



SBA 
PRESS

MILTON BROWN DO COUTTO FILHO
VINÍCIUS BIAJONI BRAGA FLÔR

COMPUTAÇÃO NUMÉRICA

MÉTODOS E IMPLEMENTAÇÕES

Blucher



sociedade brasileira de
AUTOMÁTICA

Milton Brown Do Coutto Filho

Vinícius Biajoni Braga Flôr

COMPUTAÇÃO NUMÉRICA

Métodos e implementações

2ª edição

Revista e ampliada

Computação numérica: métodos e implementações

© 2017 Milton Brown Do Coutto Filho, Vinícius Biajoni Braga Flôr

1ª edição - Editora Elsevier

2ª edição - Editora Blucher, 2024

Editora Edgard Blücher Ltda.

Publisher Edgard Blücher

Editores Eduardo Blücher e Jonatas Eliakim

Coordenação editorial Andressa Lira

Produção editorial Kedma Marques

Preparação de texto Samira Panini

Diagramação Plínio Ricca

Revisão de texto Lidiane Pedroso Gonçalves

Capa Leandro Cunha

Imagem da capa iStockphoto

Blucher

Rua Pedroso Alvarenga, 1245, 4º andar

04531-934 – São Paulo – SP – Brasil

Tel.: 55 11 3078-5366

contato@blucher.com.br

www.blucher.com.br

Segundo o Novo Acordo Ortográfico, conforme 6. ed.
do *Vocabulário Ortográfico da Língua Portuguesa*,
Academia Brasileira de Letras, julho de 2021.

É proibida a reprodução total ou parcial por quaisquer
meios sem autorização escrita da editora.

Todos os direitos reservados pela Editora
Edgard Blücher Ltda.

Dados Internacionais de Catalogação na Publicação (CIP)
Angélica Ilacqua CRB-8/7057

Do Coutto Filho, Milton Brown

Computação numérica : métodos e implementações
/ Milton Brown Do Coutto Filho, Vinicius Biajoni Braga
Flôr. – 2. ed. -- São Paulo: Blucher, 2024.

352 p.

Bibliografia

ISBN 978-65-5506-606-7

1. Cálculos numéricos 2. Computação – Matemática
4. Análise numérica - Processamento de dados I. Título
II. Flôr, Vinicius Biajoni Braga

23-5624

CDD 518

Índices para catálogo sistemático:

1. Cálculos numéricos

Conteúdo

NOTA DOS AUTORES	11
1. INTRODUÇÃO	17
1.1 Fundamentos	17
1.2 Implementação	20
1.3 História	22
1.4 Reflexões	35
2. NOÇÕES SOBRE SISTEMAS DE NUMERAÇÃO	37
2.1 Introdução	37
2.2 Representação numérica	39
2.3 Sistemas decimal e binário	46
2.4 Representação em ponto flutuante	56
2.5 Noções sobre erros	59
2.6 Exercícios propostos	64
2.7 Programas computacionais	67

3. EQUAÇÕES ALGÉBRICAS NÃO LINEARES	71
3.1 Introdução	71
3.2 Valor inicial e término de processos iterativos de solução	73
3.3 Método da bisseção	77
3.4 Método da posição falsa	81
3.5 Método do ponto fixo	85
3.6 Método de Newton	89
3.7 Método da secante	92
3.8 Exercícios propostos	97
3.9 Programas computacionais	102
4. SISTEMAS DE EQUAÇÕES ALGÉBRICAS LINEARES	107
4.1 Introdução	107
4.2 Aspectos preliminares	109
4.3 Notação	111
4.4 Regra de Cramer	112
4.5 Método da eliminação de Gauss	114
4.6 Matrizes elementares	122
4.7 Fatoração da matriz de coeficientes	128
4.8 Solução pela fatoração LDU	132
4.9 Pivoteamento	137
4.10 Matrizes interessantes	141
4.11 Métodos iterativos	144
4.12 Exercícios propostos	152
4.13 Programas computacionais	162
5. SISTEMAS DE EQUAÇÕES ALGÉBRICAS NÃO LINEARES	167
5.1 Introdução	167
5.2 Método de Newton	167
5.3 Método de Newton modificado	172
5.4 Método de Newton aplicado a funções complexas	176

5.5 Exercícios propostos	178
5.6 Programas computacionais	183
6. INTERPOLAÇÃO	185
6.1 Introdução	185
6.2 Considerações iniciais	187
6.3 Forma padrão	189
6.4 Forma de Lagrange	191
6.5 Forma de Newton	194
6.6 Erro de interpolação	199
6.7 Interpolação por partes (<i>Splines</i>)	202
6.8 Exercícios propostos	203
6.9 Programas computacionais	211
7. AJUSTE DE CURVAS	215
7.1 Introdução	215
7.2 Método dos mínimos quadrados	216
7.3 Ajuste linear	219
7.4 Ajuste polinomial	221
7.5 Ajuste não polinomial	224
7.6 Ajuste com transformação de dados	227
7.7 Exercícios propostos	230
7.8 Programas computacionais	236
8. INTEGRAÇÃO	239
8.1 Introdução	239
8.2 Considerações iniciais	239
8.3 Regra dos trapézios	242
8.4 Regra 1/3 de Simpson	243
8.5 Regra 3/8 de Simpson	245
8.6 Estimativa de erros	250

8.7 Regras de integração repetidas	253
8.8 Exercícios propostos	259
8.9 Programas computacionais	269
9. EQUAÇÕES DIFERENCIAIS ORDINÁRIAS	275
9.1 Introdução	275
9.2 Aspectos gerais	281
9.3 Método de Euler	287
9.4 Método de Euler modificado	295
9.5 Método de Runge-Kutta	297
9.6 Exercícios propostos	300
9.7 Programas computacionais	303
APÊNDICE SCILAB	311
APÊNDICE PYTHON	335
REFERÊNCIAS	345
RESPOSTAS AOS EXERCÍCIOS PROPOSTOS	349

CAPÍTULO 1

Introdução

1.1 FUNDAMENTOS

Entende-se como método: uma forma de proceder que seja regular, coerente, organizada e conduza a um certo resultado. Em grande parte, os métodos de solução de problemas de Engenharia são classificados como analíticos ou numéricos. As soluções analíticas correspondem à obtenção de expressões matemáticas envolvendo parâmetros e variáveis associadas ao problema em estudo. Já as soluções que decorrem de métodos numéricos constituem-se dos valores numéricos obtidos por tais métodos para as referidas variáveis.

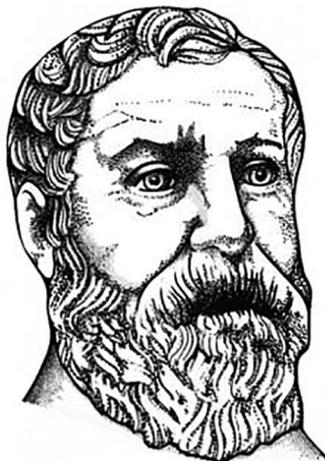


Figura 1.1 – Heron de Alexandria (10-70 d.C).

Fonte: https://pt.wikipedia.org/wiki/Heron_de_Alexandria

Para melhor entendimento sobre os dois tipos de solução, considere o problema bem simples de se obter a raiz positiva r da equação de segundo grau $x^2 - a = 0$, sendo a um número inteiro que não seja um quadrado perfeito. Por exemplo, adote 8 como tal número. A solução analítica desse problema seria $r = \sqrt{a}$. Particularmente para $a = 8$, $r = \sqrt{8}$, cujo valor obtido por uma calculadora é 2,828427125. Por outro lado, pode-se adotar um método numérico de solução, como o introduzido pelos babilônios e mais tarde formalizado por Heron de Alexandria (busto exibido na Figura 1.1), que se destaca por sua origem remota, simplicidade e eficiência. Heron foi um matemático egípcio com formação grega que viveu na segunda metade do século I d.C. Destacou-se por trabalhos de Matemática Aplicada, entre eles o intitulado *A métrica*, descoberto na cidade de Constantinopla, apenas em 1896 [Eves04]. O matemático notabilizou-se pela obtenção da fórmula da área de um triângulo, expressa em função da medida do comprimento de seus lados.

Na construção do método de Heron considera-se que, se $a = c \times d$, então será uma $r = (c + d)/2$ aproximação para \sqrt{a} , tão melhor quanto mais próximos estiverem os valores de c e d . Note que se o número fosse um quadrado perfeito, então c e d seriam números inteiros de igual valor. O problema se transforma então na obtenção de sucessivos valores para c e d , que ao se tornarem cada vez mais próximos entre si irão levar à solução do problema, isto é, ao valor de r . Assim, basta que se proceda da seguinte forma: inicialmente, escolha um valor para c (que possa ser considerado como próximo de r), e.g., $c_0 = 2$, para o caso em questão; calcule $d_0 = a/c_0 = 8/2 = 4$ e, em seguida, $c_1 = (c_0 + d_0)/2 = 3$; repetindo (agora com c_1) os cálculos anteriores, chega-se a $d_1 = a/c_1 = 8/3$ e $c_2 = (c_1 + d_1)/2 = 17/6$ e assim segue o processo para se encontrar a solução desejada. Concluída a quarta repetição, os valores obtidos para c_4 e d_4 são iguais a 2,828427125, valor este que se atribui à solução r . Em termos mais genéricos, pode-se formalizar o método por meio do seguinte processo iterativo:

$$x_{k+1} = \frac{1}{2} \left(x_k + \frac{a}{x_k} \right) \quad (1.1)$$

sendo k – índice de cada iteração (repetição).

Inicia-se o processo estabelecido na Equação (1.1) com um valor positivo x_0 arbitrariamente escolhido (e.g., a raiz quadrada do número que seja o quadrado perfeito imediatamente inferior) e segue-se até que seja obtida a solução com o grau de precisão desejado.

De um modo geral, métodos numéricos são utilizados para resolver problemas para os quais não há solução analítica ou esta apresenta elevado grau de dificuldade. Assim, deve-se destacar que o exemplo apresentado procurou apenas estabelecer minimamente a distinção entre solução analítica e solução numérica, sem compará-las. Apesar disso, dele se podem extrair alguns aspectos comuns a uma variedade de problemas que envolvem o processamento numérico e que merecem ser desde já abordados.

Usualmente, um problema de Engenharia e áreas afins surge pela observação de algum fenômeno físico e consequentemente do seu entendimento, que é obtido a partir de conhecimentos estabelecidos. Alcança-se a solução do problema após algumas etapas: enuncia-se o problema, descrevendo-o em palavras, da maneira mais clara e completa possível; segue-se uma fase de modelagem em que o problema assume forma matemática (com expressões que relacionam grandezas de interesse, denominadas variáveis, podendo também haver restrições, limites a cumprir etc.); propõem-se possíveis métodos de solução, considerando-se o grau de dificuldade que envolve o problema; aplica-se o método escolhido ao problema, obtendo-se dados e atribuindo-se valores a parâmetros (e.g., condições de início e término do processamento); realizam-se os cálculos que conduzem à solução almejada; procede-se à análise dos valores encontrados (por vezes, repetem-se algumas das etapas anteriores); por fim, chega-se à solução do problema.

Quando que se realizam tarefas que envolvem cálculos, podem surgir dúvidas sobre a confiabilidade das soluções obtidas, normalmente trazidas pelo uso indiscriminado de certos termos: exata, acurada, precisa, aproximada etc. No presente contexto, as soluções serão denominadas apenas de analíticas e de numéricas, ambas expressas por valores que apresentam determinado grau de precisão, estabelecido pelo número de algarismos significativos indicado. Tal número revela a incerteza que se tem sobre um valor calculado (mais algarismos significativos implicam em menor incerteza).

Quando necessário, a denominação *solução aproximada* será usada para indicar que certos valores obtidos foram fruto de alguma simplificação realizada no modelo adotado para o problema em estudo; por exemplo, para permitir a aplicação de algum método de solução, muitas vezes modelos não lineares dão lugar à sua forma linear. Nas demais situações, de antemão, ficará patente que a solução obtida será necessariamente aproximada, em face do grau de conhecimento disponível sobre determinado problema. Por exemplo, há funções submetidas à operação de integração para as quais obter as correspondentes funções primitivas nem sempre se consegue, sendo assim substituídas (aproximadas) por outras de fácil integração, como será visto adiante no Capítulo 8.

Para analisar a qualidade de um processo numérico de cálculo, além das possíveis fontes de erro (assunto abordado no próximo capítulo) presentes ao longo de todo o processo e que trazem incerteza aos valores obtidos como solução, deve-se considerar também aspectos relacionados à eficiência computacional, notadamente o tempo de execução do processo como um todo. Esse tempo deve ser tal que não comprometa aquilo que se possa considerar como tempo razoável para obtenção da solução de determinado problema. Por exemplo, no problema apresentado anteriormente, para encontrar a raiz positiva de $x^2 - a = 0$ por meio do método numérico adotado, com um certo (elevado) grau de precisão, apenas 4 ciclos de repetição (iterações) foram necessários, o que intuitivamente se pode considerar como uma convergência rápida para a solução. Para o mesmo grau de precisão, talvez outros métodos numéricos de solução pudessem requerer um número bem superior de iterações (ou até mesmo não

convergir), o que poderia comprometer o tempo para obtenção da solução, tornando-o inaceitável. Assim, quando o método numérico adotado for iterativo, os aspectos relevantes a considerar na sua avaliação são: grau de precisão desejado; solução inicial; garantia de convergência para a solução buscada; número máximo de iterações admitido; e tempo dispendido para se alcançar a solução. Os aspectos mencionados são dependentes da forma como os cálculos serão realizados e também da máquina utilizada, como a seguir se comenta.

1.2 IMPLEMENTAÇÃO

Uma vez escolhido o método de solução de um problema, torna-se necessário descrever as etapas de cálculo que levarão à solução desejada. Tal descrição recebe o nome de algoritmo (conjunto de regras e operações matemáticas bem definidas e estruturadas) que representa o plano de trabalho para a realização de uma tarefa. A origem da palavra algoritmo [Katz09] deriva do nome do matemático persa Al-Khwarizmi que viveu no período de 780 a 850 d.C. A civilização persa foi uma das mais expressivas civilizações da Antiguidade, estabelecida a leste da Mesopotâmia, em extenso planalto correspondente ao Irã.

Construído o algoritmo, deve-se definir a ferramenta que será adotada para implementá-lo, já que o trabalho a realizar pode ser cansativo, envolvendo cálculos volumosos, enfadonhos e repetitivos, requeridos em curto período de tempo. Buscar ferramentas que simplifiquem tarefas sempre ocupou cérebros privilegiados, desde os tempos mais remotos até os dias de hoje, como será visto sinteticamente na próxima seção deste capítulo. Assim, encontram-se na História da Computação [Fons07] narrativas de feitos marcantes de mentes geniais que realizaram cálculos matemáticos manualmente por meio de dispositivos simples (ábacos), que em seguida se tornaram implementos mecânicos (com engrenagens), para recentemente dar lugar aos computadores digitais (eletrônicos). A implementação de um algoritmo em computador, requer a tradução, em linguagem própria (código) ao ambiente de cálculo, daquilo que deve ser feito. Portanto, entende-se por programa de computador uma lista organizada de instruções (comandos), decorrentes do algoritmo construído, que ao serem executadas conduzam à solução procurada.

A comunicação entre usuário e computador se dá por meio de interfaces contidas em um sistema operacional [Tane99], sendo este responsável também pela gerência dos recursos computacionais disponíveis. Tais interfaces podem ser vistas como camadas (cascas, cápsulas) até que se chegue ao que se poderia chamar de comunicação direta ou de linguagem de máquina (instruções em código binário). O Unix (desenvolvido pela Bell Labs., na década de 1970) e o DOS (*Disk Operating System*, usado pela Microsoft) são exemplos de sistemas operacionais.

Usualmente, programas de computador utilizados nas aplicações científicas são escritos em linguagens características das camadas mais externas ou altas do sistema operacional (*e.g.*, Fortran, C etc.). Isso torna mais fácil e concisa a tarefa do calculista de

codificar as instruções contidas nos programas. Os métodos numéricos apresentados neste livro terão seus algoritmos implementados em **Scilab** (<https://www.scilab.org>) e **Python** (<https://www.python.org>), ambientes de computação numérica de alto nível [Leit09], [Mene19]. Uma introdução ao uso do **Scilab** e **Python** será apresentada ao fim do livro no Apêndice.

Tradicionalmente, as instruções codificadas em determinada linguagem de programação, elementos constitutivos dos programas de computador, podem ser declarativas, imperativas ou explicativas. As instruções *declarativas* estão entre as que ocupam as linhas iniciais de um programa, estabelecendo que variáveis dele farão parte (dando-lhes nomes) e a que tipo de dados estão associadas (inteiro, real, caractere e booleano são os mais comuns). A essas instruções preliminares, seguem-se as instruções *imperativas* que efetivam ações estabelecidas no algoritmo. A principal instrução imperativa é de atribuição de valores a variáveis, obtidos por meio da execução de operações matemáticas (soma, multiplicação, potenciação etc.), assim como pela aplicação de funções (trigonométricas, exponenciais, logarítmicas etc.). Instruções de controle são outras instruções imperativas destinadas a permitir a execução ou não de certas partes do programa, a depender de condições preestabelecidas e dos comandos do tipo *se-então-senão* e *enquanto* que destinam-se a esse propósito. Há também controles destinados à repetição de certas seções do programa (conhecidas por laços), muito úteis na programação de métodos numéricos iterativos. As instruções *explicativas* (comentários) são formas sintáticas (usualmente precedidas por marcadores que as identificam) inseridas ao longo do programa de modo a facilitar sua leitura.

A forma como os programas recebem, exibem e exteriorizam dados (por intermédio de comandos de entrada/saída) pode variar. Dados de entrada podem ser digitados diretamente via teclado do computador durante a execução do programa (suspendida para recebê-los) ou colhidos de arquivos preparados para tal. Não é boa prática embuti-los, considerando-os assim parte do algoritmo. Dados de saída (resultados) são apresentados na tela do monitor do computador, impressos em dispositivos periféricos próprios ou armazenados em arquivos destinados a esse fim. Além de expressos numericamente, os resultados de um programa podem ser transformados em figuras e gráficos.

Uma ideia interessante para a elaboração de programas que seriam constituídos por uma longa lista de instruções consiste na modularização, *i.e.*, na criação de porções de código (subprogramas ou funções) responsáveis pela realização de tarefas específicas, parte do todo. A criação de módulos permite que sejam escritos e testados de forma independente, trazendo assim ao software maior confiabilidade, legibilidade e flexibilidade para a manutenção.

Este livro se propõe a fazer uma introdução à Computação Numérica. Entretanto, embora distante do escopo aqui apresentado, vale observar que há certos problemas para os quais podem ser construídos algoritmos numéricos em que a realização de cálculos (que não apresentem ordem de precedência) ocorre de forma não sequencial.

Tal forma de proceder se conhece por processamento paralelo e visa à obtenção de alto desempenho computacional [Hage11].

1.3 HISTÓRIA

Na forma pictórica da Figura 1.2, a História pode ser vista como a Ciência que estuda a ação humana no tempo e espaço, procedendo à análise de eventos ocorridos no passado. No que diz respeito à capacidade de realizar cálculos, há feitos marcantes deixados na história que alicerçaram a Computação Científica recente, merecedores de registros indelévels nesse campo do conhecimento. Aqui, inicialmente, busca-se fazer uma referência sintética às ferramentas de cálculo concebidas nos primeiros tempos, como dispositivos simples de operação manual, que contabilizam, embora incertos, dois milhares de anos a.C. Em seguida, registra-se a existência de máquinas mecânicas com engenhosas engrenagens, para se chegar então ao advento das máquinas eletrônicas digitais da atualidade. A rapidez com que os computadores se desenvolveram nos últimos 50 anos parece tornar tão remota suas origens que, para muitos, resta difícil a percepção de que outras máquinas possam ter existido como suas precursoras. Para mais detalhes sobre a História da Computação, encontra-se em [Ifra01] uma compilação dos principais feitos até então alcançados.



Figura 1.2 – História: ação no tempo e espaço.

Fonte: <https://escolaeducacao.com.br/wp-content/uploads/2018/07/conceitos-de-historia.jpg>

Ábaco

Acredita-se que circundando 2500 a.C tenham surgido na Mesopotâmia os primeiros quadros para contagem, precursores dos ábacos, que podem ser considerados como uma extensão do ato natural de se contar nos dedos. Relatos referem-se a um quadro de contar como uma placa de madeira ou pedra lisa, coberta por areia ou pó,

em que linhas eram traçadas (com os próprios dedos ou hastes pontiagudas) e entre estas (os espaços entre linhas poderiam ser destinados às dezenas, centenas etc.) colocados pequenos seixos (pedrinhas suavemente erodidas), representando as quantidades desejadas. Presente em diversas culturas (babilônios, egípcios, gregos), o quadro para contagem mais antigo, que sobreviveu ao tempo, foi encontrado em 1846 na ilha grega de Salamina (usado pelos babilônios, cerca de 300 a.C, de mármore branco, media 149 cm de comprimento, 75 cm de largura e 4,5 cm de espessura).

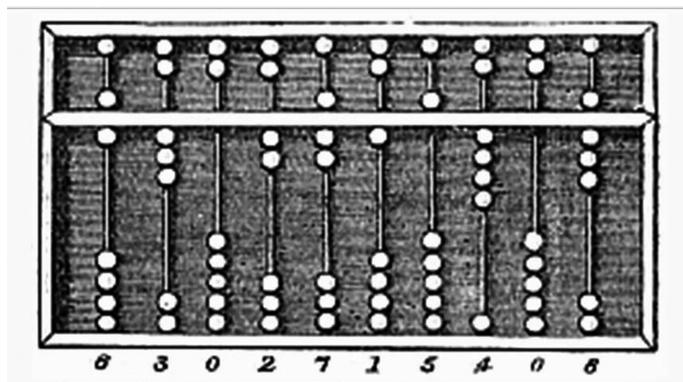


Figura 1.3 – Exemplo de ábaco.

Fonte: https://i.ytimg.com/vi/7D_L4l6ybMc/maxresdefault.jpg

Na Roma e Grécia antigas se calculava com o ábaco, dispositivo obtido pela necessidade de portabilidade, a qual era inexistente nos antecessores quadros de contagem. Nesses ábacos se usavam pequenas bolas feitas de mármore (cada uma conhecida por *calculus*) que deslizavam em fendas de uma placa de bronze. Depois, chineses (século I), japoneses e russos (século XVII) o modificaram. Operado manualmente, o ábaco-compõe-se de uma moldura retangular na qual são fixados fios paralelos pelos quais deslizam pequenas peças correspondentes aos elementos de contagem, como em um colar, exemplificadas na Figura 1.3, em que está representado o número 6302715408. Deixa-se ao leitor buscar entender a representação do referido número.

Régua de cálculo

Com a descoberta dos logaritmos (do grego *logos* – razão e *aritos* – números) em 1614 pelo escocês John Napier, retratado na Figura 1.4, conhecido por Neper, foi possível criar dispositivos (os bastões de Napier) que auxiliassem à multiplicação. Em 1633, o sacerdote inglês William Oughtred (Figura 1.5) representou os logaritmos de Napier por segmentos marcados com traços em régua (de madeira, marfim etc.). Dessa forma, as operações de multiplicação e divisão corresponderiam à adição e subtração dos comprimentos de tais segmentos.

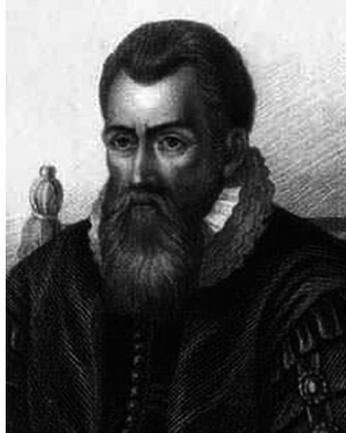


Figura 1.4 – John Napier (1550-1617).

Fonte: <https://mathshistory.st-andrews.ac.uk/Biographies/Napier/Napier.jpeg>

Lembre-se que o logaritmo do produto de dois números a e b iguala-se à soma dos logaritmos desses números e a divisão à diferença, ou seja:

$$\ln(a \times b) = \ln(a) + \ln(b); \quad \ln(a / b) = \ln(a) - \ln(b)$$

Tal dispositivo, ilustrado na Figura 1.6, tornou-se conhecido como Régua de Cálculo, tendo sido usado até a década de 1970, quando então foi substituído pela calculadora eletrônica de bolso.



Figura 1.5 – William Oughtred (1575-1660).

Fonte: <https://mathshistory.st-andrews.ac.uk/Biographies/Oughtred/Oughtred.jpeg>

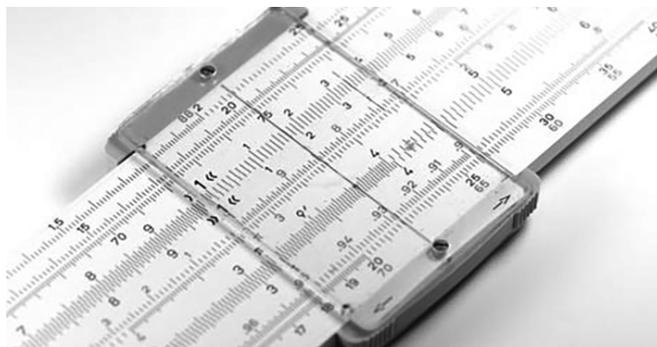


Figura 1.6 – Régua de Cálculo.

Fonte: <https://catarinabaptista.files.wordpress.com/2010/06/reguacalc1.jpg>

A seguir, na Figura 1.7, são exibidas duas escalas de uma Régua de Cálculo, em que se ilustra a operação de multiplicação de um valor qualquer da escala superior por 1,5, resultando no valor que se lê na escala inferior. Para tal, bastou que as duas régua que contém as divisões que representam valores na escala logarítmica fossem posicionadas adequadamente: o início da escala superior alinhado com 1,5 da escala inferior. Por exemplo, o valor 5 (da escala superior) ao ser multiplicado por 1,5 resulta no valor 7,5 (lido na escala inferior). Isso corresponde à soma do comprimento do segmento registrado na escala superior que vale $\ln(5)$ com o da escala inferior de valor igual a $\ln(1,5)$, totalizando portanto:

$$\ln(7,5) = \ln(5) + \ln(1,5) = \ln(5 \times 1,5)$$

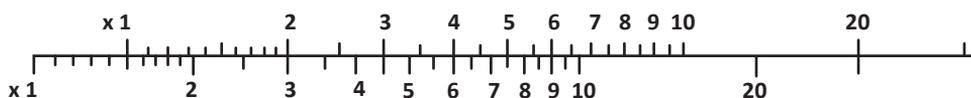


Figura 1.7 – Duas escalas logarítmicas de uma Régua de Cálculo.

Máquina de Pascal

O francês Blaise Pascal (Figura 1.8), em 1642, aos 19 anos, construiu uma máquina de calcular, para auxiliar no seu trabalho de contabilidade, baseada em dois conjuntos de discos interligados por engrenagens (rodas com dez dentes): um deles, para introdução de dados, e outro, destinado ao armazenamento de resultados. Utilizando o sistema decimal, quando um disco ultrapassava o valor 9, retornava ao valor 0 e aumentava de uma unidade o valor contido no disco imediatamente superior. A **Máquina de Pascal**, Pascalina (como se tornou conhecida), foi a primeira calculadora mecânica que se tem

registro, capaz de somar e subtrair. O sucesso comercial das Pascalinas não foi satisfatório devido ao seu elevado custo e baixa confiabilidade, verificada nas cinquenta versões produzidas. A Figura 1.9 corresponde a uma Pascalina.

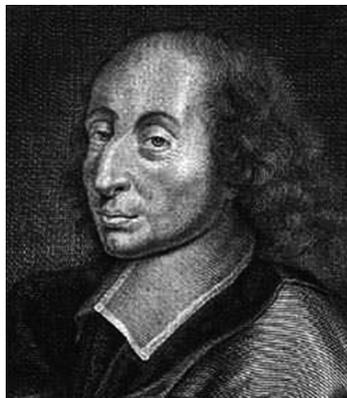


Figura 1.8 – Blaise Pascal (1623-1662).

Fonte: https://mathshistory.st-andrews.ac.uk/Biographies/Pascal/Pascal_6.jpeg



Figura 1.9 – Pascalina de 6 dígitos.

Fonte: https://upload.wikimedia.org/wikipedia/commons/thumb/8/80/Arts_et_Metiers_Pascaline_dsc03869.jpg/450px-Arts_et_Metiers_Pascaline_dsc03869.jpg

Calculadora de Leibniz

O alemão Gottfried Wilhelm von Leibniz (Figura 1.10), em 1671, introduziu o conceito de multiplicar e dividir, por meio de adições e subtrações sucessivas. Leibniz criou uma máquina de calcular (Figura 1.11) superior à Pascalina, pois era capaz de mecanicamente realizar as quatro operações aritméticas. Em 1694, tal máquina foi

construída, mas seu difícil uso e frequentes erros não permitiram que prosperasse. Com base na máquina de Leibniz, em 1820, Charles Xavier Thomas (1785-1870), projetou e construiu uma máquina de calcular também capaz de operar a soma, subtração, multiplicação e divisão (esta, possível com o auxílio do seu operador), atribuindo-lhe o nome de *Arithmometer*. Tal máquina foi a primeira calculadora que logrou êxito de produção, alcançando cinco mil unidades, de diversos modelos, no período entre 1851 e 1914.



Figura 1.10 – G. W. von Leibniz (1646-1716).

Fonte: <https://mathshistory.st-andrews.ac.uk/Biographies/Leibniz/Leibniz.jpeg>

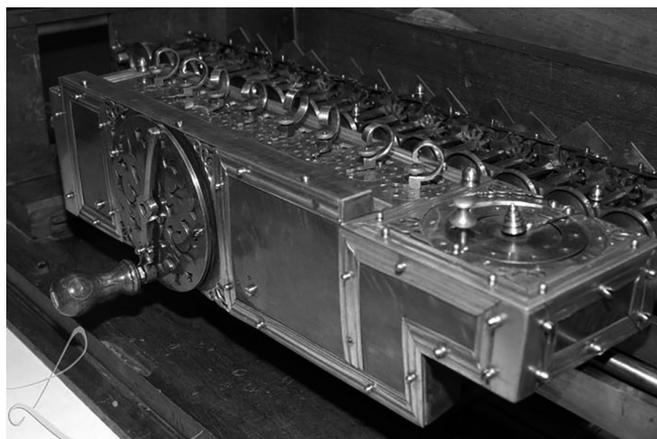


Figura 1.11 – Máquina de Leibniz.

Fonte: <https://assets.sutori.com/user-uploads/image/dd9dc17d-7a21-4d85-9950-a688383e4f2f/b7783bbd06d6fb-18cb565278862f595f.jpeg>

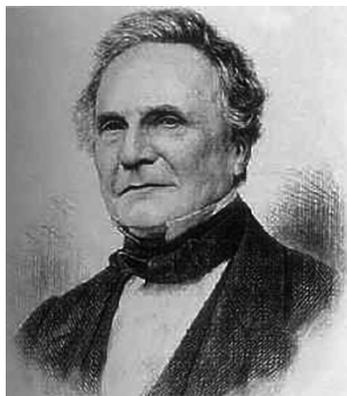


Figura 1.12 – Charles Babbage (1792-1871).

Fonte: https://mathshistory.st-andrews.ac.uk/Biographies/Babbage/Babbage_4.jpeg

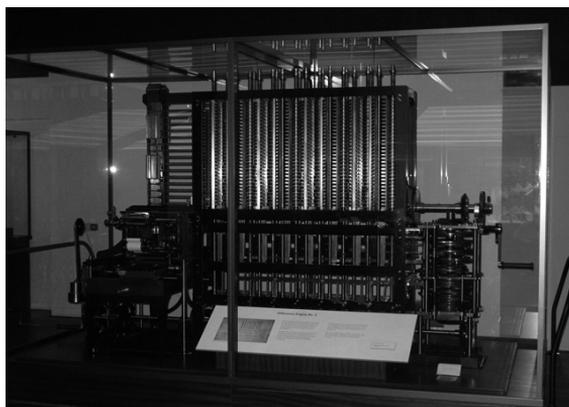


Figura 1.13 – Máquina de Babbage para diferenças.

Fonte: <https://img.ibxk.com.br/2011/12/materias/36169350115135250.jpg?ims=704x>

Calculadora analítica

Entre 1802 e 1822, o inglês Charles Babbage (Figura 1.12), com base em diversos estudos (notadamente os de J. H. Mueller e Joseph Marie Jacquard), projetou uma máquina de diferenças (Figura 1.13). A partir de 1801, o francês J. M. Jacquard (Figura 1.14) sugeriu que o controle de teares se desse por intermédio de cartões perfurados (Figura 1.15) que conteriam comandos necessários à automatização da tecelagem de padrões complicados. O tecelão Jacquard não poderia imaginar que seu invento para a indústria têxtil teria adiante importante influência na programação de computadores.

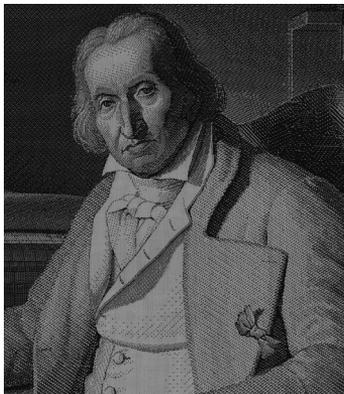


Figura 1.14 – J. M. Jacquard (1752-1834).

Fonte: https://pt.wikipedia.org/wiki/Joseph-Marie_Jacquard



Figura 1.15 – Tear mecânico de Jacquard.

Fonte: https://pt.wikipedia.org/wiki/Joseph-Marie_Jacquard

A programação por cartões perfurados foi proposta por Basile Bouchon, Jean-Baptiste Falcon e Jacques Vaucanson, entre 1725 e 1749. Em seguida, J. H. Muller, em 1786, planejou a construção de uma máquina diferencial, com o intuito de construir tabelas matemáticas para funções polinomiais que poderiam ser úteis para se obter valores aproximados para outras funções (logarítmicas, trigonométricas etc.). O método das diferenças, conveniente para os implementos mecânicos da época, trabalha apenas com adições (subtrações), iterativamente, com um algoritmo muito simples, como a seguir descrito, sem perda de generalidade, por meio de um exemplo numérico.

Tabela 1.1 – Diferenças da função $2x^2 - 3x + 2$

x	$p_2(x)$	dif_1	dif_2
0	2,00		
		0,28	
0,1	1,72		0,04
		0,24	
0,2	1,48		0,04
		0,20	
0,3	1,28		0,04
		0,16	
0,4	1,12		

Considere um polinômio do segundo grau a ser tabelado, por exemplo: $p_2(x) = 2x^2 - 3x + 2$. Suponha então que se deseja tabelar esse polinômio, a partir de $x = 0$, com incrementos iguais a 0,1, para obter $p_2(0)$, $p_2(0,1)$, $p_2(0,2)$, e assim por diante, como mostram as duas primeiras colunas da Tabela 1.1. A terceira coluna dessa tabela (dif_1 , primeiras diferenças) resulta da subtração de valores vizinhos consecutivos de $p_2(x)$. A última coluna, dif_2 , contém as diferenças das diferenças calculadas na coluna dif_1 . Nota-se que todos os valores obtidos para dif_2 são iguais. Não por coincidência, já que são diferenças de uma função polinomial de grau dois.

As colunas da Tabela 1.1 foram sendo obtidas da esquerda para a direita. Para tabelar novos valores de $p_2(x)$, basta que se caminhe no sentido inverso. O valor de $p_2(0,5)$ será obtido, partindo-se da coluna dif_2 com o valor 0,04 que será subtraído de 0,16 colhido em dif_1 , ou seja, $0,16 - 0,04 = 0,12$. O próximo valor de $p_2(x)$, em $x = 0,5$, será encontrado por meio da subtração de 0,12 do último valor de $p_2(x)$ tabelado, ou seja, $p_2(0,5) = p_2(0,4) - 0,12 = 1,00$. Para calcular $p_2(0,6)$, repete-se o mesmo algoritmo, agora usando os novos valores agregados à tabela, correspondentes ao cálculo anterior de $p_2(0,5)$, *i.e.*: toma-se 0,04 em dif_2 , subtrai-se este valor de 0,12 obtido em dif_1 e então encontra-se 0,08; depois subtrai-se este resultado de $p_2(0,5)$, obtendo $p_2(0,6) = 1,00 - 0,08 = 0,92$. O processo descrito pode continuar da mesma forma, até que tenham sido obtidos todos os valores de $p_2(x)$ para compor a tabela desejada. Destaca-se o fato de que os valores do polinômio são obtidos sem uma única multiplicação, requerendo assim da máquina de diferenças a preparação para receber apenas operações de subtração. De um cálculo de $p_2(x)$ para o outro, necessita-se armazenar apenas dois valores: os últimos elementos das colunas $p_2(x)$ e dif_1 . Os valores obtidos pelo uso das diferenças são confirmados por:

$$p_2(0,5) = 2 \times 0,5^2 - 3 \times 0,5 + 2 = 1 \text{ e } p_2(0,6) = 2 \times 0,6^2 - 3 \times 0,6 + 2 = 0,92$$

Em 1823, o governo britânico financiou a construção de uma nova versão da máquina de diferenças de Babbage, projetada para 7 registros e impressão de resultados. Porém, o inventor encontrou dificuldades para executar o projeto, por conta das limitações da indústria da época, requerendo de Babbage a confecção de peças e ferramentas. Em decorrência disso, teve o apoio financeiro suspenso, mas continuou até que em 1842 deixou de lado a máquina de diferenças. Contudo, isso foi temporário, já que entre 1847 e 1849, retornou ao projeto original da máquina de diferenças para aperfeiçoá-lo.

Como Babbage não terminava suas máquinas, duvidava-se se elas de fato funcionariam. Assim, de 1989 a 1991, com base nos planos que Babbage aperfeiçoou, o *Science Museum* de Londres construiu uma máquina de diferenças (finalizada pouco antes do aniversário de duzentos anos do seu criador). Em 2000, a impressora idealizada por Babbage também foi concluída. Ambas funcionaram perfeitamente, pondo fim ao ceticismo que rondava os projetos de Babbage, no que diz respeito à construção de máquinas funcionais. A título de curiosidade, cabe mencionar que a máquina – com capacidade para armazenar 7 números (com 31 dígitos) – pesa 2,6 toneladas, apenas em sua seção calculadora.

Em 1833, com o auxílio de Ada Byron King (Figura 1.16), Condessa de Lovelace, filha do poeta Lord Byron, Babbage iniciou o projeto de uma máquina que chamou de Analítica. Esta, mais geral do que aquela de diferenças, operava comandada por um conjunto de cartões perfurados que continham não apenas números a serem processados, mas também instruções, já que a máquina possuía unidade de memória, processamento aritmético e entrada/saída de dados. Ada concebeu uma série de instruções para a Máquina Analítica, tornando-se a primeira programadora de computador. Babbage investiu os recursos financeiros que possuía, assim como os de seu filho, com quem trabalhou durante anos para construir a Máquina Analítica. Babbage faleceu em 1871, sem pôr a termo a construção pretendida. Babbage merece ser considerado pioneiro na arte de conceber computadores e Ada no que toca à programação.



Figura 1.16 – Ada Byron King (1815-1852).

Fonte: https://mathshistory.st-andrews.ac.uk/Biographies/Lovelace/lovelace_9.jpg

Lógica Booleana

Embora não tenha materializado a construção de uma máquina de calcular, o inglês George Boole (Figura 1.17) desenvolveu com sucesso os fundamentos do primeiro sistema formal para o raciocínio lógico, quase um século antes que fossem inventados computadores digitais apoiados na Lógica Booleana. Boole percebeu que com os operadores lógicos E, OU e NÃO era possível descrever qualquer proposição lógica (todo enunciado que se avalia como verdadeiro ou falso, sim ou não etc.).



Figura 1.17 – George Boole (1815-1864).

Fonte: <https://mathshistory.st-andrews.ac.uk/Biographies/Boole/Boole.jpeg>

Em 1938, o americano Claude Elwood Shannon (1916-2001) aplicou a Álgebra Booleana e Aritmética Binária para mostrar que era possível simplificar o arranjo de relés eletromecânicos usados para chaveamento de circuitos telefônicos. Também mostrou que se pode, por meio do arranjo de relés resolver problemas de Álgebra Booleana. Explorar o uso de chaves elétricas em problemas de lógica tornou-se um conceito básico de todos os computadores digitais eletrônicos. Shannon contribuiu significativamente para o campo da Informação, estabeleceu como unidade fundamental o *bit* (termo oriundo de *binary digit*) e mostrou que os dígitos da base binária poderiam ser associados a dois estados (e.g., ligado-desligado), podendo representar qualquer coisa.

Máquina de tabulação

Por volta de 1890, o norte-americano Herman Hollerith (Figura 1.18) protagonizou uma nova maneira de processar dados, por intermédio de uma máquina eletromecânica (Figura 1.19) que inventara, capaz de contar e compilar (tabular), destinada ao censo populacional dos Estados Unidos. Baseou-se na ideia de Babbage de ter como memória

do seu engenho registros em cartões perfurados. Os dados dos indivíduos (*e.g.*, sexo, idade, ocupação) eram registrados por meio de perfurações em locais específicos de cartões de papel, cuja leitura usava um dispositivo, como um pente com agulhas metálicas que ao ultrapassar um furo de cartão fechavam um circuito elétrico que acionava o sistema de contagem. Considera-se que a máquina de Hollerith tenha reduzido sobremaneira o tempo necessário à conclusão do censo (o anterior, em 1880, levou 7 anos para ser tabulado de forma totalmente manual), apresentando uma capacidade de processamento pelo menos dez vezes superior a de métodos manuais. Hollerith fundou uma companhia para produzir máquinas de tabulação que anos depois, em 1924, incorporando-se a outras comporiam a *International Business Machines (IBM)*.



Figura 1.18 – Herman Hollerith (1860-1929).

Fonte: <https://mulheresnacomputacao.com/2012/07/31/hollerith/>

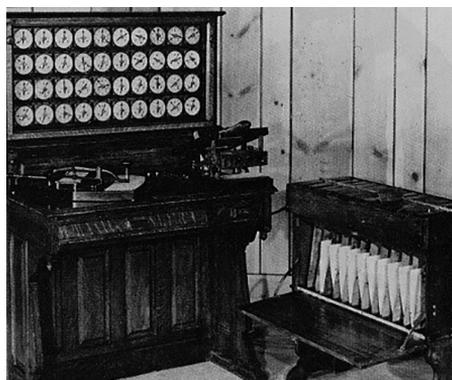


Figura 1.19 – Máquina tabuladora eletromecânica.

Fonte: <https://mulheresnacomputacao.com/2012/07/31/hollerith/>

Computadores

Provavelmente impulsionados pela indústria militar, com o início da década de 1940, o alemão Konrad Zuse (1910-1995) produziu os computadores Z-1 a Z-4 e o norte-americano Howard Aiken (1900-1973) os Mark I a IV, consideradas as primeiras séries de computadores de tantas outras que levaram ao progresso hoje alcançado. Os estadunidenses John Presper Eckert (1919-1995) e John Mauchly (1907-1980) projetaram o computador chamado ENIAC (*Electronic Numerical Integrator and Calculator*), objetivando realizar cálculos balísticos. Finalizado em 1946 (vários meses após o final da Segunda Guerra Mundial), o ENIAC era fisicamente um gigante de 30 toneladas, 5,50 m de altura e 25 m de comprimento, ocupando 180 m² de área construída. Composto por cerca de 18.000 válvulas e outros milhares de componentes eletrônicos, apresentava custos de manutenção e conservação proibitivos, já que dezenas/centenas de válvulas frequentemente deixavam de funcionar. Ademais, a energia térmica delas desprendida exigia um sistema de refrigeração sofisticado, ampliando ainda mais o já elevado consumo de energia elétrica da máquina (cerca de 200 kW).

Os computadores que sucederam ao ENIAC buscaram ampliar a capacidade de cálculo e reduzir tamanho e consumo. Eckert e Mauchly também construíram o UNIVAC (*Universal Automatic Computer*), finalizado em 1951, destinado ao processamento de dados dos censos da população americana. O UNIVAC usava 5.200 válvulas eletrônicas, pesava 13 toneladas e consumia 125 kW e ocupava por volta de 35 m². Encontra-se no endereço eletrônico www.computerhistory.org farto material sobre a evolução dos computadores ao longo do tempo.

A arquitetura de um computador depende do seu projeto lógico, enquanto a sua implementação depende da tecnologia disponível. As três primeiras gerações de computadores refletiram a evolução dos componentes básicos do computador (hardware) e um aprimoramento dos programas (software) existentes. Os computadores de primeira geração (1945-1959) usaram válvulas eletrônicas, quilômetros de fios, eram lentos, enormes e esquentavam muito. A segunda geração (1959-1964) substituiu as válvulas eletrônicas por transistores e fios por circuitos impressos, o que tornou os computadores mais rápidos, menores e de custo mais baixo. Já a terceira geração de computadores (1964-1970) foi construída com circuitos integrados, proporcionando maior compactação, redução dos custos e velocidade de processamento. A quarta geração, de 1970 até hoje, caracteriza-se por um aperfeiçoamento da tecnologia existente, otimizando a máquina para atender os problemas do usuário, maior grau de miniaturização, confiabilidade e velocidade de processamento. O termo quinta geração, atribuído a computadores “inteligentes”, caracteriza os avanços alcançados em meados da década de 1990. Posteriormente, o termo passou a envolver elementos de diversas áreas de pesquisa relacionadas à inteligência computacional, sistemas especialistas e linguagem natural. Acresça-se a isso a conectividade, *i.e.*, a capacidade de conexão em rede de computadores. O conceito de supervia da informação capturou a imaginação tanto de profissionais da computação como de usuários comuns.

1.4 REFLEXÕES

Neste capítulo, buscou-se uma caracterização preliminar do conteúdo do presente livro, no que diz respeito à natureza de soluções numéricas (não analíticas) comumente propostas para problemas oriundos de diversos campos do conhecimento científico, descritos em linguagem matemática. Tais soluções devem ser sistematizadas por meio do que se denomina método numérico, a ser implementado usando-se um conjunto de instruções (algoritmo) executadas de acordo com os recursos de cálculo disponíveis. Além disso, foram reunidos os principais marcos deixados na História sobre a capacidade de se realizar cálculos, a começar pelos implementos manuais, passando-se por aqueles de natureza mecânica, até se atingir os computadores eletrônicos. No fim do livro, na seção Bibliografia, há uma lista de referências para uma leitura mais completa sobre a História da Computação, tema muitas vezes deixado de lado por estudantes apressados em colher apenas um verniz de cunho mais técnico.

De modo a reduzir o esforço dispendido em tarefas de cálculo repetitivo (iterativo), portanto visto como laborioso e enfadonho, observou-se o desenvolvimento constante de uma série de máquinas destinadas a substituir o cérebro humano em tais tarefas. Disso resultou um desejo mais amplo de se obter a capacidade de resolver problemas concernentes às necessidades e hábitos da vida em sociedade (urbana, pautada pela praticidade, conforto, comodidade etc.), de forma sistemática e rápida, automatizados por meio de máquinas de uso geral. Nesse ponto, para encurtar caminhos próprios do raciocínio mais elaborado, deixa-se ao leitor uma sutil indagação, cuja resposta só o tempo trará: que limites para a criação de máquinas “computadoras” deverão ser estabelecidos para o homem, quando atualmente estas começam a ser comandadas por voz, comunicam-se entre si, processam imagens, buscando ser dotadas da capacidade de visão, e assumem a forma de robôs? Estaria o homem buscando criar máquinas à sua imagem e semelhança? Há muito que refletir...



ESTE LIVRO TRATA A COMPUTAÇÃO NUMÉRICA DE FORMA DIDÁTICA E AGRADÁVEL, AOS INTERESSADOS EM ALCANÇAR UM BOM ENTENDIMENTO SOBRE OS PRINCIPAIS MÉTODOS NUMÉRICOS E SUAS IMPLEMENTAÇÕES, NECESSÁRIOS AO CÁLCULO DA SOLUÇÃO DE DIVERSOS PROBLEMAS CIENTÍFICOS DO MUNDO REAL.

Adota-se um grau de profundidade da matéria compatível com o desejado em cursos de graduação, contudo sem deixar de indicar aspectos relevantes ao aprofundamento do tema, usualmente próprios de estudos de pós-graduação. Vários exercícios foram formulados, resolvidos e propostos (todos com respostas), para a fixação da teoria exposta. Cálculos envolvendo números complexos são enfocados, fato usualmente deixado de lado em livros semelhantes a este. A prática de programação nos ambientes Scilab e Python também estão presentes no texto.



www.blucher.com.br

Blucher



Clique aqui e:

[VEJA NA LOJA](#)

Computação numérica Métodos e implementações

Milton Brown do Coutto Filho, Vinícius Biajoni Braga Flôr

ISBN: 9786555066067

Páginas: 352

Formato: 17 x 24 cm

Ano de Publicação: 2024
