RESOLUÇÃO E IMPLEMENTAÇÃO DE MÉTODOS NUMÉRICOS PARA RESOLUÇÃO DE EQUAÇÕES NÃO LINEARES

Salete Maria Chalub Bandeira[1]

Simone Maria Chalub Bandeira Bezerra[2]

RESUMO: Apresenta-se, neste trabalho, o resultado parcial de uma pesquisa realizada nos últimos 7 anos, desenvolvida em conjunto com discentes dos Cursos de Bacharelado em Sistemas de informação, Licenciatura em Matemática e Bacharelado em Engenharia Civil, da Universidade Federal do Acre. Nessa pesquisa, verificou-se a importância da construção de um *Software* Matemático para auxiliar na resolução de equações não lineares, aplicando os Métodos Iterativos: Bissecção e Posição Falsa, assunto trabalhado na disciplina de Cálculo Numérico/ Métodos Numéricos. Deixou-se, para outra oportunidade, a abordagem dos métodos do Ponto Fixo, Newton-Raphson e Secante.
PALAVRAS-CHAVE: Equações não lineares, método da bissecção, método da posição falsa.

ABSTRACT: This work introduces partial result of a search which it has been carried ou in the early 7 years. It has been developed with Information Systems', Mathematics Licenciatiship's and civil Engineering's students in Universidade Federal do Acre. Searchers found out the importance of a mathematic software establishment to be used in solving of not linear equations troughout this present work. It has been applied Iterative Methods: Bissection and False Position. These subjects have been studied in Numeric Calculus/Numeric Methods.
Key words – Not linear equations, bissection methods, false position methods.

1 INTRODUÇÃO

Nas áreas mais diversas das ciências exatas ocorrem, com muita freqüência, situações que envolvem a resolução de uma equação do tipo f(x)=0. Conforme *Figura* 1a, que representa um dispositivo não linear, a função g que dá a tensão em função da corrente é não linear. Dados E e R e supondo conhecida a característica do dispositivo v = g(i), e desejando-se saber a corrente que vai fluir no circuito deve-se resolver a equação $E-Ri-g(i)=0$, aplicando-se a lei de *Kirchoff*. Esta situação, segundo Ruggiero (1996), tem o aspecto de um polinômio de terceiro grau, assunto que será discutido no presente estudo.

---

[1] Professora do Departamento de Matemática e Estatística da Universidade Federal do Acre. Mestre em Ciências da Computação: Matemática Aplicada. Endereço eletrônico para correspondência: saletechalub@gmail.com
[2] Professora do Departamento de Matemática e Estatística da Universidade Federal do Acre. Especialista em Matemática; mestranda do Curso Desenvolvimento Regional na Universidade Federal do Acre. Endereço eletrônico para correspondência: simonechalub@hotmail.com

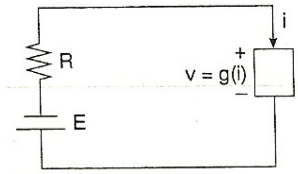


Figura 1a

Serão estudados métodos numéricos para encontrar um valor *x*, real que anule uma função $f(x)$, não linear. Segundo Hattori (1995), procura-se encontrar *x* que satisfaça a equação não linear $f(x) = 0$. Se $f(x)$ for um polinômio a equação é algébrica , caso não seja um polinômio a equação é transcendental.

Para Sperandio (2003), uma equação polinomial, algébrica ou transcendental é representada por $f(x) = 0$ **(1)**, onde *f* é uma função não linear a uma variável que pode ser uma função polinomial, algébrica ou transcendental. Entende-se por função transcendental aquela que envolve funções transcendentais como $\cos x, e^x, \ln x$, dentre várias outras. A equação $x^5 - 4x^3 + 10x - 100 = 0$ é um exemplo de equação polinomial, e a equação $x\text{tg}\, x - 1 = 0$, um exemplo de equação transcendental. Já a equação $1/(\sqrt{x^3 + 2}) - 20x = 0$ é um exemplo de equação algébrica.

As soluções da equação **(1)** são denominadas raízes da equação ou zeros da função *f*.

Em alguns casos de equações polinomiais, os valores de *x* que anulam *f*(*x*), podem ser reais ou complexos. Neste estudo, o interesse está voltado apenas para os zeros reais de *f*(*x*). Os zeros reais são representados pelas abscissas dos pontos onde uma curva intercepta o eixo $\overrightarrow{ox}$, conforme ilustrações a seguir.

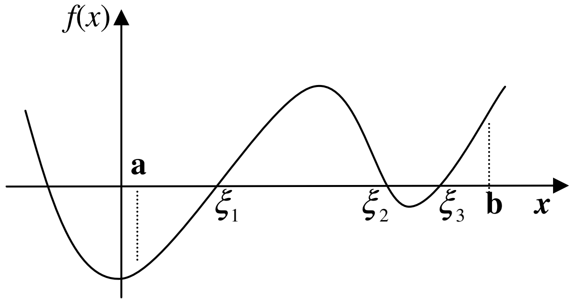


Figura 1b

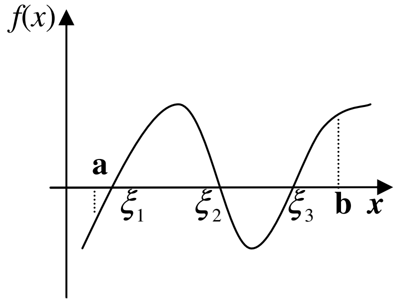


Figura 1c

Conforme Bandeira (1998) diz-se que o método é *iterativo* quando, partindo de uma aproximação inicial, é possível chegar-se a aproximações mais precisas que dependem, sempre, de valores anteriormente calculados.

Em geral, os métodos iterativos são formados por quatro partes:

> *Estimativa inicial*: parte-se de uma ou mais raízes como valores iniciais;
> *Atualização*: há uma forma que permite atualizar a solução aproximada;
> *Critério de parada*: deve haver um critério para estabelecer quando o processo iterativo deve parar;
> *Estimativa de exatidão*: processo associado ao critério de parada que permite estimar o erro cometido. (ROQUE, 2000).

A idéia central dos métodos numéricos para resolver funções não lineares é partir de uma aproximação inicial para a raiz e em seguida refinar essa aproximação através de um processo iterativo. Assim, o primeiro passo é o de isolar uma raiz, o que significa encontrar um intervalo $[a,b]$ que contém a raiz, e em seguida refinar, que consiste em, escolhidas aproximações iniciais no intervalo encontrado no primeiro passo, melhorá-las sucessivamente até obter-se uma aproximação para a raiz dentro de uma precisão ε prefixada.

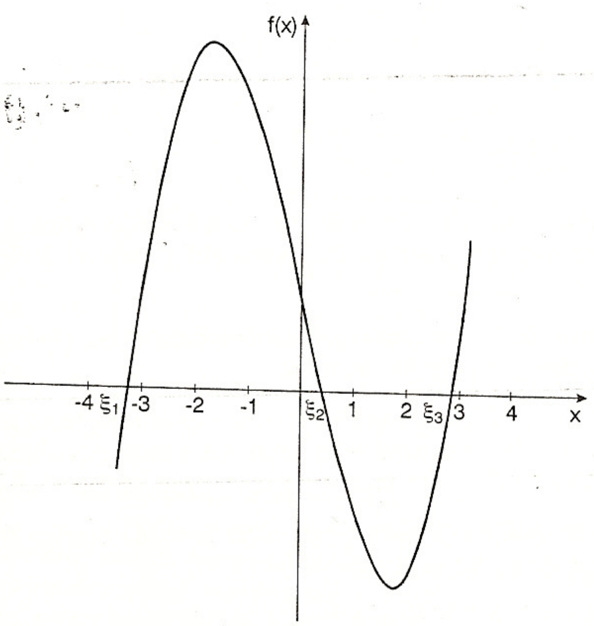
No isolamento das raízes, é feita uma análise *teórica e gráfica* da função $f(x)$ e o sucesso da fase seguinte, do refinamento depende da precisão desta análise.

Na *análise teórica* usamos freqüentemente o *Teorema* 1: Se,

> (a) $f(x)$ for diferenciável em [a, b];
> (b) $f(a)f(b)<0$ e
> (c)$f'(x)$ tiver sinal constante em [a, b], então existirá uma única raiz real no intervalo [a, b]. (HATTORI, 1995).

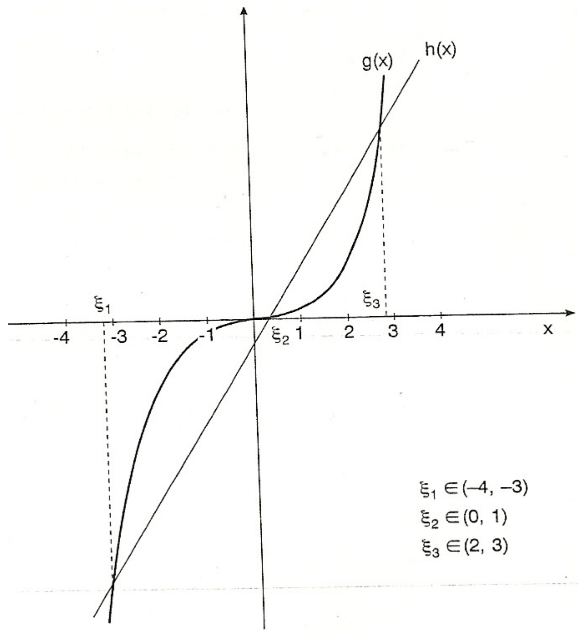
A *análise gráfica* da função $f(x)$ ou da equação $f(x) = 0$ é fundamental para obter-se boas aproximações para a raiz. Para tanto, diz Ruggiero (1996), é suficiente utilizar um dos seguintes processos:

1. Esboçar o gráfico da função $f(x)$ e localizar as abscissas dos pontos onde a curva intercepta o eixo $\vec{ox}$, observe *Figura* 3a;
2. A partir da equação $f(x)=0$, obter a equação equivalente $g(x)=h(x)$, esboçar os gráficos das funções $g(x)$ e $h(x)$ no mesmo eixo cartesiano e localizar os pontos x onde as duas curvas se interceptam, pois neste caso $f(\xi)=0 \Leftrightarrow g(\xi)=h(\xi)$, conforme *Figura* 3b;
3. Usar os programas que traçam gráficos de funções, disponíveis em algumas calculadoras ou *softwares* matemáticos.



$f(x) = x^3 - 9x + 3$

Figura 1d



$g(x) = x^3$ e $h(x) = 9x - 3$

Figura 1e

2 MATERIAL E MÉTODOS

Neste artigo, pretende-se analisar problemas de matemática, em particular solução de equações não lineares ($f(x)$=0). Serão estudados e implementados métodos numéricos iterativos para encontrar um valor de x, real, que anule a função $f(x)$ não linear. Para tal, os seguintes passos serão seguidos:

- Descrição e implementação dos métodos iterativos: Bissecção ou Método do Meio Intervalo e Posição Falsa, deixando-se para outra oportunidade, o método do Ponto Fixo, o método de Newton-Raphson e o método da Secante.
- Construção dos algoritmos e escolha da linguagem de programação a ser utilizada para implementar os métodos iterativos supracitados.

Tenciona-se elaborar um *software* matemático que possa servir como instrumento auxiliar nas aulas de “Cálculo Numérico”, ministradas nos cursos de Matemática, Engenharia Civil e Sistemas de Informação da Universidade Federal do Acre - UFAC.

3 DISCUSSÕES E RESULTADOS

Conforme acima mencionado, serão apresentados os Métodos Iterativos: Bissecção e Posição Falsa com seus respectivos algoritmos e telas do protótipo (*software* implementado em ambiente *C++ Builder* 5.0), descritos nas aulas dos cursos que têm em sua estrutura curricular a disciplina de Cálculo Numérico.

3.1 Método da Bissecção

Supondo que sejam satisfeitas as condições do *Teorema* 1. O objetivo é reduzir a amplitude do intervalo [a, b] que contém a raiz até se atingir a precisão requerida: $(b - a) < \varepsilon$, usando para isto a sucessiva divisão de [a,b] ao meio, observe-se a *Figura* 3a. A aproximação da raiz é calculada por $x_{k+1} = \dfrac{a_k + b_k}{2}, k = 0,1,2,\cdots$. Depois de obtido x_{k+1} é preciso verificar em qual dos subintervalos $[a_k, x_{k+1}]$ ou $[x_{k+1}, b_k]$ está a raiz procurada. Para isso, examina-se o sinal de $f(a_n)\ f(x_{n+1})$. Se for negativo, a raiz está no intervalo $[a_k, x_{k+1}]$; caso contrário, está em $[x_{k+1}, b_k]$. Sabendo em que intervalo está a raiz, faz-se $x_{k+1} = a_{k+1}$ ou $x_{k+1} = b_{k+1}$ conforme é descrito no *Algoritmo* 1.

Algoritmo 1: Método da Bissecção ou do Meio Intervalo

```
class bisseccao{
public:
   class funcao funcao1;
   float calc_raiz_bisseccao(float I0, float I1, float E1){
        int i, k;
        float a, b, x, fa, fb, fx, e;
        a = I0;
        b = I1;
        fa = funcao1.calc_funcao(a);
        //printf("\nF(a) = %f",fa);
        fb = funcao1.calc_funcao(b);
        //printf("\nF(b) = %f", fb);
```

```
    if (a>b || a==b || (fa*fb)>=0)
    {
        ShowMessage("Intervalo Inválido!");
        return 0;
     }
     e = E1;
    if (modulo(b-a)<e)
    {
            //printf("\nA raiz aproximada e: %f na iteracao: %d.",x,0);
            return 0;
    }
    for (k=1;  ; k++)
    {
            x = (a+b)/2;            /* Calculo da nova aproximação*/
            fa = funcao1.calc_funcao(a);
            fb = funcao1.calc_funcao(b);
            fx = funcao1.calc_funcao(x);
            if (modulo(b-a)<e || modulo(fx)<e)
            {
                return x;
            }
            if ((fx*fa)<0)
            {
                    if (x<a)
                    {
                            b = a;
                            a = x;
                    }
                    else
                    {
                            b = x;
                            //a = a;
                    }
            }
            else if ((fx*fb)<0)
            {
                    if (x<b)
                    {
                            a = x;
                            //b = b;
                    }
                    else
                    {
                            a = b;
                            b = x;
                    }
            }
    }

}
```

Segundo Ruggiero (1996), a estimativa do número de iterações é dada por $k > \dfrac{\log(b_0 - a_0) - \log(\varepsilon)}{\log 2}$. **(2)**

A convergência é lenta, observe exemplo a seguir:

$b_0 - a_0 = 3$ e $\varepsilon = 10^{-7}$, então aplicando **(2)** $k \geq 24{,}8 \Rightarrow k = 25$.

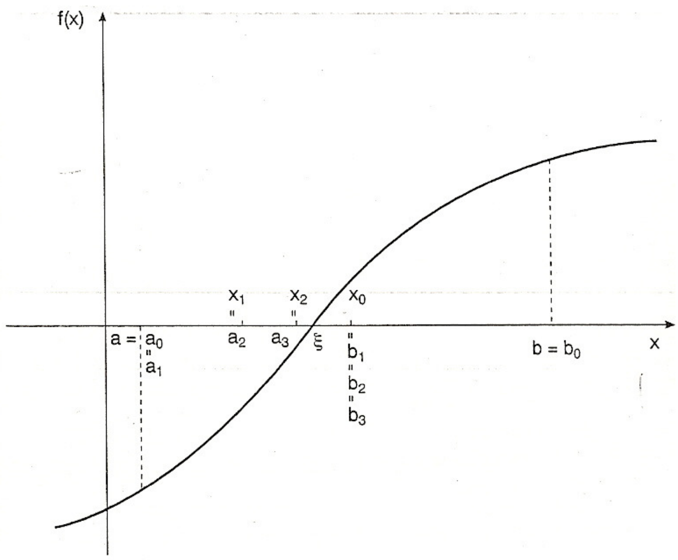


Figura 3a - Método da Bissecção

3.2 Método da Posição Falsa

Supondo que sejam satisfeitas as condições do *Teorema* 1. Em vez de tomar a média aritmética entre a e b, o método da posição falsa toma a média aritmética ponderada entre a e b com pesos |f(a)| e |f(b)|, respectivamente:

$x_{k+1} = \dfrac{a|f(b)| + b|f(a)|}{|f(b)| + |f(a)|} = \dfrac{af(b) - bf(a)}{f(b) - f(a)}$, visto que f(a) e f(b) têm sinais opostos e $k = 0,1,2,3,\ldots$. Graficamente, este ponto x_{k+1} é a intersecção entre o eixo das abscissas e a reta r(x) que passa por (a,f(a)) e (b,f(b)), segundo Ruggiero (1996), conforme *Figura* 3b e *Figura* 3c:

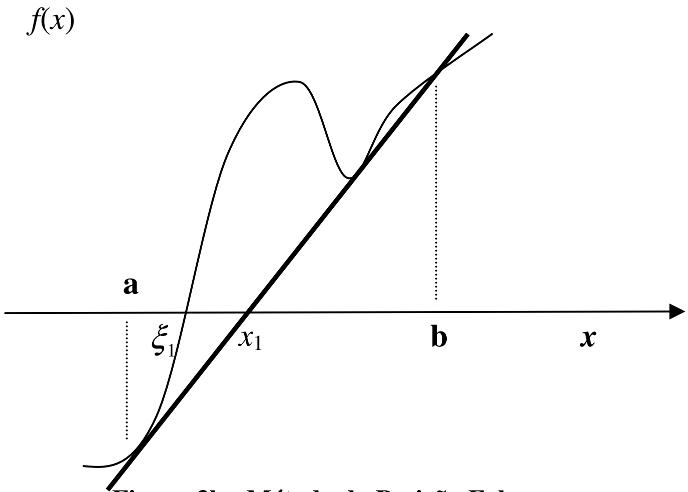


Figura 3b - Método da Posição Falsa

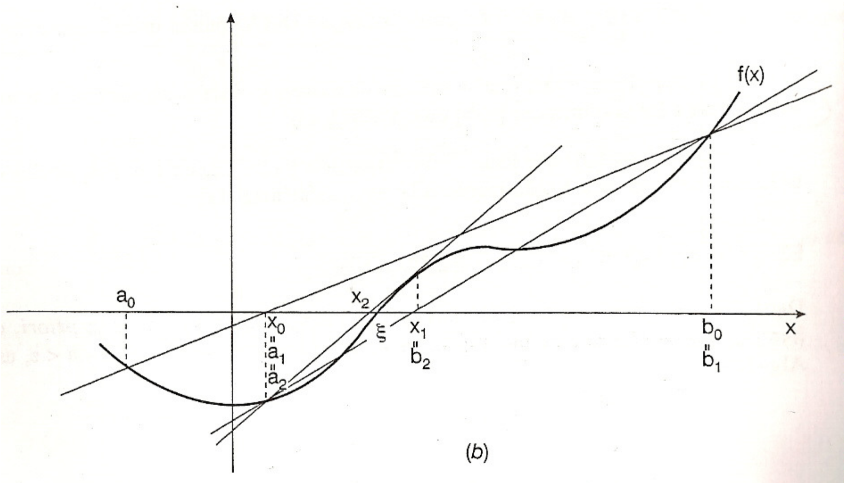


Figura 3c - Método da Posição Falsa

A reta $r(x)$ necessariamente cruza o eixo dos x em um ponto x_1 entre a e b, conseqüentemente dividindo o intervalo $[a,b]$ em dois subintervalos $[a,x_1]$ e $[x_1,b]$. Um dos subintervalos conterá a raiz. Na *Figura* 3b supracitada o intervalo considerado é $[a,x_1]$, visto que $f(a)<0$, $f(x_1)>0$, então $f(a)f(x_1)<0$, portanto pelo *Teorema*1 existe pelo menos uma raiz real no intervalo $[a,x_1]$, o processo se repete até pelo menos um dos critérios de parada ($b-a<\varepsilon$ ou $|f(x)|<\varepsilon$) ser satisfeito, observe *Figura* 3c. Observe-se, o *Algoritmo* 2, que descreve a situação acima.

Algoritmo 2: Método da Posição Falsa

```
class posicaofalsa{
 public:
   class funcao funcao1;
   float calc_raiz_posicaofalsa(float I0, float I1, float E1, float E2){
      int i, k;
      float a, b, x, fa, fb, fx, e1, e2;
      a = I0;
      b = I1;
      fa = funcao1.calc_funcao(a);
      fb = funcao1.calc_funcao(b);
      if (a>b || a==b || (fa*fb)>=0)
      {
         ShowMessage("ERRO: Intervalo Invalido!");
            return 0;
      }
      e1 = E1;
      e2 = E2;
      if (modulo(b-a)<e1)
      {
            return x;
      }
      for (k=1;  ; k++)
      {
            fa = funcao1.calc_funcao(a);
            fb = funcao1.calc_funcao(b);
            x = (a*fb - b*fa)/(fb - fa);          /* Calculo da nova aproximação */
            fx = funcao1.calc_funcao(x);
            if (modulo(b-a)<e2 || modulo(fx)<e2)
            {
                  return x;
            }
            if ((fx*fa)<0)
            {
                  if (x<a)
                  {
                        b = a;
                        a = x;
                  }
                  else
                  {
                        b = x;
                        //a = a;
                  }
            }
            else if ((fx*fb)<0)
            {
                  if (x<b)
                  {
                        a = x;
```

```
                        //b = b;
                }
                else
                {
                        a = b;
                        b = x;
                }
        }
    }
}
```

Na *Tela* 3 e *Tela* 4 do *software*, $f(x) = x^3 - 9x + 3$, $I = [2,3]$, aplicando o método da bissecção, encontra-se a raiz aproximada com quatro iterações, para $\varepsilon = 0{,}07$ e, aplicando o método da posição falsa, encontra-se a raiz aproximada com uma iteração a menos, para $\varepsilon = 0{,}09$.

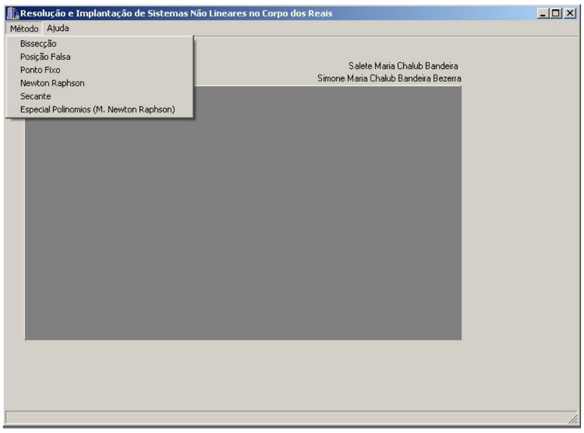
5 CONCLUSÕES

O estudo dos *Métodos* I*terativos* na Resolução de Equações Não Lineares é de suma importância para as áreas de conhecimento nas quais podem surgir problemas modelados na forma de equação polinomial, algébrica ou transcendental, sendo representada por $f(x) = 0$. Conforme já mencionado, o objetivo deste estudo é construir um *software* matemático utilizando os métodos iterativos intitulados de bissecção e posição falsa, conforme algoritmos 1 e 2 construídos para este fim.

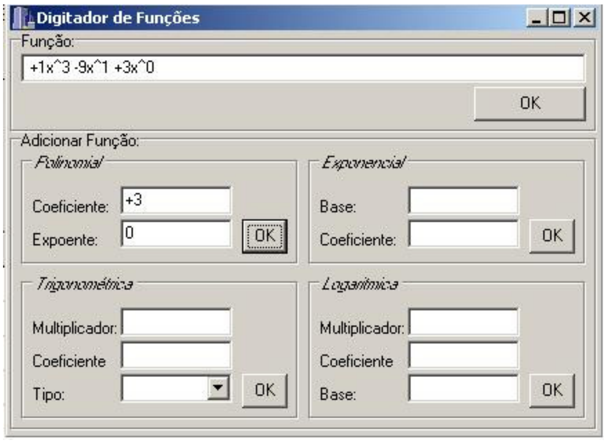
O método da bissecção, teoricamente, sempre converge, é sempre possível obter um intervalo que contém a raiz da equação em estudo, sendo que o comprimento deste intervalo final satisfaz a precisão requerida. As iterações não envolvem cálculos laboriosos. Na prática pode não convergir devido ao erro de arredondamento. A garantia teórica da convergência tem um custo: a convergência é lenta, pois se o intervalo inicial é tal que $b_0 - a_0 > \varepsilon$ e se ε for muito pequeno, o número de iterações tende a ser muito grande.

O método da posição falsa, assim como o método da bissecção, gera uma seqüência convergente, atendendo as condições do *Teorema* 1, porém um dos seus defeitos é o fato das aproximações geradas pela fórmula de iteração se aproximarem da raiz de um lado só.

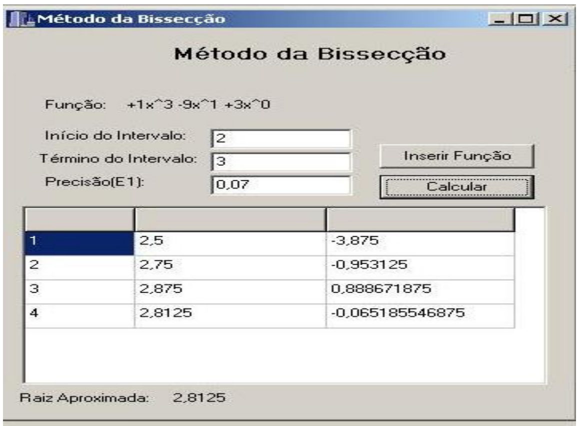
Observe-se, a seguir, as telas do *software* desenvolvido, resultado desta pesquisa.



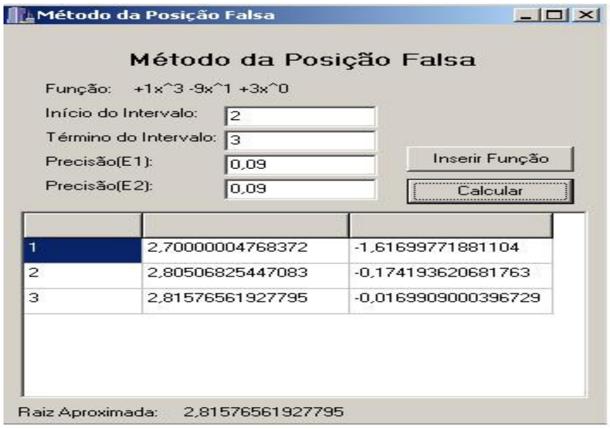
Tela 1 - Tela Inicial do Software



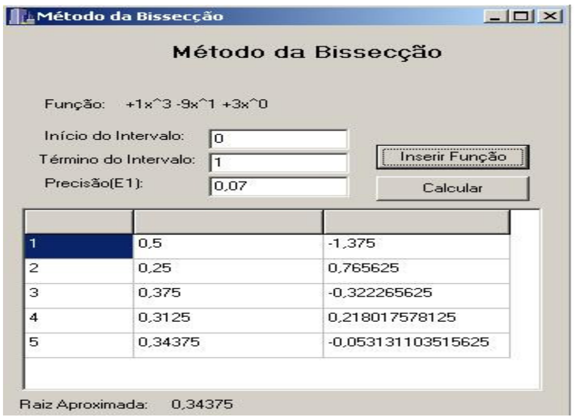
Tela 2 - Digitador de Funções



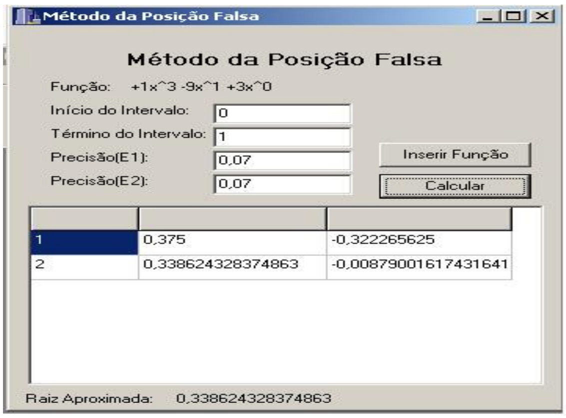
Tela 3 - Método da Bissecção



Tela 4 - Método da Posição Falsa



Tela 5 – Método da Bissecção



Tela 6 - Método da Posição Falsa

REFERÊNCIAS

BANDEIRA, S. M. C. *Solução Exata de Sistemas de Equações Lineares Utilizando a Aritmética Residual*. Campina Grande: UFPB, 1998.

HATTORI, Mário T. & QUEIROZ, Bruno C. N. *Métodos e Softwares Numéricos*. Paraíba, Universidade Federal da Paraíba, 1995.

ROQUE, Valdir L. *Introdução ao Cálculo Numérico – Um texto integrado com DERIVE©*. São Paulo: Editora Atlas, 2000.

RUGGIERO, Márcia A. Gomes & LOPES, Vera Lúcia da Rocha. *Cálculo Numérico – Aspectos Teóricos e Computacionais*. 2ª edição. São Paulo: Pearson Education do Brasil, 1996.

SPERANDIO, Décio *et. al*. *Cálculo Numérico*: Características Matemáticas e Computacionais dos Métodos Numéricos. São Paulo: Prentice Hall, 2003.