. 15, no. 2, pp. 1-9, 2020. <https://doi.org/10.29333/iejme/6440>
- [22] J. Ward and R. Churchill. *Complex variables and applications*. Mc Graw Hill, 9th edition, 2009.
- [23] S. Weerakoon and T. Fernando. A variant of Newton's method with accelerated third-order convergence. *Appl. Math. Lett*, vol. 13, pp. 87-93, 2000. [https://doi.org/10.1016/S0893-9659\(00\)00100-2](https://doi.org/10.1016/S0893-9659(00)00100-2)