Métodos Numéricos

Prof. Dr. Jonatha Costa

2024

Organização

MN aplicados à Engenharia: Objetivo da aula

- Apresentar conteúdo de:
    - Sistemas de numeração
    - Conversão entre sistemas
    - Operações aritméticas com binários

MN aplicados à Engenharia: Sistemas de Numeração

Decomposição de um Número em um Sistema de Bases Em geral qualquer número pode ser decomposto numa soma dos dígitos que o constitui (d_j) multiplicado pelas potências da sua base (β):

Figura: Termo geral numérico

$$
\begin{aligned}
(N)_B &= (\overbrace{d_n d_{n-1} d_{n-2} \ldots d_0}^{\text{Parte inteira}}, \overbrace{d_{-1} d_{-2} \ldots d_{-m}}^{\text{Parte fracionária}})_\beta \\
&= d_n\beta^n + d_{n-1}\beta^{n-1} + d_{n-2}\beta^{n-2} + \ldots + d_0\beta^0 + d_{-1}\beta^{-1} + d_{-2}\beta^{-2} + d_{-m}\beta^{-m}
\end{aligned}
$$

Fonte: DIAS,(2019)

onde os digitos d_j pertencem aos números naturais e satisfazem a condição: $0 \leq d_j \leq (\beta - 1)$

MN aplicados à Engenharia:Sistemas de Numeração

Sistema de Numeração Decimal ou Base 10

Todos os múltiplos e submúltiplos de um número são escritos com potencias de 10.

- 1537=
  $(1537)_{10} = 1 \times 10^3 + 5 \times 10^2 + 3 \times 10^1 + 7 \times 10^0$
- 36,189=
  $(36,189)_{10} = 3 \times 10^1 + 6 \times 10^0 + 1 \times 10^{-1} + 8 \times 10^{-2} + 9 \times 10^{-3}$
- $6,032 \times 10^{23} =$
  $(6,032 \times 10^{23})_{10} = 6 \times 10^{23} + 0 \times 10^{22} + 3 \times 10^{21} + 2 \times 10^{20}$

MN aplicados à Engenharia: Sistemas de Numeração

Sistema de Numeração Binário ou Base 2

Neste caso todos os múltiplos e submúltiplos de um número são escritos com potencias de 2.

- $(10111)_2 =$
  $(10111)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
- $(10,1)_2 =$
  $(10,1)_2 = 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1}$

Nota: Os computadores digitais operam basicamente com dois tipos de sinais de tensão: baixo e alto. Matematicamente, pode-se expressar esses valores por 0 (baixo) e 1 (alto)

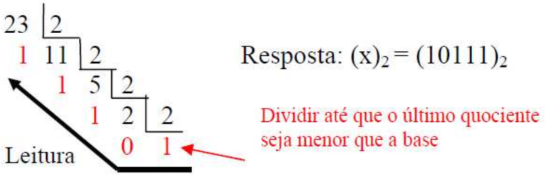
MN aplicados à Engenharia: Sistemas de Numeração

Conversão de números (decimal para binário)

Devemos aplicar um método para a parte inteira (divisões sucessivas) e um método para a parte fracionaria, se houver (multiplicações sucessivas).

Ex: $(23)_{10} \longrightarrow (x)_2$

Figura: Decimal para binário - 23_{10}



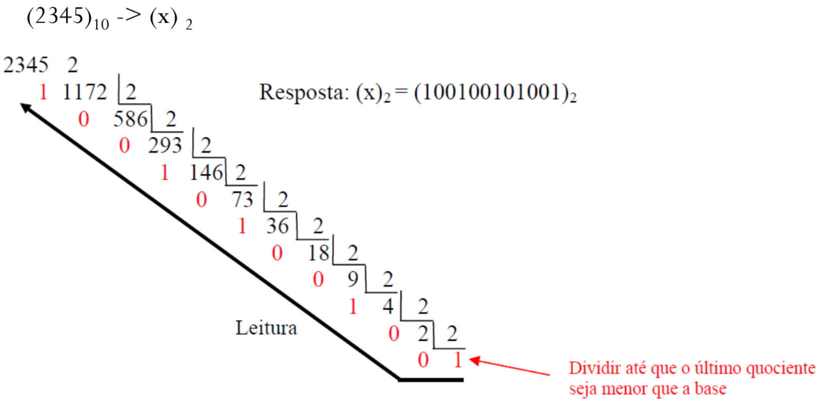
Fonte: DIAS,(2019)

A partir de uma sequência de 0s e de 1s podemos expressar “qualquer” número decimal?

MN aplicados à Engenharia: Sistemas de Numeração

Conversão de números (decimal para binário)

Figura: Decimal para binário - 2345_{10}



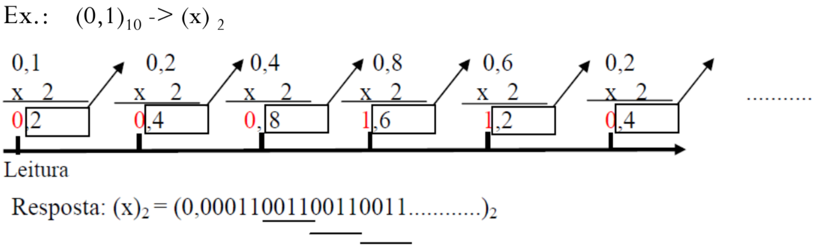
Fonte: DIAS,(2019)

MN aplicados à Engenharia: Sistemas de Numeração

Conversão de números (decimal para binário)

Para números fracionários utilizamos a regra da multiplicação:

Figura: Decimal para binário - $0,1_{10}$



Fonte: DIAS,(2019)

Conclui-se que o número $(0,1)_{10}$ NÃO tem representação binária finita!

Por mais moderno que seja o computador ele nunca vai saber exatamente o que significa o numero $(0,1)_{10}$, pois sua conversão para binário sempre acarretará numa aproximação.

MN aplicados à Engenharia: Sistemas de Numeração

Conversão de números (decimal para binário)

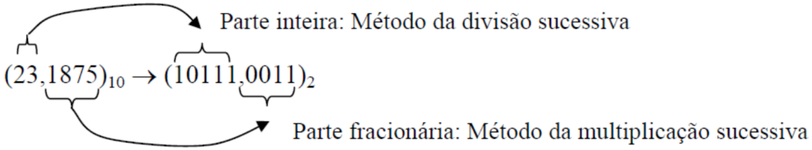
Alerta: O fato de um número não ter representação finita no sistema binário pode acarretar a ocorrência de erros aparentemente inexplicáveis nos cálculos dos dispositivos eletrônicos.

MN aplicados à Engenharia: Sistemas de Numeração

Conversão de números (decimal para binário)

Resumo:

Figura: Decimal para binário - resumo



Fonte: DIAS,(2019)

MN aplicados à Engenharia: Sistemas de Numeração

Adição de binários

Propriedades:

- $0 + 0 = 0$
- $1 + 0 = 1$
- $0 + 1 = 1$
- $1 + 1 = 0$ e eleva 1 para **somar** ao dígito imediatamente à esquerda deste.

Figura: Adição de binários

$$
\begin{array}{rr}
 & 1 \\
 & 1100 \\
+ & 111 \\
 & ----- \\
= & 10011
\end{array}
\qquad\qquad
\begin{array}{rr}
 & 11 \\
 & 1100 \\
+ & 1111 \\
 & ----- \\
= & 11011
\end{array}
$$

Fonte: DIAS,(2019)

MN aplicados à Engenharia: Sistemas de Numeração

Subtração de binários

Propriedades:

- $0 - 0 = 0$
- $1 - 0 = 1$
- $0 - 1 = 1$ e eleva 1 para **subtrair** ao dígito imediatamente à esquerda deste.
- $1 - 1 = 0$

Figura: Subtração de binários

```
     1 111
    1101110
-     10111
    -------
=   1010111
```

Fonte: DIAS,(2019)

MN aplicados à Engenharia: Sistemas de Numeração

Multiplicação de Binários

Efetua-se o produto entre 0 e 1 elemento-a-elemento seguindo-se a soma na ordem coluna-coluna.

Figura: Multiplicação de binários

```
          1 0 1 1
        x 1 0 1 0
        ---------
          0 0 0 0
+     1 1 0 1 1
+     0 0 0 0
+ 1 0 1 1
---------------
= 1 1 0 1 1 1 0
```

```
          1 1 1
      x   1 1 1
      ---------
        1 1
      1 1 1 1 1
+     1 1 1 1
+   1 1 1 1
  --------------
=   1 1 0 0 0 1
```

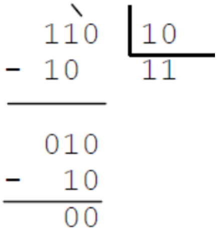
Fonte: DIAS,(2019)

MN aplicados à Engenharia: Sistemas de Numeração

Divisão de Binários

Efetua-se a divisão entre elemento-a-elemento seguindo-se a subtração na ordem coluna-coluna.

Figura: Divisão de binários



Fonte: DIAS,(2019)

Organização

MN aplicados à Engenharia: Objetivo da aula

- Apresentar conteúdo de :
    - Aritmética de ponto flutuante
    - Armazenamento de dados no computador
    - Erros de arredondamento e de truncamento

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Sejam as grandezas:

- Massa do Elétron: $9 \times 10^{-28} gramas$
- Massa do Sol: $2 \times 10^{33} gramas$
- Faixa de variação: $> 10^{60}$
- Representação de grandezas extremas
  - 000000000.1324468585133
  - 1367634123544 5403.341464684654

Como representar tais números de modo equânime?

MN aplicados à Engenharia: Aritmética de Ponto flutuante

- Solução: notação científica:

$$Algarismo \times Base^{expoente}$$

- Sistema de representação de maneira que a faixa de variação dos números seja independente do número de dígitos significativos dos números representados.
- Ponto flutuante

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Representação em Ponto flutuante - base decimal

Para acomodar números grandes e pequenos, números reais são escritos na representação em ponto flutuante (também chamada de notação científica) e cuja a forma:

$$d, ddddd \times 10^p$$

Nessa representação, um algarismo é escrito à esquerda da vírgula decimal, e o resto dos algarismos significativos é escrito à direita da vírgula. O número $0, dddddd$ é chamado de **mantissa**.

- $6519, 23$ é escrito como $6, 51923 \times 10^3$
- $0, 00000391$ é escrito como $3, 91 \times 10^{-6}$

A potência de 10^p, representa a ordem de grandeza do número(O), a qual é estruturada como $(p + 1)$ se a ordem for menor que 5.

Exemplo:

- $3, 91 \times 10^{-6}$ é da ordem de $10^{-6}.O(10^{-6})$
- $6, 51923 \times 10^3$ é da ordem de $10^4.O(10^4)$.

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Representação em Ponto flutuante - base binária

A representação binária em ponto flutuante tem a forma:

$$1,bbbbb \times 2^{bbb}$$

Nessa forma, a mantissa é $0,bbbbbb$, e a potência de 2 é chamada de expoente. Tanto a mantissa quanto o expoente são escritos na forma binária.

Para representar um número decimal em Ponto flutuante - base binária deve-se proceder a normalização do número e depois a conversão do número para a representação binária.
Um número n, portanto, deve ser dividido e multiplicado pela *maior potência de 2*, imediatamente menor

que o próprio número n. Depois tanto a mantissa quanto o expoente são escritos na forma binária, posto que a parte inteira é 1.

Exemplo: Como é escrito 50_{10} em binário? E na representação de ponto flutuante binária?

1. Normalizar - $50_{10} = \dfrac{50}{2^5} \times 2^5 = 1,5625 \times 2^5$
2. Converter mantissa - $0,5625_{10} = 1001_2$
3. Converter expoente - $5_{10} = 101_2$
4. Compor o número - $1,1001 \times 2^{101}$

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Representação em Ponto flutuante - exemplo

Formato	***Número***
Decimal	81

↓

Base binária	**Expoente**
2	6
2^6 < 81 ?	Ok

↓

Resultado Ponto flutuante decimal

$$\frac{81}{(2\hat{}6)} \times (2\hat{}6) \longrightarrow 1{,}265625 \times (2\hat{}6)$$

Parte inteira	1
Parte fracionária	0,265625
Expoente	6

Ponto flutuante decimal para binário

1,265625 x (2^6) ⟶ ?

Conversão para binário - parte inteira

1=1

Conversão para binário - mantissa 010001

0,265625	x 2 =	0,53125	0
0,53125	x 2 =	1,0625	1
0,0625	x 2 =	0,125	0
0,125	x 2 =	0,25	0
0,25	x 2 =	0,5	0
0,5	x 2 =	1	1
0	x 2 =	0	

Conversão para binário - expoente

6 110

Resultado Ponto flutuante binário

1,265625 x (2^6) ⟶ 1,010001 x (2^110)

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Representação em Ponto flutuante - exemplo

Método alternativo

1. Converter o número decimal para binário
$81_{10} = 1010001_2$
2. Deslocar a vírgula para direta e registrar o número de deslocamentos
$1,010001$ e 6 deslocamentos.
3. Converter a quantidade de deslocamentos para binário
$6_{10} = 110_2$
4. Reescrever o número unindo os termos acima
$1,010001\ x\ 2^{110}$

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Representação em Ponto flutuante

Distribuição dos bits na representação IEEE 754/2008

Tipo	Sinal	Expoente	Mantissa
Simples (32 bits)	bit 31	bits 30-23	bits 22-0
Dupla (64 bits)	bit 63	bit 62-52	bits 51-0

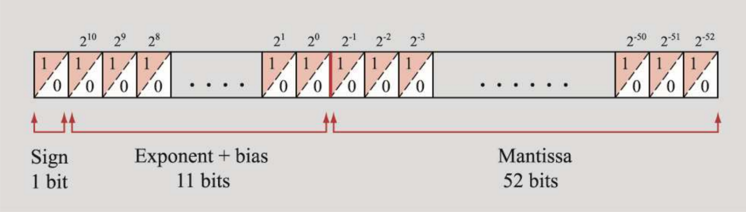
- Precisão simples 32 bits (dupla 64 bits)
- 1 bit para o sinal de número em ambas as precisões
- 8 bits para o expoente (11 bits para o expoente) + polarização (127 simples e 1023 dupla)
- 23 bits para a mantissa (52 bits para a mantissa)

A polarização é introduzida para se evitar o uso de um dos bits para representar o sinal do expoente (uma vez que o expoente pode ser positivo ou negativo).

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Resumo:

Figura: Representação IEEE 754/2008 - Processador de 64 bits



Fonte: GILAT,(2008)

Processador de 64 bits:

1 bit para sinal, 11 bits para o expoente e 52 bits para a mantissa, bias $= 1023$

Processador de 32 bits:

1 bit para sinal, 8 bits para o expoente e 23 bits para a mantissa, bias $= 127$

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Qual a representação do número 81 decimal na norma IEEE 754/2008 precisão de 32 e 64 bits?

- 81_{10} em ponto flutuante *binário* resulta em $1,010001\ x\ (2^{110})$, conforme já visto.
- Precisão simples (32 bits)
  - Bit 31 (sinal) = positivo 0;
  - Bits 23 ao 30 (expoente com bias de 127) $= 6_{10} + 127_{10} = 133_{10} = 10000101_2$
  - Bits 00 ao 22 (mantissa) $= 010001_2$
  - Logo: **0 | 10000101 | 0100010000000...0000 - 32bits**
- Precisão dupla (64 bits)
  - Bit 63 (sinal) = positivo 0;
  - Bits 52 ao 62 (expoente com bias de 1023) $= 6_{10} + 1023_{10} = 1029_{10} = 10000000101_2$
  - Bits 00 ao 51 (mantissa) $= 010001_2$
  - Logo: **0 | 10000000101 | 01000100000...0000 - 64bits**
- Nota:

  Lembre-se de que a sintaxe é : *sinal* + *expoente* + *mantissa*!

MN aplicados à Engenharia: Aritmética de Ponto flutuante

Representação em Ponto flutuante
Considerações

- Mais bits para a mantissa fornece mais precisão (t)
- Mais bits para o expoente, aumenta o range de valores (e).
- O tamanho da palavra do computador depende de características internas à arquitetura do mesmo. Em geral, os microcontroladores tem tamanho de palavra de 16 bits, os microcomputadores padrão PC tem tamanho de palavra de 32 bits, 64 bits ou mais.

Figura: Encapsulamentos de processadores

Fonte: Autor desconhecido

MN aplicados à Engenharia: Erros em soluções numéricas

Erros em soluções numéricas

- Erros de arredondamento
- Erros de truncamento
- E rro total

MN aplicados à Engenharia: Erros em soluções numéricas

Erros de arredondamento

- Os números são representados em computador através de um número finito de bits.
- Os números podem ser encurtados e descartados ou arredondados.

Exemplo:

Figura: Erros de arredondamento

$$2/3 = 0{,}666666\ldots$$

$$2/3 = \underbrace{0{,}6666}\ \text{ou}\ \underbrace{0{,}6667}$$

Encurtado / arredondado

Fonte: DIAS,(2019)

MN aplicados à Engenharia: Erros em soluções numéricas

Erros de arredondamento:

Exemplo: Efetue a subtração entre dois números reais $p = 9890,9$ e $q = 9887,1$ mantendo o formato original, encurtando e arredondando os valores para três algarismos significativos.

- $p - q = 9890,9 - 9887,1 = 3,8$ (formato original)
- Se apenas três algarismos são permitidos na mantissa, tem-se a **representação em ponto flutuante**:
    - $p - q = 9,890\ x\ 10^3 - 9,887\ x\ 10^3 = 0,003\ x\ 10^3 = 3$ (corte)
    - $p - q = 9,891\ x\ 10^3 - 9,887\ x\ 10^3 = 0,004\ x\ 10^3 = 4$ (arredondamento)

A diferença verdadeira (exata) entre os números é de 3,8. Esses resultados mostram que, no presente problema, o arredondamento leva a um valor mais próximo à resposta verdadeira.

MN aplicados à Engenharia: Erros em soluções numéricas

Exemplo

- Considere a equação quadrática $x^2 - 100,0001x + 0,01 = 0$, para a qual as soluções exatas são $x_1 = 100$ e $x_2 = 0,0001$.
- As soluções podem ser calculadas com a fórmula quadrática $x_1 = \dfrac{-b + \sqrt{b^2 - 4ac}}{2a}$ e $x_2 = \dfrac{-b - \sqrt{b^2 - 4ac}}{2a}$, o que, em ambiente de simulador computacional Matlab/Octave resultando em: $x_1 = 99.99990000000000$ e $x_2 = 1.000000000033197e - 004$.
- O valor calculado no simulador para x_2 não é exato devido a erros de arredondamento. Tais erros ocorrem no numerador da expressão de x_2 . Como b é negativo, o numerador envolve a subtração de dois números que são quase iguais.
- Em alguns casos, a forma das expressões matemáticas pode ser mudada para uma forma diferente, menos propensa a erros de arredondamento como:
$$x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}\left(\frac{-b + \sqrt{b^2 - 4ac}}{-b + \sqrt{b^2 - 4ac}}\right) = \frac{2c}{-b + \sqrt{b^2 - 4ac}}.$$
- O que resulta e $x_2 = 1.000000000000000e - 004$.

MN aplicados à Engenharia: Erros em soluções numéricas

Erros de truncamento:

- Ocorrem quando os métodos numéricos usados na solução de um problema utilizam uma aproximação.
- **Exemplo**: A função sen(x) pode ser aproximada pela série de Taylor.

$$sen(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \frac{x^{11}}{11!} + \dots$$

  - O valor exato do $sen(\pi/6) = 0,5$
  - Usando **um** termo da série de Taylor: $sen(\pi/6) = (\pi/6) = 0,5235988$
  - Usando **dois** termos $sen(\pi/6) = \frac{\pi}{6} - \frac{(\pi/6)^3}{3!} = 0,4996742$

MN aplicados à Engenharia: Erros em soluções numéricas

Erro total (Erro real)

- A solução numérica é uma aproximação sempre inclui erros de arredondamento e, dependendo do método numérico utilizado, também pode incluir erros de truncamento.
- Juntos, os erros de arredondamento e de truncamento resultam no erro numérico total incluído na solução numérica.
- **Erro total** - ou erro real, é a diferença entre a solução verdadeira (exata) e a solução numérica sendo expresso por :

$$ErroReal = SolucaoExata \; - \; SolucaoNumerica$$

- **Erro relativo** - valor absoluto da razão entre o erro real e a solução exata sendo expresso por:

$$Erro\ Relativo\ Real = \frac{Erro\ Real}{Solucao\ Exata}$$

MN aplicados à Engenharia: Erros em soluções numéricas

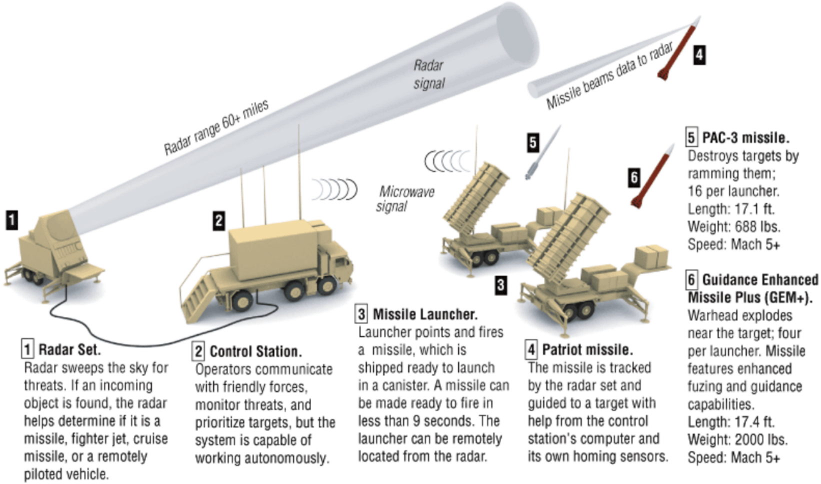
Desastres causados por erros numéricos:

Figura: Defesa antiaérea - Patriot

Fonte: DIAS,(2019)

MN aplicados à Engenharia

Figura: Defesa antiaérea - Patriot



Fonte: DIAS,(2019)

MN aplicados à Engenharia: Erros em soluções numéricas

Desastres causados por erros numéricos:

Figura: Míssel Iraniano

Fonte:DIAS,(2019)

MN aplicados à Engenharia: Erros em soluções numéricas

Desastres causados por erros numéricos:

O que aconteceu?

- Dia 25 de fevereiro de 1991 – Guerra do Golfo;
- Arábia Saudita – Sistema Patriot falhou na interceptação de mísseis SCUD;
  – Morte de 28 soldados americanos;
- Para prever onde mísseis estarão, ambos tempo e velocidade precisam ser números reais;
- O tempo em dezenas de segundos era multiplicado por $1/10$ para produzir o tempo em segundos;

MN aplicados à Engenharia: Erros em soluções numéricas

Desastres causados por erros numéricos:

- 1/10 quando convertido para base binária resulta em uma dízima com período igual a 104 bits;
- Patriot possuía um registrador de 24 bits;

- Conversão binária de 1/10
  0.0001100110011001100110011001100...
- O registrador do Patriot gravou
  0.00011001100110011001100
- Erro gerado
  0.0000000000000000000000011001100...

MN aplicados à Engenharia: Erros em soluções numéricas

Desastres causados por erros numéricos

- Tempo depois de 100 horas
  $0.000000095x100x60x60x10 = 0,34$ segundos
- Velocidade do Scud $= 1.676m/s$
  Distância percorrida em $0,34$ segundos $= 0,5km$
- Quem foi o responsável?
  O projetista? O exército americano? Operadores?

MN aplicados à Engenharia: Erros em soluções numéricas

Desastres causados por erros numéricos

Figura: Ariane 5

Fonte: DIAS,(2019)

Explosão do foguete francês Ariane 5.

- Em 4 de junho de 1996, menos de um minuto após o lançamento, o foguete francês Ariane 501 se autodestruiu.
- Foi indicada pelo CNES (Centro Nacional de Estudos Espaciais) e pela ESA (Agência Espacial Européia) uma comissão para investigar a ocorrência.
- Comissão indica um erro no software de controle como a origem na falha do lançamento.

MN aplicados à Engenharia: Erros em soluções numéricas

Desastres causados por erros numéricos

Figura: Ariane 5

Fonte: DIAS,(2019)

A investigação preliminar dos dados de voo revelou:

- Comportamento do lançador em condições nominais, até 36 segundos após a decolagem.
- Falha do sistema de referência inercial (SRI) secundário, seguido imediatamente de falha do sistema de referência inercial em operação.
- Guinada dos bocais dos sistemas de propulsão até o seu ângulo máximo, causando uma guinada abrupta do veículo.
- Autodestruição do lançador, disparada corretamente como consequência da ruptura das juntas entre os sistemas de propulsão sólidos e o primeiro estágio.

MN aplicados à Engenharia: Erros em soluções numéricas

Desastres causados por erros numéricos

Figura: Ariane 5

Fonte: DIAS,(2019)

A investigação preliminar dos dados de voo revelou:

- A origem da falha foi confinada ao sistema de controle, mais especificamente, aos dois sistemas referenciais inerciais (SRI), que nitidamente deixaram de funcionar, por volta de $36,7s$ após a decolagem.
- A anomalia interna de *software* do SRI ocorreu durante a execução de uma conversão de dados de um número de 64 bits em ponto flutuante para um inteiro de 16 bits com sinal. O valor do número em ponto flutuante era maior do que poderia ser representado pelo inteiro de 16 bits com sinal. O resultado foi um operando inválido.

Organização

MN aplicados à Engenharia: Objetivo da aula

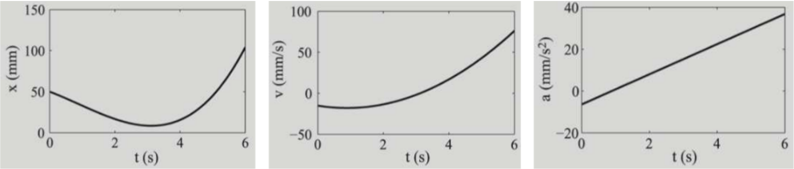
- Apresentar conceitos de diferenciação
- Revisar estruturas de operação de diferenciação
- Apresentar a interpretação gráfica de diferenciação
- Apresentar estruturas de diferenciação: progressiva, regressiva e central

MN aplicados à Engenharia: Diferenciação Numérica

Fundamentos

- É a taxa na qual uma grandeza varia.

Figura: Curvas de deslocamento mm/s



Fonte: GILAT,(2008)

- Em um circuito elétrico: a corrente elétrica(i) é a taxa de variação da carga q, a corrente em um capacitor está relacionada à derivada temporal da tensão(v).
- Na análise da condução de calor, a quantidade de fluxo de calor(ϕ) é determinada a partir da derivada da temperatura(Δt).

MN aplicados à Engenharia: Diferenciação Numérica

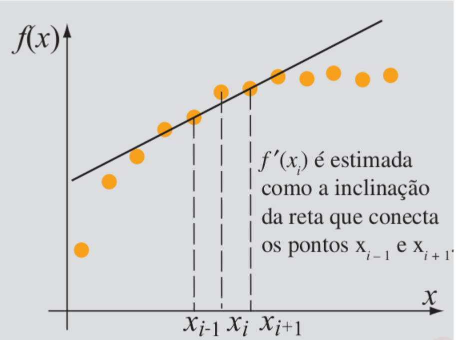
Fundamentos

- A diferenciação é usada na obtenção dos **máximos** e **mínimos** de uma função.
- A *teoria de Métodos Numéricos*, solução por aproximação, é usada quando a diferenciação de uma função é difícil ou impossível através de uma solução analítica.
- A diferenciação numérica é usada em um conjunto de pontos discretos, medidos ou gravados em experimentos.

MN aplicados à Engenharia: Diferenciação Numérica

Derivada por diferenças finitas

Figura: Derivada por diferenças



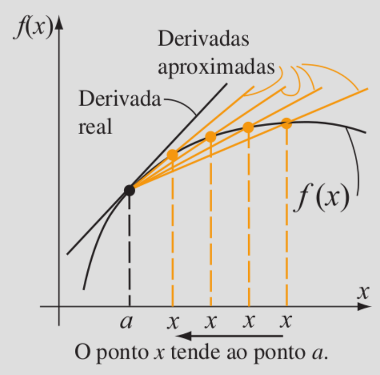
Fonte: GILAT,(2008)

- A aproximação da derivada em um ponto x_i se baseia nos valores dos pontos na vizinhança de x_i.
- A aproximação depende da precisão dos pontos, do espaçamento entre eles e da expressão específica usada na aproximação.
- A **derivada** no ponto x_i é aproximada pela inclinação da reta que liga o ponto antes de x_i ao ponto após x_i .
- Problemas de ruído penalizam este tipo de aproximação.

MN aplicados à Engenharia: Diferenciação Numérica

Derivada por diferenças finitas

Figura: Derivada aproximada



Fonte: GILAT,(2008)

- Definição do Cálculo

$$\left.\frac{df(x)}{dx}\right|_{x=a} = f'(a) = lim_{x \to a} \frac{f(x) - f(a)}{x - a}$$

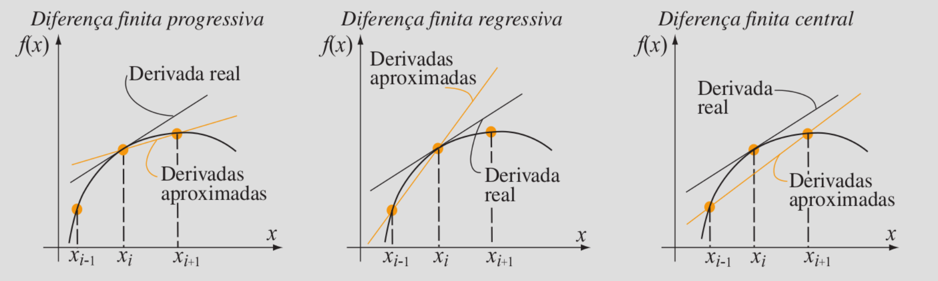
- É o valor da inclinação da **reta tangente** à função em $x = a$, obtida com a escolha de um ponto x próximo a $x = a$ e o cálculo da inclinação da reta que conecta os dois pontos.
- *Precisão* aumenta à medida que $x \to a$ tal que no imite em que o ponto x tende ao ponto a, a derivada é a inclinação da reta tangente a $f(x)$ em $x = a$.
- Na aproximação de derivadas utilizando diferenças finitas, valores da função em diferentes pontos na vizinhança do ponto $x = a$ são utilizados na estimativa da inclinação.

MN aplicados à Engenharia: Diferenciação Numérica

- Na aproximação de derivadas usando diferenças finitas, valores da função em diferentes pontos na vizinhança do ponto $x = a$ são usados na estimativa da inclinação.
- As fórmulas de diferenças finitas progressiva, regressiva e central são as mais simples aproximações da derivada por diferenças finitas.
- Diferença finita
    - *Progressiva* de x_{i+1} a x_i.
    - *Regressiva* de x_i a x_{i-1}.
    - *Central* de x_{i+1} a x_{i-1}.

MN aplicados à Engenharia: Diferenciação Numérica

Figura: Equações de diferença para a derivada primeira



Fonte: GILAT,(2008)

$$\left.\frac{df(x)}{dx}\right|_{x=x_i} = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}; \quad \left.\frac{df(x)}{dx}\right|_{x=x_i} = \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}; \quad \left.\frac{df(x)}{dx}\right|_{x=x_i} = \frac{f(x_{i+1}) - f(x_{i-1})}{x_{i+1} - x_{i-1}}$$

MN aplicados à Engenharia: Diferenciação Numérica

Exemplo I

Considere a função $f(x) = x^3$. Calcule numericamente a derivada primeira no ponto $x = 3$ aplicando as fórmulas de diferenças finitas progressiva, regressiva e central, usando:

(a) os pontos $x = 2$, $x = 3$ e $x = 4$.

Compare os resultados com a derivada exata (analítica).

- Diferenciação analítica: $f'(x) = 3.x^2 = f'(3) = 3.3^2 = 27$

(a) Diferenciação numérica:

* Diferenciação finita progressiva:

$$\frac{df(x)}{dx}\bigg|_{x=3} = \frac{f(4) - f(3)}{4 - 3} = \frac{4^3 - 3^3}{1} = 37 \rightarrow Erro = \left|\frac{37 - 27}{27}.100\right| = 37,04\%$$

* Diferenciação finita regressiva:

$$\frac{df(x)}{dx}\bigg|_{x=3} = \frac{f(3) - f(2)}{3 - 2} = \frac{3^3 - 2^3}{1} = 19 \rightarrow Erro = \left|\frac{19 - 27}{27}.100\right| = 29,63\%$$

* Diferenciação finita central:

$$\frac{df(x)}{dx}\bigg|_{x=3} = \frac{f(4) - f(2)}{4 - 2} = \frac{4^3 - 2^3}{2} = 28 \rightarrow Erro = \left|\frac{28 - 27}{27}.100\right| = 3,704\%$$

- $Nota : Erro_{relativo} = \left|\frac{Y_{aproximado} - Y_{real}}{Y_{real}}.100\right| = Y\%$

MN aplicados à Engenharia: Diferenciação Numérica

Exemplo I

Considere a função $f(x) = x^3$. Calcule numericamente a derivada primeira no ponto $x = 3$ aplicando as fórmulas de diferenças finitas progressiva, regressiva e central, usando:

(b) os pontos $x = 2,75$, $x = 3$ e $x = 3,25$.

Compare os resultados com a derivada exata (analítica).

- Diferenciação analítica: $f'(x) = 3.x^2 = f'(3) = 3.3^2 = 27$

(b) Diferenciação numérica:

* Diferenciação finita progressiva:
$$\left.\frac{df(x)}{dx}\right|_{x=3} = \frac{f(3,25) - f(3)}{3,25 - 3} = \frac{3,25^3 - 3^3}{0,25} = 29,3125 \rightarrow Erro = \left|\frac{29,3125 - 27}{27}.100\right| = 8,565\%$$
* Diferenciação finita regressiva:
$$\left.\frac{df(x)}{dx}\right|_{x=3} = \frac{f(3) - f(2,75)}{3 - 2,75} = \frac{3^3 - 2,75^3}{1} = 24,8125 \rightarrow Erro = \left|\frac{24,8125 - 27}{27}.100\right| = 8,102\%$$
* Diferenciação finita central:
$$\left.\frac{df(x)}{dx}\right|_{x=3} = \frac{f(3,25) - f(2,75)}{3,25 - 2,75} = \frac{3,25^3 - 2,75^3}{3,25 - 2,75} = 27,0625 \rightarrow Erro = \left|\frac{27,0625 - 27}{27}.100\right| = 0,2315\%$$

- $Nota : Erro_{relativo} = \left|\frac{Y_{aproximado} - Y_{real}}{Y_{real}}.100\right| = Y\%$

Conclusão: Erro depende fortemente da distância entre os pontos!

MN aplicados à Engenharia

Figura: Solução computacional

```
X=[2,3,4]
ponto_p=3                       # Função filtro para localizar o ponto_p na lista
i=filter(lambda n:n == ponto_p,X)  # Índice de locação do ponto na lista X
i=X.index(ponto_p)
f=lambda x:x**3                 # Função de análise f(x)=x³
df = lambda x:3*x**2            # Derivada Analítica incluída manualmente

der = df(ponto_p)               # Derivada numérica
dprog  =( f(X[i+1]) - f(X[i]) ) / (X[i+1]-X[i])
dcentr =( f(X[i+1]) - f(X[i-1]))/ (X[i+1]-X[i-1])
dregr  =( f(X[i]) - f(X[i-1]) ) / (X[i]-X[i-1])

print('-'*15),print("  RESULTADOS"),print('-'*15)
print("Dif.Analítica \tDif.Prog\tDif.Central\tDif.Regr")
print(f" \t{der}\t  {dprog}         \t  {dcentr}     \t   {dregr}")

# ***************************************************************************
# Resultados do erros
E=[]
E.append(round(abs(dprog-der)/der,2))
E.append(round(abs(dcentr-der)/der,2))
E.append(round(abs(dregr-der)/der,2))
print(f"\n Erros:\t\t {E[0]*100}%        \t  {E[1]*100}%     \t   {E[2]*100}%")
```

Fonte: Autor,(2020)

Figura: Resultados

```
---------------
  RESULTADOS
---------------
Dif.Analítica   Dif.Prog        Dif.Central     Dif.Regr
        27       37.0             28.0            19.0

 Erros:         37.0%            4.0%            30.0%
```

Fonte: Autor,(2020)

(*)Dados de entrada: $x = [2\ 3\ 4]$ e $p = 3$

MN aplicados à Engenharia

Em um experimento de vibração, um bloco de massa m é preso a uma mola com dureza k e a um amortecedor com coeficiente de amortecimento c. Para que o experimento tenha início, o bloco é retirado da posição de equilíbrio e solto. A posição do bloco em função do tempo é gravada em uma freqüência de 5 Hz (5 vezes por segundo). Os dados no intervalo $4 \leq t \leq 8s$ são dados dispostos nas colunas 1 (x-(cm)) e coluna 2(t-(s)) da matriz ao lado:

Pede-se:

(a) A velocidade do bloco é a derivada da posição em relação ao tempo. Use a fórmula de diferença finita central para calcular a velocidade nos tempos $t = 5s$ e $t = 6s$.

(b) Escreva uma função no Matlab/Octave (Python) que calcule a derivada de uma função descrita por um conjunto de pontos. A função deve calcular a derivada no primeiro e no último ponto usando as fórmulas de diferenças finitas progressiva e regressiva, respectivamente, e usando a fórmula de diferença finita central nos demais pontos.

Figura: x(cm) e t(s)

-5.87000	4.00000
-4.23000	4.20000
-2.55000	4.40000
-0.89000	4.60000
0.67000	4.80000
2.09000	5.00000
3.31000	5.20000
4.31000	5.40000
5.06000	5.60000
5.55000	5.80000
5.78000	6.00000
5.77000	6.20000
5.52000	6.40000
5.08000	6.60000
4.46000	6.80000
3.72000	7.00000
2.88000	7.20000
2.00000	7.40000
1.10000	7.60000
0.23000	7.80000
-0.59000	8.00000

Fonte: (Autor,2020)

MN aplicados à Engenharia

(a) Calculando a velocidade pela diferenciação finita central

t=5s $$\frac{df(x)}{dx}\Big|_{x=xi} = \frac{f(x_{i+1}) - f(x_{i+1})}{x_{i+1} - x_{i-1}} = \frac{df(x)}{dx}\Big|_{x=5} = \frac{f(5,2) - f(4,8)}{5,2 - 4,8} = \frac{3,31 - 0,67}{0,4} = 6,6\ cm/s$$

t=6s $$\frac{df(x)}{dx}\Big|_{x=xi} = \frac{f(x_{i+1}) - f(x_{i+1})}{x_{i+1} - x_{i-1}} \frac{df(x)}{dx}\Big|_{x=6} = \frac{f(6,2) - f(5,8)}{6,2 - 5,8} = \frac{5,77 - 5,55}{0,4} = 0,55\ cm/s$$

(b) - Script

MN aplicados à Engenharia

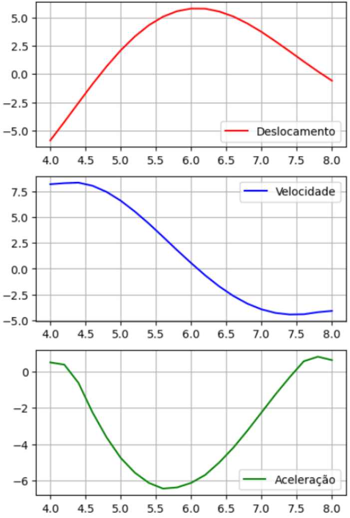
Figura: Script e gráficos

```python
# Solução utilizando vetor de zeros e atualizando-o
import numpy as np
import matplotlib.pyplot as plt
#Dados
t=np.arange(4,8.2,0.2)
x=np.array([-5.87, -4.23, -2.55, -0.89, 0.67, 2.09, 3.31, 4.31,
            5.06, 5.55, 5.78, 5.77, 5.52, 5.08, 4.46, 3.72, 2.88,
            2.00,1.10, 0.23, -0.59])

def derive(X,Y):                                # Derivadas
    dx=np.zeros(len(x))                         # Vetores de derivadas
    i=0                                         # Primeiro elemento do vetor
    dx[i] = (Y[i+1] - Y[i] ) / (X[i+1]-X[i])   # Progressiva
    i=-1                                        # Último elemento do vetor
    dx[-1] = (Y[i] - Y[i-1] ) / (X[i]-X[i-1])  # Regressiva
    for i in range(1,len(x)-1):                 # Elementos intermediários
        dx[i] = (Y[i+1] - Y[i-1])/ (X[i+1]-X[i-1]) # Central
    return dx

def graf(t,x,vel,acc):
    labels=['Deslocamento','Velocidade','Aceleração']
    colors=['r','b','g']
    plt.figure(figsize=(5,8))
    for i,j in enumerate([x,vel,acc]):
        plt.subplot(3,1,i+1)
        plt.plot(t,j,label = labels[i],color=colors[i])
        plt.legend(),plt.grid(True)

vel,acc = derive(t,x), derive(t,derive(t,x))
graf(t,x,vel,acc)
```



Fonte: (Autor,2024)

MN aplicados à Engenharia

Diferenciação Numérica usando a expansão da série de Taylor

- A partir da expansão em série de Taylor pode-se deduzir as fórmulas de diferenças finitas progressiva, regressiva e central.
- Essas fórmulas fornecem uma estimativa da derivada em um ponto usando valores de pontos em sua vizinhança.
- O número de pontos usados nos cálculos varia com a fórmula, e os pontos podem estar à frente, atrás ou em ambos os lados do ponto onde se calcula a derivada.
- Uma vantagem do uso da expansão em série de Taylor na dedução das fórmulas está no fato de ela fornecer uma estimativa do **erro de truncamento** presente na aproximação.

MN aplicados à Engenharia

Diferença finita progressiva com dois pontos para a derivada primeira

- $f(x_{i+1})$ aproximado pela série de Taylor em termos do valor da função e de suas derivadas no ponto xi resulta em:

$$f(x_{i+1}) = f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2!}h^2 + \frac{f'''(x_i)}{3!}h^3 + \frac{f^{(4)}(x_i)}{4!}h^4 + \ldots$$

  $h = x_{i+1} - x_i$ é a distância entre os pontos.

- Reescrevendo as equações com **dois termos e um resíduo**(ξ) (valor entre x_i e x_{i+1}) tem-se:

$$f(x_{i+1}) = f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2!}h^2$$

- Isolando-se o **termo de interesse** $f'(x_i)$ tem-se:

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h} - \frac{f''(\xi)}{2!}h$$

- Um valor aproximado de $f'(x_i)$ introduz um erro de truncamento (discretização). Declarando como zero o termo de $f(\xi)$ o erro de truncamento é da ordem de h(escrito como $O(h)$), por ser este proporcional àquele.

MN aplicados à Engenharia

Diferença finita progressiva com dois pontos para a derivada primeira

Portanto:

- **Erro de truncamento** - erro dado em função da distância entre pontos vale:

$$Erro\ de\ truncamento = \frac{f''(\xi)}{2!}h = O(h)$$

- **Aproximação** da derivada primeira finita progressiva por Série de Taylor

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h} - O(h)$$

MN aplicados à Engenharia

A mesma estrutura de aproximação para a derivada primeira é usada,portanto, para:

- Diferença finita regressiva com dois pontos resultando em:

$$f'(x_i) = \frac{f(x_i) - f(x_{i-1})}{h} - O(h)$$

- Diferença finita central com dois pontos resultando em:

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{h} - O(h^2)$$

Na aproximação por diferença central o erro de truncamento é da ordem de h^2. Isso indica que a aproximação por diferença central fornece uma aproximação mais precisa para a derivada!

MN aplicados à Engenharia

Aproximação de derivadas finitas por **Série de Taylor** - *Derivada primeira*

- **Derivada primeira com 2 pontos**
  - Diferença progressiva - $f'(x_i) = \dfrac{f(x_{i+1}) - f(x_i)}{h}$ com $O(h)$
  - Diferença regressiva - $f'(x_i) = \dfrac{f(x_i) - f(x_{i-1})}{h}$ com $O(h)$
  - Diferença central - $f'(x_i) = \dfrac{f(x_{i+1}) - f(x_{i-1})}{h}$ com $O(h^2)$
- **Derivada primeira com 3 pontos**
  - Diferença progressiva - $f'(x_i) = \dfrac{-3f(x_i) + 4f(x_{i+1}) - f(x_{i+2})}{2h}$ com $O(h^2)$
  - Diferença regressiva - $f'(x_i) = \dfrac{f(x_{i-2}) - 4f(x_{i-1}) + 3f(x_i)}{2h}$ com $O(h^2)$
- **Derivada primeira com 4 pontos** - necessita de x_{i-2} *a* x_{i+1}
  - Diferença central
    $f'(x_i) = \dfrac{f(x_{i-2}) - 8f(x_{i-1}) + 8f(x_{i+1}) - f(x_{i+2})}{12h}$ com $O(h^4)$

MN aplicados à Engenharia

Aproximação de derivadas finitas por **Série de Taylor** - *Derivada segunda*

- **Derivada segunda com 3 pontos**
  - Diferença progressiva - $f''(x_i) = \dfrac{f(x_i) - 2f(x_{i+1}) + f(x_{i+2})}{h^2}$ com $O(h)$
  - Diferença regressiva - $f''(x_i) = \dfrac{f(x_{i-2}) - 2f(x_{i-1}) + f(x_i)}{h^2}$ com $O(h)$
  - Diferença central - $f''(x_i) = \dfrac{f(x_{i-1}) - 2f(x_i) + f(x_{i+1})}{h^2}$ com $O(h^2)$
- **Derivada segunda com 4 pontos**
  - Diferença progressiva - $f''(x_i) = \dfrac{2f(x_i) - 5f(x_{i+1}) + 4f(x_{i+2}) - f(x_{i+3})}{h^2}$ com $O(h^2)$
  - Diferença regressiva - $f''(x_i) = \dfrac{-f(x_{i-3}) + 4f(x_{i-2}) - 5f(x_{i-1}) + 2f(x_i)}{h^2}$ com $O(h^2)$
- **Derivada segunda com 5 pontos** - central necessita de x_{i-2} *a* x_{i+2}
  - Diferença central $f''(x_i) = \dfrac{-f(x_{i-2}) + 16f(x_{i-1}) - 30f(x_i) + 16f(x_{i+1}) - f(x_{i+2}))}{12h^2}$ com $O(h^4)$

MN aplicados à Engenharia

Aproximação de derivadas finitas por **Série de Taylor** - *Derivada terceira*

- **Derivada terceira com 4 pontos**
  - Diferença progressiva - $f'''(x_i) = \dfrac{-f(x_i) + 3f(x_{i+1}) - 3f(x_{i+2}) + f(x_{i+3})}{h^3}$ com $O(h)$
  - Diferença regressiva - $f'''(x_i) = \dfrac{-f(x_{i-3}) + 3f(x_{i-2}) - 3f(x_{i-1}) + f(x_i)}{h^3}$ com $O(h)$
  - Diferença central - $f'''(x_i) = \dfrac{-f(x_{i-2}) + 2f(x_{i-1}) - 2f(x_{i+1}) + f(x_{i+2})}{2h^3}$ com $O(h^2)$
- **Derivada terceira com 5 pontos**
  - Diferença progressiva - $f'''(x_i) = \dfrac{-5f(x_i) + 18f(x_{i+1}) - 24f(x_{i+2}) + 14f(x_{i+3}) - 3f(x_{i+4})}{2h^3}$ com $O(h^2)$
  - Diferença regressiva - $f'''(x_i) = \dfrac{3f(x_{i-4}) - 14f(x_{i-3}) + 24f(x_{i-2}) - 18f(x_{i-1}) + 5f(x_i)}{2h^3}$ com $O(h^2)$
- **Derivada terceira com 6 pontos** - central necessita de x_{i-2} *a* x_{i+3}
  - Diferença central $f'''(x_i) = \dfrac{f(x_{i-3}) - 8f(x_{i-2}) + 13f(x_{i-1}) - 13f(x_{i+1}) + 8f(x_{i+2}) - f(x_{i+3}))}{8h^3}$ com $O(h^4)$

MN aplicados à Engenharia

Exemplo comparativo

MN aplicados à Engenharia

Considere a função $f(x) = x^3$. Calcule numericamente a derivada primeira no ponto $x = 3$ aplicando as fórmulas de diferenças finitas progressiva, regressiva e central, usando:

(a) os pontos $x = 3$, $x = 4$ e $x = 5$.

(b) os pontos $x = 3$, $x = 3,25$ e $x = 3,5$.

Compare os resultados com a derivada exata (analítica).

- Diferenciação analítica: $f'(3) = 3.3^2 = 27$
- Diferenciação numérica com três pontos:

(a) os pontos $x = 3$, $x = 4$ e $x = 5$.

$$\left.\frac{df(x)}{dx}\right|_{x=3} = \frac{-3f(x_i) + 4f(x_{i+1}) - f(x_{i+2})}{2h} = \frac{-3f(3) + 4f(4) - f(5)}{2(4-3)} = \frac{-3.(27) + 4.(64) - (125)}{2(4-3)} = 25 \rightarrow$$

$$Erro = \left|\frac{25-27}{27}.100\right| = 7,4\%$$

(b) os pontos $x = 3$, $x = 3,25$ e $x = 3,5$.

$$\left.\frac{df(x)}{dx}\right|_{x=3} = \frac{-3f(x_i) + 4f(x_{i+1}) - f(x_{i+2})}{2h} = \frac{-3f(3) + 4f(3,25) - f(3,5)}{2(3,25-3)} = \frac{-3.(27) + 4.(3,25^3) - (3,5^3)}{2(0,25} =$$

$$26,875 \rightarrow Erro = \left|\frac{26,875-27}{27}.100\right| = 0,46\%$$

MN aplicados à Engenharia

Considere a função $f(x) = x^3$. Calcule numericamente a derivada primeira no ponto $x = 3$ aplicando as fórmulas de diferenças finitas progressiva, regressiva e central, usando:

- Para os pontos $x = 3$, $x = 4$ e $x = 5$, tem-se:
  - Resultado de 37 com erro de $37,04\%$ - Diferença finita progressiva com 2 pontos
  - Resultado de 25 com erro de $7,04\%$ - Diferença finita progressiva com 3 pontos
- Para os pontos $x = 3$, $x = 3,25$ e $x = 3,5$, tem-se:
  - Resultado de $26,875$ com erro de $0,46\%$ - Diferença finita progressiva com 2 pontos
- Os resultados mostram que a diferença finita progressiva com três pontos fornece uma derivada **primeira** muito mais precisa do que a diferença finita progressiva com dois pontos.
- Para $h = 1$, o erro cai de $37,04\%$ para $7,4\%$.
- Para $h = 0,25$, o erro cai de $8,57\%$ para $0,46\%$.

MN aplicados à Engenharia

Considere a função $f(x) = \frac{2^x}{x}$. Calcule numericamente a derivada segunda no ponto $x = 2$ aplicando a fórmula de diferença central com três pontos, usando:

(a) os pontos x = 1,8, x = 2 e x = 2,2.

(b) os pontos x = 1,9, x = 2 e x = 2,1.

- **Solução analítica** - A derivada segunda da função $f(x) = \frac{2^x}{x}$ é

$$f''(x) = \frac{2^x[ln(2)^2]}{x} - \frac{2.2^x[ln(2)]}{x^2} + \frac{2.2^x}{x^3}$$

  de modo que o valor da derivada em $x = 2$ é **f"(2) = 0,5746**.

- **Diferenciação numérica**

Figura: Solução python $f''(x)$

```
import sympy as sp
f=lambda x:(2**x)/x
def derivada(k,ordem=1):
        x = sp.Symbol('x')
        fun = k(x)
        dh=sp.diff(fun,x,ordem)
        return sp.lambdify(x,dh,'numpy')
df = derivada(f,2)
df(2)
```

```
0.5746116667165122
```

Fonte: Autor,(2024)

MN aplicados à Engenharia

Considere a função $f(x) = \frac{2^x}{x}$. Calcule numericamente a derivada segunda no ponto $x = 2$ aplicando a fórmula de diferença central com três pontos, usando:

(a) os pontos x = 1,8, x = 2 e x = 2,2.

(b) os pontos x = 1,9, x = 2 e x = 2,1.

- **Diferenciação numérica**

Figura: Solução Python itens ‘a’ e ‘b’

```
a,b =[1.8,2,2.2],[1.9,2,2.1]
def der_st(X):
    h=X[0]-X[1]
    y=[2**x/x for x in X]
    df=(y[0]-2*y[1]+y[2])/h**2 # central
    return round(df,6)
print(der_st(a), der_st(b))
```

```
0.577482 0.575324
```

Fonte: Autor,(2024)

- Erro do item (a):
  $erro = \frac{0{,}577482-0{,}5746}{0{,}5746}.100 = \mathbf{0{,}5016\%}$
- Erro do item (b):
  $erro = \frac{0{,}575324-0{,}5746}{0{,}5746}.100 = \mathbf{0{,}126\%}$

Os resultados mostram que a fórmula da derivada central com três pontos fornece uma aproximação bastante precisa para o valor da derivada segunda.

Organização

MN aplicados à Engenharia

- Apresentar conteúdo de Integração Numérica
  - Método do retângulo, ponto central e trapézio
  - Método 1/3 de Simpson e 3/8 de Simpson

MN aplicados à Engenharia

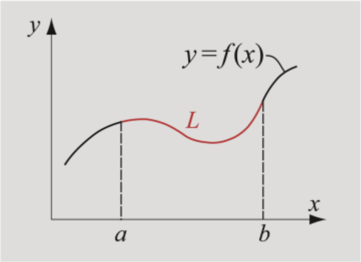
Exemplo de uso

(a) Comprimento de uma curva: $L = \int_a^b \sqrt{1 + [f'(x)]^2} dx$

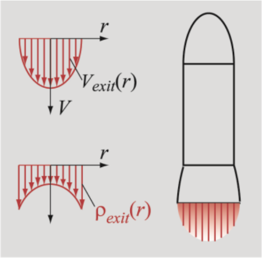
(b) Impulsão de um foguete: $T = \int_0^R 2\pi\rho(r) V_{saida}^2(r) dr$

Figura: Aplicação de integração numérica

(a) Comprimento de curva



(b) Impulsão de foguete



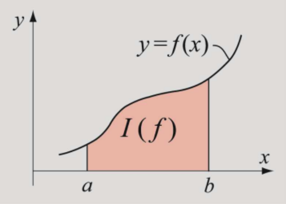
Fonte: GILAT,(2008)

MN aplicados à Engenharia

Introdução

- Seja f(x) uma função contínua no intervalo $[a, b]$, então existe a função primitiva $F(x)$, tal que $F'(x) = f(x)$.
- O valor da integral definida de $f(x)$ pode ser calculado usando (Fórmula de Newton-Leibniz):

Figura: $Integral\ definida(antiderivada)$



Fonte: GILAT,(2008)

$$I(f) = \int_a^b f(x)dx = F(b) - F(a)$$

onde $f(x)$, chamada de integrando, é função da variável independente x, enquanto a e b são os limites de integração. Graficamente, o valor da integral corresponde à área sombreada sob a curva de $f(x)$ entre a e b.

MN aplicados à Engenharia

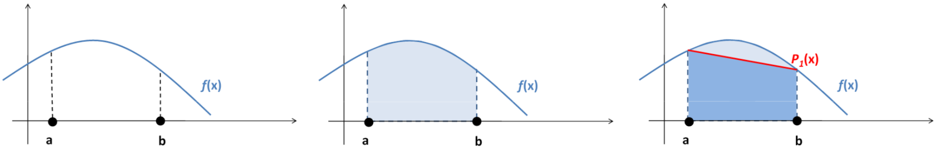
Introdução

- A função $f(x)$ é aproximada por uma função $p(x)$, mais simples cuja primitiva é mais fácil de se obter.

$$I = \int_a^b f(x)dx \approx \int_a^b p(x)dx$$

Figura: Delimitação de intervalo e área entre $[a, b]$

(a) Intervalo (b) Área do intervalo (c) Limitação em $p_1(x)$



Fonte: DIAS,(2019)

I $\approx$ área delimitada por $P_1(x)$ no intervalo $[a, b]$

MN aplicados à Engenharia

Introdução

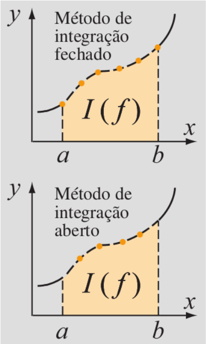
- Os métodos de integração **numérica** aproximam valores de integrais definidas.
- A integração numérica é útil quando:
    - * Não se conhece a função $f(x)$. Tem-se apenas uma tabela de valores para f.
    - * f é conhecida, mas é muito complexa, o que dificulta a determinação da primitiva.
    - A ideia básica é substituir $f(x)$ por um polinômio que a aproxime razoavelmente no intervalo $[a, b]$.
    - Assim o problema se resume à integração de polinômios, o que é trivial de realizar.

MN aplicados à Engenharia

Introdução Existem vários métodos disponíveis para o cálculo numérico de integrais. Em cada um desses métodos, uma fórmula é deduzida para calcular o valor aproximado de uma integral a partir dos pontos fechado discretos do integrando.

- Métodos fechados - os pontos finais do intervalo (e o integrando) são usados na fórmula que estima o valor da integral.
  - Método trapezoidal
  - Método de Simpson
- Métodos abertos - o intervalo de integração se estende além do limite especificado pelos pontos finais
  - Método do ponto central

Figura: Métodos de integração



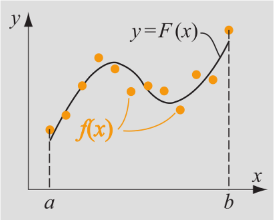
Fonte: GILAT,(2008)

MN aplicados à Engenharia

Fórmula de Newton-Cotes

- Há vários métodos que podem ser usados para calcular o valor de uma integral a partir dos pontos discretos do integrando. Dentre estes, as fórmulas de Newton-Cotes são as mais comumente empregadas.
  - O valor do integrando é estimado entre os pontos discretos empregando-se uma função de fácil integração, analítica ou interpolada a partir de pontos discretos do integrando original.
- Seja $f(x)$ uma função definida por um conjunto discreto de pontos é possível obter uma curva $F(x)$ que melhor se ajuste a estes pontos.

Figura: Fórmula de Newton-Cotes



Fonte: GILAT,(2008)

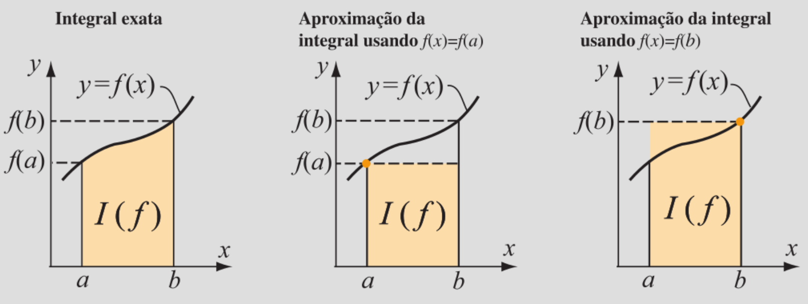
$$I(f) = \int_a^b f(x)dx \approx \int_a^b F(x)dx$$

MN aplicados à Engenharia

Método do Retângulo Simples

- Consiste em assumir que $f(x)$ é uma constante dentro do intervalo $[a, b]$.

Figura: Método do Retângulo



Fonte: GILAT,(2008)

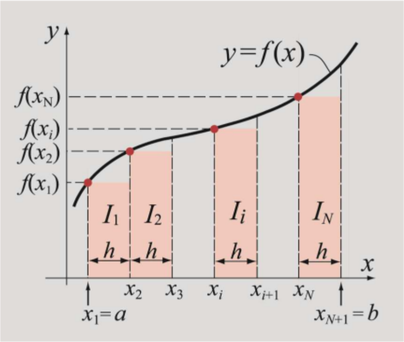
$$I(f) = \int_a^b f(x)dx = f(a)(b-a) \approx f(b)(b-a)$$

MN aplicados à Engenharia

Método do Retângulo Composto

- Visa reduzir o erro apresentado no método do retângulo, criando N subintervalos.

Figura: Método do Retângulo Composto



Fonte: GILAT,(2008)

$$I(f) = \int_a^b f(x)dx \approx h \sum_{i=1}^{N} f(x_i)$$

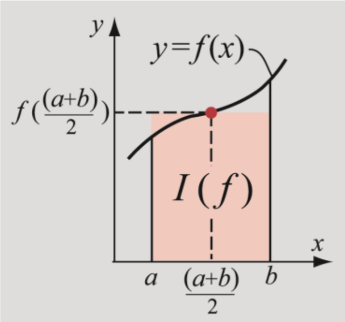
(*) Quando os subintervalos têm a mesma largura h, doutro modo os várias intervalos permanecem desagrupados.

MN aplicados à Engenharia

Método do Ponto Central

- É uma melhoria em relação ao Método do Retângulo, onde ao invés de aproximar o integrando usando os valores da função em $x = a$ ou $x = b$, utiliza-se o valor do integrando no meio do intervalo $x = \frac{(a+b)}{2}$.

Figura: Método do Ponto Central



Fonte: GILAT,(2008)

$$I(f) = \begin{cases} \int_a^b f(x)dx & = \\ \int_a^b f(\frac{a+b}{2})dx & = \\ f(\frac{a+b}{2}).(b-a) \end{cases}$$

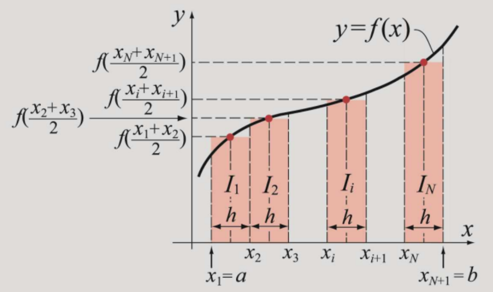
(*) Este método é mais preciso do que o método do retângulo simples porque, para uma função crescendo (conforme figura), as regiões da área abaixo da curva que são ignoradas podem ser compensadas de forma aproximada pelas regiões incluídas acima da curva.

MN aplicados à Engenharia

Método do Ponto Central Composto

- É uma melhoria em relação ao Método do Ponto Central.

Figura: Método do Ponto Central Composto



Fonte: GILAT,(2008)

$$I(f) = \int_a^b f(x)dx =$$

$$I(f) = h\sum_{i=1}^{N} f\left(\frac{x_i + x_{(i+1)}}{2}\right)$$

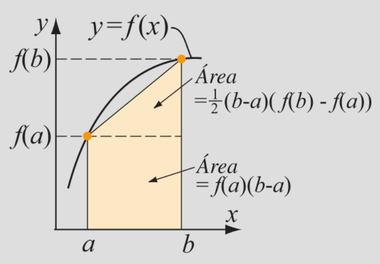
A integral em cada subintervalo é calculada usando o método do ponto central, e o valor total da integral é obtido com a soma dos valores das integrais obtidos nos subintervalos.

MN aplicados à Engenharia

Método do Trapézio Simples

- A ideia da regra do trapézio é aproximar a função $f(x)$ por um polinômio de 1 ordem (reta).
- Nessa aproximação a integral da função $f(x)$ pode ser aproximada pela área de um trapézio.

Figura: Método simples



Fonte: GILAT,(2008)

- O uso do polinômio interpolador de Newton entre os pontos $x = a$ e $x = b$ resulta em:

$*$ $f(x) \approx f(a) + (x-a)f[a,b] = f(a) + (x-a)\dfrac{[f(b)-f(a)]}{b-a}$

- Realizando a integração analítica:
  - $$I(f) \approx \int_a^b \left(f(a) + (x-a)\frac{[f(b)-f(a)]}{b-a} \right) dx$$
  - $$I(f) \approx f(a)(b-a) + \frac{1}{2}[f(b)-f(a)](b-a)$$
- Simplificando : $I(f) \approx \dfrac{[f(a)+f(b)]}{2}(b-a)$

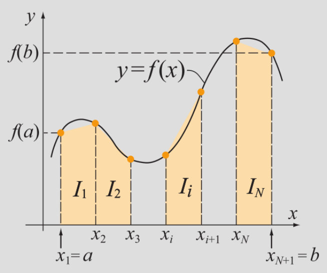
Neste método, a área abaixo da curva f(x) é aproximada pela área de um trapézio (retângulo + triângulo)

MN aplicados à Engenharia

Método do Trapézio Composto

- A integral ao longo do intervalo $[a, b]$ pode ser avaliada de forma mais precisa com a **subdivisão** do intervalo, a avaliação da integral em cada um dos subintervalos (com o método trapezoidal) e a **soma** dos resultados.
- O primeiro ponto é $x_1 = a$ e o último ponto é $x_{N+1} = b$ (são necessários $N + 1$ pontos para definir N intervalos).

Figura: Método composto



Fonte: GILAT,(2008)

- A aplicação do método trapezoidal em cada subintervalo $[x_i, x_{i+1}]$ resulta em:
  * $I_i(f) = \int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{[f(x_i) + f(x_{i+1})]}{2}(x_{i+1} - x_i)$
- Substituindo a aproximação trapezoidal:
  - $I(f) = \int_a^b f(x)dx \approx \frac{1}{2}\sum_{i=1}^{N}[f(x_i) + f(x_{i+1})](x_{i+1} - x_i)$
- Simplificando : $I(f) \approx \frac{h}{2}\sum_{i=1}^{N}[\ f(x_{i+1}) + f(x_i)\ ]$
- Programação: $I(f) \approx \frac{h}{2}[f(a)+f(b)]+h\sum_{i=1}^{N} f(x_i)$, caso h tenha larguras idênticas.

MN aplicados à Engenharia

Exemplos adaptado de gilat- Distância percorrida por um avião em desaceleração

Um Boeing de massa $m = 97000$ kg aterrissa a uma velocidade de $93m/s$ e liga os seus reversos em $t = 0$. A força F aplicada no avião à medida que ele reduz a sua velocidade é dada por $F = -5v^2 - 570000$, onde v é a velocidade do avião. Usando a segunda lei de Newton do movimento e da dinâmica dos fluidos, a relação entre a velocidade e a posição x do avião pode ser escrita como: $mv\frac{dv}{dx} = -5v^2 - 570000$, onde x é a distância medida a partir da localização do jato em $t = 0$. Determine a distância percorrida pelo avião antes que sua velocidade se reduza a $40m/s$ usando o *método trapezoidal composto*.

- **Solução analítica:**
  - Isolando dx tem-se: $\dfrac{97000vdv}{(-5v^2 - 570000)} = dx$
  - Integrando em ambos os lados tem-se:
    $$\int_0^x dx = -\int_{93}^{40} \frac{97000v}{5v^2 + 570000}dv = \int_{40}^{93} \frac{97000v}{5v^2 + 570000}dv$$
  - Resolvendo a integral tem-se: $z = 5u^2 + 570000$ e, portanto, $x = 574,1494m$

MN aplicados à Engenharia

Exemplo adaptado de Gilat,(2008) : Distância percorrida por um avião em desaceleração

Figura: Solução computacional

```
clc; clear all;
%**************************
f= @(v)(97000*v/(5*v^2+570000)); % função
a= 40;                          %limite inferior
b= 93;                          %limite superior
Ns=[10 100 1000];               %Intevalos

for(j=1:3)
  N=Ns(j);
  %***************************
  h=(b-a)/N;                 % largura dos intervalos
  x=a:h:b;                   % vetor x de coordenadas dos subintervalos
  for(i=1:N+1)
    F(i)=f(x(i));            % vetor f(x) de imagem das coordenadas
  end
  %**************************
   I(j)=h*(F(1)+F(N+1))/2 + h*sum(F(2:N)); % Método trapezoidal
end
%**************************
g=int(sym(f));clc;           % Solução analítica
dg=matlabFunction(g);        % Def. antideriv. de f.
sx=dg(b)-dg(a)
%**************************
for(k=1:length(I))           % Erros
Er(k) = abs( (I(k)-sx)*100/sx );
end
%**************************
sx =  574.15                     % Solução analítica
I = 574.09    574.15    574.15      % Solução m.trapezio N=[10 100 1000]
Er = 1.1134e-02   1.1134e-04   1.1134e-06 % Erros em percentual
```

Fonte: Autor,(2020)

MN aplicados à Engenharia

Exemplo adaptado de gilat- Distância percorrida por um avião em desaceleração

Resultado da integral por método "analítico": 574.14941316747

Tabela: Resultado dos métodos

Método	**Resultado da integral**	**Erro**(%)
Retângulo $f(x) = f(a)$	355.778546712803	38.0338046937918
Retângulo $f(x) = f(b)$	779.644350952719	35.7911953008144
Ponto central	577.385584212426	0.563646146932926
Trapézio simples	567.711448832761	1.1213046964887
Retângulo composto	572.02944491021	0.369236336159371
Ponto central composto	574.149732785691	0.000055668126401556
Trapézio composto	574.148773931409	0.000111336186410645

Fonte: Autor,(2020)

Métodos compostos usando N=100.

MN aplicados à Engenharia

Exemplo adaptado de gilat- Distância percorrida por um avião em desaceleração

Figura: Scritp comparativo dos métodos

```
clc; clear all;
%****************************
f=@(x)(97000*x/(5*x.^2+570000)); a=40;b=93;  % Função e limites
for(j=1:7)                                  % 7 métodos
   N=100;h=(b-a)/N;x=a:h:b;                 % Intervalos, largura e subintervalos
   soma=0;                                  % Acumulador de soma
   if (j==1) I(j)=f(a)*(b-a);end            % RETÂNGULO SIMPLES f(x)=f(a)
  if (j==2) I(j)=f(b)*(b-a);end             % RETÂNGULO SIMPLES f(x)=f(b)
  if (j==3) I(j)=f((a+b)/2)*(b-a);end       % PONTO CENTRAL     f(x)=f((a+b)/2)
  if (j==4) I(j)=((f(a)+f(b))/2)*(b-a);end  % TRAPÉZIO
  if (j==5)                                 % RETÂNGULO COMPOSTO
    for(i=1:N) soma=soma+f(x(i))*h; end;I(j)=soma; end
  if(j==6)                                  % PONTO CENTRAL COMPOSTO
    for(i=1:N) soma=soma+f((x(i)+ x(i+1))/2)*h; end; I(j)=soma; end
  if(j==7) ;                                % TRAPÉZIO COMPOSTO
  for(i=1:N) soma=soma + 0.5*(f(x(i))+f(x(i+1)))*(x(i+1)-x(i)); end;
    I(j)=soma; end
end

g=int(sym(f));  dg=matlabFunction(g);   % SOLUÇÃO ANALÍTICA - antideriv. de f.
sx=dg(b)-dg(a);
for(k=1:length(I)) Er(k) = abs( (I(k)-sx)*100/sx ); end;  % Erros
clc;
fprintf('Valor da solução analítica %f.\n',sx)      % Resultados
fprintf('\t\t  Métodos    Valores dos métodos \t Erros\n')
Val_Er=[(1:7)' I' Er']
```

Fonte: Autor,(2020)

MN aplicados à Engenharia

Métodos de Simpson

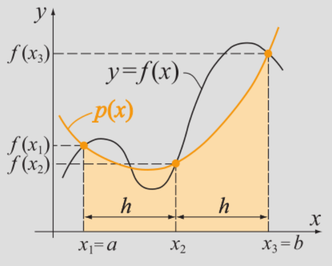
- O método trapezoidal aproxima o integrando por uma linha reta. Em tese, uma melhor aproximação pode ser obtida com a representação do integrando como uma função não-linear de fácil integração.
- Uma classe de métodos com essa característica (regras ou métodos de Simpson) usa para aproximar o integrando polinômios:
    - quadráticos (método de Simpson 1/3)
    - cúbicos (método de Simpson 3/8)

MN aplicados à Engenharia

Método de 1/3 de Simpson Simples

- Um polinômio quadrático (de segunda ordem) é usado para aproximar o integrando
- Os coeficientes do polinômio quadrático são determinados a partir de três pontos: $x_1 = a$, $x_2 = (a+b)/2$ e $x_3 = b$ tal que $p(x) = \alpha + \beta(x - x_1) + \gamma(x - x_1)(x - x_2)$ onde α,β e γ são constantes desconhecidas tais que o polinômio deve passar por todos os pontos, $p(x_1) = f(x_1), p(x_2) = f(x_2)$ e $p(x_3) = f(x_3)$.
- O que resultado em: $\alpha = f(x_1)$,$\beta = [f(x_2) - f(x_1)]/(x_2 - x_1)$ e $\gamma = \dfrac{f(x_3) - 2f(x_2) + f(x_1)}{2(h)^2}$, onde $h = (b-a)/2$
- E, portanto: $I = \int_{x_1}^{x_3} f(x)dx \approx \int_{x_1}^{x_3} p(x)dx =$ $I = \dfrac{h}{3}[f(x_1) + 4f(x_2) + f(x_3)]$

Figura: Simpson 1/3



Fonte: GILAT,(2008)

* Logo: $I = \dfrac{h}{3}[f(a) + 4f(\dfrac{a+b}{2}) + f(b)]$

O nome 1/3 no método vem do fator de 1/3 multiplicando a expressão entre colchetes.

MN aplicados à Engenharia

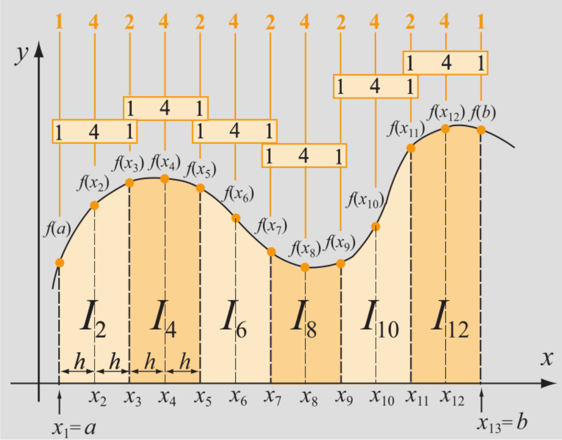
Método de 1/3 de Simpson Composto

- No método de Simpson 1/3 composto divide-se o intervalo $[a, b]$ em N subintervalos de largura h, onde $h = (b - a)/N$
- O método de Simpson 1/3 é aplicado em dois subintervalos adjacentes de cada vez, por serem necessários 3 pontos para definir um polinômio quadrático. Portanto, o intervalo total deve ser dividido em um *número par de subintervalos.*
- A integral ao longo de 2 intervalos adjacentes $[x_{i-1}, x_i]$ e $[x_i, x_{i+1}]$ fica definida como: $I_i(f) = \int_{x_{i-1}}^{x_{i+1}} f(x)dx = \frac{h}{3}[f(x_{i-1}) + 4f(x_i) + f(x_{i+1})]$, onde $h = x_{i+1} - x_i = x_i - x_{i-1}$
- Agrupados em termos similares resulta em:
$$I(f) \approx \frac{h}{3}\left[f(a) + 4\sum_{i=2,4,6}^{N} f(x_i) + 2\sum_{j=3,5,7}^{N-1} f(x_j) + f(b)\right], \text{ onde } h = (b-a)/N$$
- Condições necessárias
  - Os subintervalos devem ser **igualmente** espaçados.
  - O número de subintervalos no domínio $[a, b]$ deve ser um número **par**

MN aplicados à Engenharia

Método de 1/3 de Simpson Composto

Figura: Soma ponderada - Simpson 1/3 composto



Fonte: GILAT,(2008)

- $I(f) \approx \frac{h}{3}\left[f(a) + 4\sum_{i=2,4,6}^{N} f(x_i) + 2\sum_{j=3,5,7}^{N-1} f(x_j) + f(b)\right]$
- A equação é uma soma ponderada da função nos pontos que definem os subintervalos tal que o peso é 4 nos pontos x_i **pares** e 2 x_i nos **ímpares** (exceto o primeiro e o último ponto).
- Estes são os pontos centrais de cada *par de subintervalos adjacentes*.
- Cada ponto é usado uma vez como o ponto final à direita de um par de subintervalos, e uma vez como o ponto final à esquerda do par de subintervalos seguinte.

MN aplicados à Engenharia

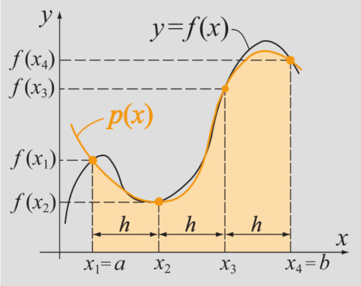
Método de 3/8 de Simpson

- Um polinômio cúbico (de terceira ordem) é usado para aproximar o integrando
- Os coeficientes do polinômio quadrático são determinados a partir dos pontos: $x_1 = a$,$x_4 = b$ e dois pontos x_2 e x_3 que dividem o intervalo em três seções iguais, tal que: $p(x) = c_3x^3 + c_2x^2 + c_1x + c_0$, onde c_3 , c_2 , c_1 e c_0 são constantes avaliadas a partir da condição que diz que passa pelos pontos $p(x_1) = f(x_1)$,$p(x_2) = f(x_2)$, $p(x_3) = f(x_3)$ e $p(x_4) = f(x_4)$.
- Resultando em:

$$I = \int_a^b f(x)dx \approx \int_a^b p(x)dx = \frac{3}{8}h[f(a)+3f(x_2)+3f(x_3)+f(b)]$$

- O valor da integral é mostrado como a **área sombreada** entre a curva $p(x)$ e o eixo x. Note que a equação é uma soma ponderada dos valores de $f(x)$ nos dois pontos finais $x_1 = a$ e $x_4 = b$, e nos pontos x_2 e x_3 que dividem o intervalo em três seções de largura(h) iguais.

Figura: Simpson 3/8



Fonte: GILAT,(2008)

* Logo:

$$I = \frac{3}{8}h[f(a) + 3f(x_2) + 3f(x_3) + f(b)]$$

O nome 3/8 no método vem do fator de 3/8 multiplicando a expressão entre colchetes.

MN aplicados à Engenharia

Método de 3/8 de Simpson Composto

- No método de Simpson 3/8 composto divide-se o intervalo $[a, b]$ em N subintervalos de largura h, em que $h = (b - a)/N$
- Sendo necessários 4 pontos para definir um **polinômio cúbico**, o método é aplicado em **três** subintervalos adjacentes de cada vez e, portanto, o número de subintervalos dever ser divisível por 3.
- A integral ao longo de **3 intervalos** adjacentes para 4 grupos fica definida como: $I(f) \approx \frac{3h}{8}(f(a) + 3[f(x_2) + f(x_3) + f(x_5) + f(x_6) + f(x_8) + f(x_9) + f(x_{11}) + f(x_{12})] + 2([f(x_4) + f(x_7) + f(x_{10})] + f(b))$
- No caso geral que envolve a divisão do domínio $[a, b]$ em N subintervalos (em que N é pelo menos 6 e divisível por 3), resulta em:

$$I(f) \approx \frac{3h}{8}\left[f(a) + 3\sum_{i=2,5,8}^{N-1}[f(x_i) + f(x_{i+1}]) + 2\sum_{j=4,7,10}^{N-2} f(x_j) + f(b)\right],$$

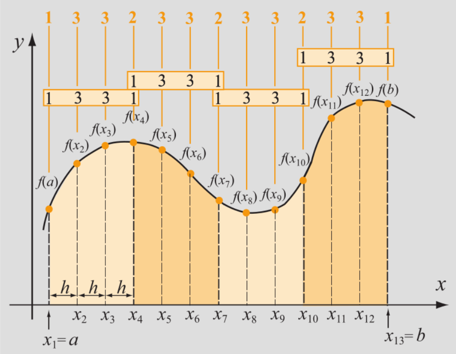
em que $h = (b - a)/N$

- Condições necessárias
  - Os subintervalos devem ser **igualmente** espaçados.
  - O número de subintervalos no domínio $[a, b]$ deve ser um número divisível por 3.

MN aplicados à Engenharia

Método de 3/8 de Simpson Composto

Figura: Soma ponderada - Simpson 3/8 composto



Fonte: GILAT,(2008)

- Uma combinação dos métodos de Simpson pode ser usada para realizar a integração quando houver um número ímpar qualquer de subintervalos.
- Isso é feito usando o método de Simpson 3/8 nos três primeiros $([a, x_2], [x_2, x_3]e[x_3, x_4])$ ou nos três últimos subintervalos $([x_{N-2}, x_{N-1}], [x_{N-1}, x_N]$ e $[x_N, x_b])$, aplicando-se o método de Simpson 1/3 no número restante (par) de subintervalos.

MN aplicados à Engenharia

Exemplo adaptado de gilat- Distância percorrida por um avião em desaceleração
Resultado da integral por método "analítico": 574.14941316747

Tabela: Resultado dos métodos - Simpson

Método	**Resultado da integral**	**Erro**(%)
1/3 Simpson	574.160872419205	0.00199586579239805
3/8 Simpson	574.154491982205	0.000884580671689455
1/3 Simpson Composto	574.149413169273	3.14101915880905e-10
3/8 Simpson Composto	574.149413171202	6.50024440177668e-10

Fonte: Autor,(2020)

Métodos compostos usando N=100.

MN aplicados à Engenharia

Figura: Scritp comparativo dos métodos

```
clc; clear all; pkg load symbolic; format long g
%**************************
f=@(x)(97000*x/(5*x.^2+570000)); a=40;b=93;  % Função e limites
for(j=1:4)                                   %
   N=100;h=(b-a)/N;x=a:h:b;                  % Intervalos, largura e subinterval
   %*******************************************************************
  if (j==1) tic;h=(b-a)/2;                         % 1/3 simpson
    I(j)=(h/3)*(f(a)+ 4*f((a+b)/2)+f(b) ); t(j)=toc; end
  if (j==2) tic;h=(b-a)/3; x2=a+h; x3=a+2*h; % 3/8 simpson
    I(j)=(3*h/8)*(f(a)+ 3*f(x2)+3*f(x3)+f(b)); t(j)=toc;end
  if (j==3) tic;
    part=0; part=0; if(mod(N,2)==0) N=N;else N=N+1;end    % N par
     h=(b-a)/N;x=a:h:b;                              % 1/3 simpson composto
     for(i=2:N)
        if (mod(i,2)==0) part=part+4*f(x(i)); else part=part+2*f(x(i));end
     end
     I(j)= (f(a)+part+f(b))*h/3;t(j)=toc;
  end
  if (j==4) tic;
      if(mod(N,3)==0) N=N;else                      % N div 3
      while(mod(N,3)!=0)
      N=N+1;  end; end
    part1=0;part2=0; h=(b-a)/N; x=a:h:b;           % 3/8 simpson composto
       for(i=2:3:N)
        part1=part1+3*(f(x(i))+f(x(i+1))); end
       for(i=4:3:N)
        part2=part2+2*(f(x(i)));           end
  I(j)= (f(a)+part1+part2+f(b))*h*3/8;
  t(j)=toc;
  end
end
%*******************************************************
g=int(sym(f));  dg=matlabFunction(g);   % SOLUÇÃO ANALÍTICA - antideriv. de f.
sx=dg(b)-dg(a);
for(k=1:length(I)) Er(k) = abs( (I(k)-sx)*100/sx ); end;clc;  % Erros
fprintf('\n\nValor da solução analítica %f.\n\n',sx)          % Resultados
fprintf(' Valores dos métodos \t Erros\n')
Val_Er=[ I' Er' t']
```

Fonte: Autor,(2020)