

CALCULADORA PARA MÉTODOS DE RESOLUÇÃO DE PROBLEMAS DE CÁLCULO NUMÉRICO

FONTENELE, Felipe. LANA, Mateus. TORRES, Pedro. SANTANA, Thiago.

*Ciência da Computação,
Universidade Federal de Ouro Preto
Ouro Preto, Minas Gerais*

Resumo: Este artigo descreve um estudo à respeito dos métodos estudados na disciplina BCC760 - Cálculo Numérico, onde foi implementado uma calculadora na qual é capaz de resolver todos os métodos vistos em sala de aula. Posteriormente foi realizada uma descrição dos métodos, a modelagem do sistema implementado e exemplos de uso para a calculadora. Entretanto, a calculadora possui algumas limitações que foram exemplificadas no artigo.

Palavras-chave: cálculo numérico, sistemas lineares, interpolação polinomial, integração numérica, raízes.

1. Introdução:

Nos estudos de Cálculo Numérico analisamos e aprendemos algoritmos que encontram resultados numéricos de problemas das mais diferentes áreas do conhecimento humano, modelados matematicamente. Em geral, os algoritmos de métodos numéricos se dividem em diretos, recursivos e iterativos. Os iterativos apresentam uma sucessão de passos que converge ou não para o valor aproximado da solução exata. É objetivo da análise numérica encontrar sucessões que aproximem os valores exatos com um número mínimo de operações elementares.

O objetivo deste trabalho é permitir que diversos problemas de cálculo numérico sejam solucionados computacionalmente através dos métodos estudados, proporcionando um ambiente no qual os algoritmos possam ser testados. Para isso, iremos desenvolver as implementações de quase todos os métodos vistos na disciplina, apresentando uma espécie de "calculadora de análise numérica" na qual um usuário fornece como

entrada as informações necessárias para solucionar um problema e o método a ser utilizado na resolução.

Sendo assim, foram codificados na linguagem C++, os algoritmos para os métodos numéricos de resolução de sistemas lineares: Eliminação de Gauss com pivotação parcial, Decomposição LU com pivotação parcial, Jacobi e Gauss-Seidel. Interpolação polinomial: Lagrange, Diferenças Divididas e Diferenças Finitas Ascendentes. Integração Numérica: Regra dos Trapézios, Primeira Regra de Simpson e Segunda Regra de Simpson. E para finalizar métodos para encontrar raízes: Bisseção, Falsa Posição e Newton-Raphson.

Este relatório será organizado da seguinte maneira: as sub-seções 2.1, 2.2, 2.3 e 2.4 presentes na seção 2 - Metodologia irão descrever os métodos de resolução de sistemas lineares, de Interpolação polinomial, de Integração numérica e métodos para encontrar raízes, respectivamente. A seção 3 apresenta os resultados computacionais de testes realizados com cada um dos métodos implementados. Por último, na seção 4 temos as conclusões do trabalho.

2. Metodologia:

Os métodos implementados seguiram um padrão de menu onde primeiramente se escolhe um tópico, isto é, resolver sistemas lineares, interpolação polinomial, integração numérica ou encontrar raízes. Após escolher o tópico, é escolhido o método que se deseja trabalhar como mostra a figura abaixo.

```
===== MENU PRINCIPAL =====
|| 1 - Sistemas Lineares. ||
|| 2 - Interpolacao Polinomial. ||
|| 3 - Integracao Numerica. ||
|| 4 - Raizes. ||
|| 5 - Sair. ||
===== MENU PRINCIPAL =====

Escolha: 4

===== MENU RAIZES =====
|| 1 - Metodo Bissecao. ||
|| 2 - Metodo Falsa Posicao. ||
|| 3 - Metodo Newton-Raphson. ||
|| 4 - Voltar. ||
===== MENU RAIZES =====

Escolha: _
```

Um exemplo de execução seria: 4 para escolher Raízes, 3 para escolher método de Newton-Raphson e após isso será pedido os dados do problema como exemplificado abaixo:

```
Informe o intervalo [a b] para calculo das raizes:
a = 0.1
b = 0.5
Informe a precisao desejada: 0.001

Raiz Newton-Raphson = 0.3806
Precisao = 0.00045
```

2.1 Métodos para resolução de sistemas lineares

Sistemas lineares é um conjunto de equações lineares, com m equações e n incógnitas que possuem a seguinte forma $a_1x_1 + a_2x_2 + \dots + a_nx_n = b$, em que a_1, a_2, \dots, a_n são os coeficientes reais e o termo independente é representado pelo número real b . A

solução de um sistema linear é a solução de todas as equações lineares. Existem muitas maneiras de resolver um sistema de equações lineares. Os métodos numéricos destinados a resolver sistemas lineares são divididos em dois grupos: os métodos diretos, que produzem a solução exata de um sistema, a menos de erros de arredondamento, depois de um número finito de operações aritméticas; e os métodos iterativos, que tratam-se de métodos nos quais a solução de um sistema linear $Ax = b$ é obtida como limite de uma sequência de aproximações sucessivas sendo dada uma aproximação inicial. Neste trabalho foram implementados dois métodos diretos: Gauss com pivotação parcial e Decomposição LU; e dois métodos iterativos: Jacobi e Gauss Seidel. Ao testar o programa, o usuário escolhe o método que deseja e fornece como entrada as informações necessárias ao mesmo. Posteriormente, ele insere a matriz de coeficientes e, em seguida, o vetor de termos independentes.

2.1.1 Gauss com pivotação parcial (Direto)

O método de Gauss consiste em operar transformações elementares sobre as equações de um sistema $Ax = b$ até que, depois de $n-1$ passos, se obtenha um sistema triangular superior, $Ux = c$, equivalente ao sistema dado, sistema esse que é resolvido por substituições retroativas. Nessas transformações são realizados os cálculos dos multiplicadores, que no método sem pivotação são os elementos da diagonal da matriz de coeficientes, essa operação é feita da seguinte maneira:

$$m_{ij} = a_{ij} \div a_{jj}$$

As transformações elementares propriamente ditas, são feitas nas linhas da matriz da seguinte forma:

$$L_i \leftarrow m_{ij} * L_j + L_i$$

A estratégia de pivoteamento consiste em:

1. No início da etapa k da etapa de eliminação, escolher para pivô o maior elemento, em módulo, dentre os coeficientes.

2. Trocar as linhas k e i se necessário

2.1.2 Decomposição LU (Direto)

Em muitas situações, é desejável resolver vários sistemas lineares nos quais a matriz dos coeficientes é a mesma. Nesses casos, é indicado resolver o sistema linear $Ax = b$ por uma técnica de decomposição da matriz A. Dentre as técnicas de decomposição mais utilizadas, destacamos a da decomposição LU.

Por esta técnica, uma matriz A é decomposta como o produto de duas matrizes L e U, sendo L uma matriz triangular inferior e U, uma matriz triangular superior, isto é:

$$A = L * U$$

Desta forma, podemos reescrever o sistema $Ax = b$ na seguinte forma:

$$Ax = (L * U) * x = L * (U * x) = b$$

Para aplicar a estratégia de pivoteamento parcial ao Método da Decomposição LU faz-se necessário armazenar um vetor de permutação P, que vai servir para armazenar as trocas feitas nas linhas, para que elas sejam realizadas também na hora de resolver os sistemas triangulares.

Fazendo-se $U * x = y$ podemos resolver o sistema $A * x = b$ em dois passos: Primeiramente, resolvemos o sistema triangular inferior $L * y = b$, obtendo y como solução. Em seguida, com a solução y obtida no passo anterior, resolvemos o sistema triangular superior $U * x = y$, obtendo x como solução. Em outras palavras, decomposmos a resolução de um sistema linear na resolução de dois sistemas triangulares: o primeiro, triangular inferior, que se resolve facilmente por substituições progressivas considerando elementos diagonais unitários) e o segundo, triangular superior, que se resolve por substituições retroativas.

Os fatores L e U podem ser obtidos utilizando-se a ideia básica do Método de Gauss. Assim, podemos concluir que $A = LU$, sendo:

1. U é a matriz triangular superior obtida ao final da fase de eliminação do método de Gauss;
2. L é uma matriz triangular inferior, cujos elementos da diagonal principal são unitários e abaixo de cada elemento diagonal $l_{kk} = 1$ encontram-se os multiplicadores da etapa k da fase de eliminação com sinal trocado.

Portanto o método consiste na seguinte sequência de passos:

1. Obter a fatoração LU da matriz A;
2. Fazer $U * x = y$;
3. Resolver o sistema triangular inferior $L * y = b$;
4. Obtida a solução y do sistema $L * y = b$, resolver o sistema triangular superior $U * x = y$.

2.1.3 Jacobi (Iterativo)

O método de Jacobi consiste na seguinte sequência de passos:

1. Escolher uma aproximação inicial $x^{(0)} = [x_1^{(0)} \ x_2^{(0)} \ \dots \ x_n^{(0)}]^T$ arbitrária;
2. Gerar aproximações sucessivas $x^{(k)}$ a partir de $x^{(k-1)}$ com base nas seguintes equações de

$$\begin{aligned} x_1^{(k)} &= \frac{b_1 - (a_{12}x_2^{(k-1)} + a_{13}x_3^{(k-1)} + \dots + a_{1n}x_n^{(k-1)})}{a_{11}} \\ x_2^{(k)} &= \frac{b_2 - (a_{21}x_1^{(k-1)} + a_{23}x_3^{(k-1)} + \dots + a_{2n}x_n^{(k-1)})}{a_{22}} \\ &\vdots \\ x_n^{(k)} &= \frac{b_n - (a_{n1}x_1^{(k-1)} + a_{n2}x_2^{(k-1)} + \dots + a_{nn-1}x_{n-1}^{(k-1)})}{a_{nn}} \end{aligned}$$

iteração:

Sinteticamente, cada componente $x_i^{(k)}$ é determinada com base na seguinte equação:

$$x_i^{(k)} = \frac{b_i - \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij}x_j^{(k-1)}}{a_{ii}} \quad \forall i = 1, 2, \dots, n$$

3. Interromper o processo quando um dos critérios abaixo for satisfeito:

- $\max_{1 \leq i \leq n} |x_i^{(k)} - x_i^{(k-1)}| < \varepsilon$, onde ε é a tolerância permitida;
- $k > \text{ITERMAX}$, onde ITERMAX é o número máximo de iterações.

2.1.4 Gauss Seidel

Este método difere do anterior apenas com relação às equações de iteração, as quais são:

$$\begin{aligned} x_1^{(k)} &= \frac{b_1 - (a_{12}x_2^{(k-1)} + a_{13}x_3^{(k-1)} + \dots + a_{1n}x_n^{(k-1)})}{a_{11}} \\ x_2^{(k)} &= \frac{b_2 - (a_{21}x_1^{(k)} + a_{23}x_3^{(k-1)} + \dots + a_{2n}x_n^{(k-1)})}{a_{22}} \\ x_3^{(k)} &= \frac{b_3 - (a_{31}x_1^{(k)} + a_{32}x_2^{(k)} + \dots + a_{3n}x_n^{(k-1)})}{a_{33}} \\ &\vdots \\ x_n^{(k)} &= \frac{b_n - (a_{n1}x_1^{(k)} + a_{n2}x_2^{(k)} + \dots + a_{n,n-1}x_{n-1}^{(k)})}{a_{nn}} \end{aligned}$$

Sinteticamente:

$$x_i^{(k)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^n a_{ij}x_j^{(k-1)}}{a_{ii}} \quad \forall i = 1, 2, \dots, n$$

2.2 Métodos de interpolação polinomial

Interpolarmos uma função, $y=f(x)$, em um intervalo finito (a,b) , consiste em substituí-la, ou aproximá-la, por outra função, $y=g(x)$. A necessidade de se utilizar esse procedimento ocorre, basicamente, quando a função é conhecida apenas por um conjunto de pontos x e $f(x)$ em determinados pontos. Ou quando a função é conhecida analiticamente, mas operações como a diferenciação e a integração são difíceis (ou mesmo impossíveis) de realizar. Os vários métodos para a determinação do polinômio interpolador têm em comum o conceito de que um polinômio nada mais é do que uma combinação linear de polinômios. O que difere um método do outro é a forma como este conceito é utilizado, ou seja, a maneira de como o polinômio interpolador é concebido.

2.2.1 Método de Lagrange

Neste método, o polinômio $y=L(x)$, que interpola uma função $y=f(x)$, em um conjunto de pontos (x_i, y_i) , $i=0,1, \dots, n$ é concebido da forma:

$L(x) = y_0.L_0(x) + y_1.L_1(x) + \dots + y_n.L_n(x)$, onde os $L_i(x)$, $i=0,1,2, \dots, n$.

Para determinar o valor de $L_i(x)$, utiliza-se a fórmula:

$$L_i(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}, \quad i=0, 1, \dots, n$$

2.2.2 Método de diferenças divididas

Neste método, sendo (x_i, y_i) , $i = 0,1, \dots, n$; um conjunto de pontos, com abscissas distintas, de uma função $y=f(x)$, tem-se que a diferença dividida de ordem k , é definida como:

$$D^k y_i = \frac{D^{k-1} y_{i+1} - D^{k-1} y_i}{x_{i+k} - x_i}, \quad \begin{cases} k=1, 2, \dots, n \\ i=0, 1, \dots, n-k \end{cases}$$

O polinômio interpolador com diferenças divididas é da forma:

$$p(x) = y_0 + (x-x_0).Dy_0 + (x-x_0)(x-x_1).D^2y_0 + \dots + (x-x_0)(x-x_1)\dots(x-x_{n-1}).D^ny_0$$

2.2.3 Método das diferenças finitas ascendentes

Sendo (x_i, y_i) , $i = 0, 1, \dots, n$; pontos de uma função, $y = f(x)$, tais que $x_{i+1} - x_i = h = \text{constante}$; $i = 0, 1, \dots, n-1$; A diferença finita ascendente de ordem k é definida como:

$$\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i, \quad \begin{cases} k=1, 2, \dots, n \\ i=0, 1, \dots, n-k \end{cases}$$

O polinômio interpolador com diferenças finitas ascendentes é da forma:

$$p(x_0 + h.z) = y_0 + z.\Delta y_0 + \frac{z(z-1)}{2!} \Delta^2 y_0 + \frac{z(z-1)(z-2)}{3!} \Delta^3 y_0 + \dots + \frac{z(z-1)\dots[z-(n-1)]}{n!} \Delta^n y_0$$

2.3 Métodos de integração numérica

O processo de integração permite a resolução de variados problemas que envolvem fenômenos físicos. Tal processo pode-se dar de forma analítica, isto é, a partir de manipulações algébricas e técnicas é possível obter a primitiva procurada. Algumas vezes, no entanto, é difícil

ou até mesmo impossível, encontrar uma primitiva. É assim então que os métodos analíticos para obtenção de uma primitiva dão lugar aos métodos numéricos para a obtenção de um valor por meio de uma aproximação.

Para chegar a esta aproximação, pode-se utilizar os métodos de Newton-Côtes, estes são, a Regra dos Trapézios, a 1ª Regra de Simpson e a 2ª Regra de Simpson, cada qual com a sua precisão para a aproximação requerida. Ambos os métodos foram implementados neste trabalho, sendo que, o programa não é capaz de receber a função da integral como parâmetro informado pelo o usuário. Desta forma caso o usuário deseje integrar uma função diferente da que se encontra no código, será necessário abrir o mesmo e alterá-la. Para testar os métodos o usuário deve escolher o método de integração desejado e informar os valores que delimitam a integral bem como o número de subintervalos.

2.3.1 Regra dos Trapézios

A ideia desse procedimento é dividir o intervalo $[a,b]$ em m subintervalos de mesmo espaçamento $h = b-a/m$ e aplicar cada subintervalo na função da integral a fim de obter-se valores de y para aplicar na fórmula da Regra dos Trapézios que é definida como:

$$I = h/2 * [y_0 + 2y_1 + 2y_2 + \dots + 2y_{n-1} + y_n]$$

Após essa aplicação, teremos o valor aproximado para a integral em questão.

2.3.2 Primeira Regra de Simpson

A ideia desse procedimento é dividir o intervalo $[a,b]$ em m subintervalos de mesmo espaçamento $h = b-a/m$ na qual a quantidade desses subintervalos deve ser par. Posteriormente deve-se aplicar cada subintervalo na função da integral a fim de obter-se valores de y para aplicar na fórmula da Primeira Regra de Simpson que é definida como:

$$I = h/3 * [y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \dots + 2y_{n-2} + 4y_{n-1} + y_n]$$

Após essa aplicação, teremos o valor aproximado para a integral em questão

2.3.2 Segunda Regra de Simpson

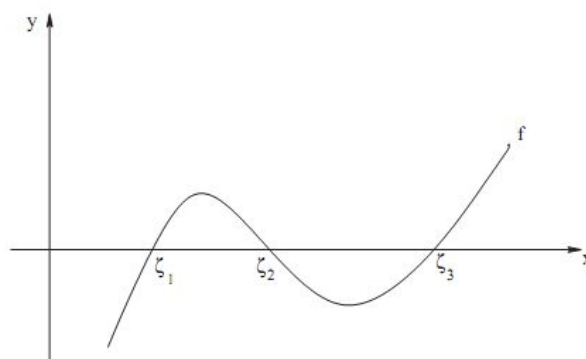
A ideia desse procedimento é dividir o intervalo $[a,b]$ em m subintervalos de mesmo espaçamento $h = b-a/m$ na qual a quantidade desses subintervalos deve ser múltiplo de 3. Posteriormente deve-se aplicar cada subintervalo na função da integral a fim de obter-se valores de y para aplicar na fórmula da Segunda Regra de Simpson que é definida como:

$$I = 3h/8 * [y_0 + 3y_1 + 3y_2 + 2y_3 + 3y_4 + 3y_5 + 2y_6 + \dots + 2y_{n-3} + 3y_{n-2} + 3y_{n-1} + y_n]$$

Após essa aplicação, teremos o valor aproximado para a integral em questão

2.4 Métodos para encontrar raízes

A necessidade de determinar valores $x =$ que satisfaçam a uma equação da forma $f(x) = 0$ ocorre com bastante frequência em uma grande variedade de problemas provenientes das Ciências e Engenharias. Estes valores são chamados de raízes da equação $f(x) = 0$ ou os zeros da função $y = f(x)$. Geometricamente, conforme figura abaixo, estes valores são os pontos de interseção de $y = f(x)$ com o eixo das abscissas.



Existem vários métodos para encontrar raízes, dentre os quais estudados em sala de aula foram: método da Bissecção, Falsa Posição e Newton-Raphson. Ambos os métodos foram implementados neste trabalho sendo que,

a primeira fase que consiste no isolamento de raízes (intervalos que contenham uma única raiz), o programa não é capaz de identificar por si próprio, sendo um parâmetro que o usuário deve informar, além do que, a função que se deseja encontrar as raízes já está contido no código, caso queira calcular raízes de outra função será necessário abrir o código e alterá-lo. Contudo, dentro de cada método, há a verificação se aquele intervalo contém somente um intervalo pelo teorema de Cauchy-Bolzano.

2.4.1 Método da Bisseção

Este método consiste em dividir o intervalo $[a, b]$, de forma iterativa, ao meio. Para verificar se a raiz está contida na primeira ou na segunda metade do intervalo inicial, é utilizado o teorema de Bolzano. Em seguida, o processo é repetido para aquela metade que contém a raiz de $f(x) = 0$, ou seja, aquela em que a função, $y = f(x)$, tem valores numéricos com sinais opostos nos seus extremos.

2.4.2 Método da Falsa Posição

Este método consiste em dividir, de forma iterativa, o intervalo $[a, b]$ no ponto em que a reta que passa por $[a, f(a)]$ e $[b, f(b)]$ intercepta o eixo das abscissas, sendo que em cada iteração é utilizado o Teorema de Cauchy-Bolzano para localizar o intervalo que contém a raiz.

2.4.3 Método de Newton-Raphson

Este método consiste em atribuir uma estimativa inicial $x_0 \in [a, b]$ para uma raiz de $f(x) = 0$ e criar uma sequência de estimativas, onde cada ponto é a interseção da reta tangente a $y = f(x)$

3. Resultados:

Os métodos implementados obtiveram resultados esperados e corretos para os testes realizados. Porém, a calculadora possui algumas limitações, onde tais

limitações serão resolvidas em trabalhos futuros para um perfeito funcionamento do sistema. Algumas limitações são: entrar com os parâmetros corretamente, pois o mesmo só funciona com os dados devidamente padronizados.

4. Discussão e conclusões:

Após a realização deste trabalho é nitidamente perceptível a evolução de uma visão mais crítica dos integrantes do grupo. Também foi percebido que os métodos numéricos possuem uma grande eficiência mediante uma precisão previamente definida.

Podemos observar que alguns métodos são mais eficientes perante outros e alguns problemas só serão resolvidos por meio destes métodos visto que o algebrismo nem sempre pode ser aplicável. Todavia, podem existir problemas para os quais esses métodos não se aplicam.

Termo de autorização

Nós, **Felipe Fontenele, Mateus Lana, Pedro Torres e Thiago Santana** na qualidade de autores do trabalho intitulado **CALCULADORA PARA MÉTODOS DE RESOLUÇÃO DE PROBLEMAS DE CÁLCULO NUMÉRICO**, autorizamos os professores do departamento de computação a divulgação total ou parcial deste trabalho para fins de estudo na disciplina de Cálculo Numérico.

Referências bibliográficas:

- [1] CAMPOS FILHO, Frederico Ferreira. Algoritmos Numéricos. 2. ed. Rio de Janeiro: Ltc, 2007.
- [2] L.C. Barroso, M.M.A. Barroso, F.F. Campos Filho, M.L.B. de Carvalho e M.L. Maia. "Cálculo Numérico (com aplicações)", Editora HARBRA, São Paulo, 2a edição, 1987.

[3] F.F. Campos Filho, “Algoritmos Numéricos”, Livros Técnicos Científicos Editora, 2a edição, Rio de Janeiro, 2007.

[4] E. Kreyzig, “Advanced Engineering Mathematics”, John Wiles & Sons Inc., 70th edition, New York, 1993.

[5] M.A.G. Ruggiero e V. L. R. Lopes, “Cálculo Numérico: Aspectos Teóricos e Computacionais”, Editora McGraw-Hill, São Paulo, 1988.

[6] FREITAS, Marcone. Métodos Numéricos, Notas de aulas, 2011. DECOM (UFOP).

[7] CONCEITO E APLICAÇÃO DE INTEGRAÇÃO NUMÉRICA PELOS MÉTODOS DE NEWTON-CÔTES: 1ª E 2ª REGRAS DE SIMPSON.

Disponível em:

<<http://www.ufjf.br/emem/files/2015/10/CONCEITO-E-APLICA%C3%87%C3%83O-DE-INTEGRA%C3%87%C3%83O-NUM%C3%89RICA-PELOS-M%C3%89TODOS-DE-NEWTON-C%C3%94TES-1%C2%AA-E-2%C2%AA-REGRAS-DE-SIMPSON.pdf>>