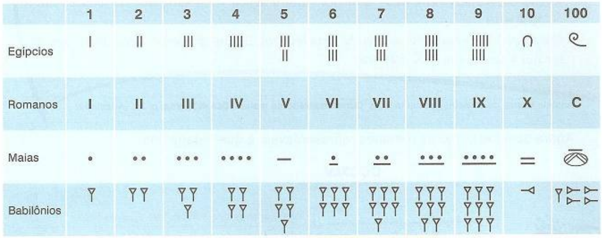
Universidade do Minho
Escola de Ciências

Departamento de Matemática

Aritmética Computacional

- Sistemas de numeração
- Sistemas de vírgula flutuante
- A norma IEEE 754
- Erros
- Estabilidade

1. Sistemas de numeração



Sistema de numeração atual:

➲ decimal ➲ posicional

Outras bases:

➠ binária ➠ octal ➠ hexadecimal

Representação de INTEIROS na base b (≥ 2)

Qualquer inteiro $N \neq 0$ tem uma representação única na forma

$$\begin{aligned} N &= (-1)^s (a_n a_{n-1} \ldots a_1 a_0)_b \\ &= (-1)^s (a_n \times b^n + a_{n-1} \times b^{n-1} + \cdots + a_1 \times b^1 + a_0 \times b^0) \end{aligned}$$

onde $s \in \{0, 1\}$, $a_i \in \{0, 1, \ldots, b-1\}$, $a_n \neq 0$.

Exemplo:

b	a_i															
2	0	1														
8	0	1	2	3	4	5	6	7								
16	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

$(123)_{10}$ $(1111011)_2$ $(173)_8$ $(7B)_{16}$

➲ Mudança da base b para a base 10:

- $(123)_8 = 1 \times 8^2 + 2 \times 8 + 3 = 83$
- $(2E)_{16} = 2 \times 16 + 14 = 46$

➲ Mudança da base 10 para a base b:

- Representação de 46 na base 8:

$$\begin{array}{ll} 46 & \underline{|8\ \ } \\ 6 & 5 \end{array} \qquad 46 = 5 \times 8 + 6 = (56)_8$$

- Representação de 46 na base 2:

$$\begin{array}{llllll} 46 & \underline{|2\ \ } & & & & \\ 06 & 23 & \underline{|2\ \ } & & & \\ 0 & 03 & 11 & \underline{|2\ \ } & & \\ & 1 & 1 & 5 & \underline{|2\ \ } & \\ & & & 1 & 2 & \underline{|2\ \ } \\ & & & & 0 & 1 \end{array}$$

$$\begin{aligned} 46 &= 2 \times 23 = 2 \times (11 \times 2 + 1) = \\ &= 11 \times 2^2 + 2 = (2 \times 5 + 1) \times 2^2 + 2 \\ &= \cdots = \\ &= (101110)_2 \end{aligned}$$

Como $8 = 2^3$ e $16 = 2^4$, a conversão

binário ↔ octal e binário ↔ hexadecimal

pode ser feita de forma quase imediata.

$46 = (5\ 6)_8 = (101\ 110)_2$

$46 = (2\ E)_{16} = (0010\ 1110)_2$

Exercícios: $(10111100110010)_2 = (\qquad)_8 \quad (3ED32)_{16} = (\qquad)_2$

Funções do MATLAB

`dec2bin` `dec2hex` `dec2base` `bin2dec` `hex2dec` `base2dec`

Representação binária - observações

- Num computador, o número de *bits* [1] disponíveis para representar inteiros determina qual o maior (e menor) inteiro representáveis.
- A representação binária do número 46 necessita de 6 bits.
- A representação de inteiros negativos em computador pode fazer-se reservando um bit para o sinal (normalmente 0 para o sinal + e 1 para o sinal -).
- Qual o maior e o menor inteiro representável num computador com 8 bits para inteiros?

[1] *bit* (binary digit) – elemento de memória básico que assume dois estados *on* e *off* que se associam aos dígitos 0 e 1

Representação de REAIS na base b $(b \geq 2)$

Qualquer número real $x \neq 0$ pode ser representado na forma

$$x = (-1)^s (a_n a_{n-1} \dots a_1 a_0 . a_{-1} a_{-2} \dots)_b$$

$$= (-1)^s (a_n \times b^n + \cdots + a_1 \times b^1 + a_0 + a_{-1} \times b^{-1} + a_{-2} \times b^{-2} + \cdots)$$

onde $s \in \{0,1\},\ a_i \in \{0,1,\dots,b-1\},\ a_n \neq 0.$

➲ Mudança da base b para a base 10:

- $(0.12)_8 = 1 \times 8^{-1} + 2 \times 8^{-2} = 0.15625$

- $(0.1101)_2 = 2^{-1} + 2^{-2} + 2^{-4} = 0.8125$

➲ Mudança da base 10 para a base b:

Se $x = (.a_{-1}a_{-2}\dots a_{-k})_b$ então, $x \times b = (a_{-1}.a_{-2}\dots a_{-k})_b$,
ou seja a_{-1} é a parte inteira de $x \times b$ e $.a_{-2}\dots a_{-k}$ é a parte fracionária.
Multiplicando esta última outra vez por b e considerando a parte inteira obtém-se a_{-2} e assim sucessivamente.

- Representação de .625 na base 2:

$$
\begin{aligned}
.625 \times 2 &= 1.250 \\
.250 \times 2 &= 0.500 \\
.500 \times 2 &= 1.000
\end{aligned}
$$

$$.625 = (.101)_2$$

Toolbox AN

`fracDec2Bin - Determina representação binária de um número fracionário`

Representação de reais em vírgula flutuante

Dado $x \in \mathbb{R}, x \neq 0$, e fixada uma base $b \geq 2$, x admite uma representação na forma[2]

$$x = (-1)^s (\sum_{k=1}^{\infty} d_k \times b^{-k}) \times b^e$$

onde

- $s \in \{0, 1\}$ sinal;
- $e \in \mathbb{Z}$;
- $d_k \in \{0, 1, \ldots, b-1\}$, $d_1 \neq 0$.

expoente

$$x = (-1)^s (\underbrace{.d_1 d_2 d_3 \ldots}_{m_x})_b \, b^e$$

mantissa

[2] a representação nessa forma é **única** se $\forall k \; \exists p \geq k : d_p \neq (b-1)$.

Exemplos

- $$\begin{aligned} -3.725 &= -0.3725 \times 10^1 \\ &= -(3 \times 10^{-1} + 7 \times 10^{-2} + 2 \times 10^{-3} + 5 \times 10^{-4}) \times 10^1. \end{aligned}$$

- $$\begin{aligned} (101.01)_2 &= (0.10101)_2 \times 2^3 \\ &= (1 \times 2^{-1} + 1 \times 2^{-3} + 1 \times 2^{-5}) \times 2^3. \end{aligned}$$

- $$\begin{aligned} 1/3 &= 0.33333\cdots \\ &= (\sum_{k=1}^{\infty} 3 \times 10^{-k}) \times 10^0 \end{aligned}$$

2. Sistema de numeração de vírgula flutuante

Um sistema de numeração de vírgula flutuante $F(b, t, m, M)$ é caracterizado por quatro parâmetros:

- b - base;
- t - número de dígitos da mantissa;
- m - valor mínimo do exponente;
- M - valor máximo do expoente.

Num computador, o número de dígitos da mantissa é fixo (t) e o expoente é limitado por um valor mínimo (m) e um valor máximo (M).Assim, os números de máquina são os que podem ser escritos como

$$x = (-1)^s (\sum_{k=1}^{t} d_k \times b^{-k}) \times b^e, \quad m \leq e \leq M, d_1 \neq 0,$$

juntamente com o número zero. Estes são os chamados números *normalizados*.

Um sistema $F(b,t,m,M)$ pode ainda admitir os chamados números *desnormalizados* ou *subnormais*, que são os números (diferentes de zero) tais que $d_1 = 0$, quando o expoente assume o valor mínimo , i.e.

$$(-1)^s(.0d_2d_3\dots d_t)_b\ b^m$$

Exemplo: Números positivos normalizados de $F(2,2,-1,1)$:

$$\begin{array}{ll} (.10)_2 \times 2^{-1} & (.11)_2 \times 2^{-1} \\ (.10)_2 \times 2^{0} & (.11)_2 \times 2^{0} \\ (.10)_2 \times 2^{1} & (.11)_2 \times 2^{1} \end{array}$$

➠ Quais são os números desnormalizados deste sistema?

Overflow e underflow

➲ O maior número de $F(b,t,m,M)$ chama-se *nível de overflow*[3] e é dado por

$$\Omega := (1-b^{-t})b^{M}.$$

➲ O menor número positivo normalizado, chamado-se *nível de underflow*[4] e é dado por

$$\omega := b^{m-1}.$$

➲ O menor número positivo de um sistema que admita números desnormalizados é b^{m-t}.

➲ Ao conjunto $R_{\mathcal{F}} := [-\Omega, -\omega] \cup \{0\} \cup [\omega, \Omega]$ chamamos *conjunto dos números representáveis*.

[3] Assumimos, se nada for dito em contrário, que a tentativa de representar um número real x tal que $|x| > \Omega$ conduzirá a uma situação de overflow.

[4] Se nada for dito em contrário, quando nos referirmos a um sistema $F(b,t,m,M)$, consideramos apenas os números normalizados. Assim, dizemos que números x tais que $0 < |x| < \omega$ conduzem a underflow.

Exemplos

- Em $F(10, 4, -2, 3)$:

$$\Omega = (1 - 10^{-4})10^3 = 0.9999 \times 10^3$$

$$\omega = 10^{-3}$$

- Em $F(2, 4, -2, 3)$:

$$\Omega = (1 - 2^{-4})2^3 = (0.1111)_2 \times 2^3$$ 7.5

$$\omega = 2^{-3}$$

Arredondamento

Seja $\mathcal{F} = F(10, 4, -2, 3)$ e $x = .75824 \times 10^{-2}$. Note-se que $x \in R_{\mathcal{F}}$, mas $x \notin \mathcal{F}$, i.e. $F \subsetneqq R_{\mathcal{F}}$.

➠ Dado $x \in R_{\mathcal{F}}$, como representá-lo no computador?

Dado $x \in R_{\mathcal{F}}$, $fl(x)$ designa o número de $F(b, t, m, M)$ obtido (salvo indicação em contrário) somando $\frac{1}{2}b^{-t}$ à mantissa e truncando o resultado para t dígitos.

Assim, no sistema referido, tem-se que

$$fl(.75824 \times 10^{-2}) = .7582 \times 10^{-2},$$

enquanto que

$$fl(.75825 \times 10^{-2}) = .7583 \times 10^{-2}$$

Em $F(2,4,-2,3)$:

$$fl(.10110 \times 2^{-2}) = .1011 \times 2^{-2}; \qquad fl(.10111 \times 2^{-2}) = .1100 \times 2^{-2}$$

Toolbox AN

`fl` - Arredondamento de número num sistema de vírgula flutuante de base 10

Exemplo:

```
>> fl(.75824*10^-2,4,-2,3,1);
         0.007582
>> fl(.75824*10^4,4,-2,3,1);
ocorreu overflow
              Inf
```

➲ Chama-se *epsilon da máquina*, e denota-se por ε, a diferença entre o número de $F(b,t,m,M)$ imediatamente superior a 1 e o número 1, isto é,

$$\varepsilon := b^{1-t}.$$

➲ A *unidade de erro de arredondamento* do sistema é $\mu := \frac{1}{2}b^{1-t} = \frac{1}{2}\varepsilon.$

Exemplo: Em $F(10,4,-2,3)$,

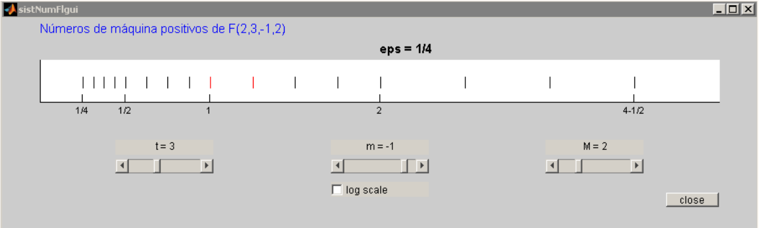
- $\mu = 0.5 \times 10^{-3}$
- $\varepsilon = 10^{-3}$

Quanto é $1+\mu$ e $1+\varepsilon$?

Simulação de um sistema de vírgula flutuante

adaptado de: Numerical Computing with MATLAB, Cleve Moler

http://www.mathworks.com/moler/chapters.html sistNumFlgui.m



Algoritmo para estimar μ

$\mu \leftarrow 1$

enquanto $(1+\mu > 1)$

$$\mu \leftarrow \frac{\mu}{2}$$

fim

Operações de vírgula flutuante

Representaremos as operações de vírgula flutuante pelo símbolo usual rodeado por $\bigcirc$; por exemplo $\oplus$, $\otimes$.

Admitimos que o resultado de uma operação de vírgula flutuante é obtido por arredondamento do resultado da operação exata , isto é,

$$x \oplus y = fl(x+y),\ x \otimes y = fl(x \times y),\ \text{etc.}$$

Exemplo: Sejam $\mathcal{F} := F(10, 4, -99, 99)$ e $x = 0.5289$, $y = 0.8012$ e $z = 0.6024$.

- $x \oplus (y \oplus z) = 0.1933 \times 10^1$
- $(x \oplus y) \oplus z = 0.1932 \times 10^1$!!!

3. A Norma IEEE 754

Com o objetivo de uniformizar as operações nos sistemas de vírgula flutuante foi publicada, em 1985, a norma IEEE 754.[5]

Esta norma especifica dois formatos básicos para representação de números em sistema de vírgula flutuante:

- o formato simples, com 32 bits;
- o formato duplo, com 64 bits.

[5] IEEE- Institute for Electrical and Electronics Engineers.

➲ Alocação dos bits no formato simples

0	1 2 3 4 5 6 7 8	9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
S	Expoente (8 bits)	Mantissa (23 bits)

➲ Alocação dos bits no formato duplo

0	1 2 3 4 5 6 7 8 9 10 11	12 13 14 15 16 17 18 19 20 21 22 23 24 25 · · · 51 52 53 54 55 56 57 58 59 60 61 62 63
S	Expoente (11 bits)	Mantissa (52 bits)

A norma IEEE 754 permite representar números normalizados na forma

$$x = (-1)^s (d_0.d_{-1}d_{-2}\cdots d_{-(t-1)})_2 2^{\mathfrak{e}},$$

- $s \in \{0, 1\}$ sinal;
- $d_0 = 1$ bit implícito;
- t número de bits da mantissa;
- $e_{\min} \leq \mathfrak{e} \leq e_{\max}$ expoente;

	formato simples	formato duplo
t	24	53
$e_{\min}$	-126	-1022
$e_{\max}$	127	1023

expoente − ε		$\mathfrak{e}$
0 0 0 0 0 0 0 0	$\hookrightarrow 0$	reservado
0 0 0 0 0 0 0 1	$\hookrightarrow 1$	-126
0 0 0 0 0 0 1 0	$\hookrightarrow 2$	-125
$\vdots$		$\vdots$
0 1 1 1 1 1 1 1	$\hookrightarrow 127$	0
1 0 0 0 0 0 0 0	$\hookrightarrow 128$	1
$\vdots$		$\vdots$
1 1 1 1 1 1 0 1	$\hookrightarrow 253$	126
1 1 1 1 1 1 1 0	$\hookrightarrow 254$	127
1 1 1 1 1 1 1 1	$\hookrightarrow 255$	reservado

Formato simples

O menor expoente representado:

$$\varepsilon = (00000001)_2 = (1)_{10},$$

correspondendo ao expoente mínimo

$$\mathfrak{e} = e_{\min} = -126.$$

O maior expoente representado:

$$\varepsilon = (11111110)_2 = (254)_{10},$$

correspondendo ao expoente máximo

$$\mathfrak{e} = e_{\max} = 127.$$

$$\boxed{\mathfrak{e} = \varepsilon - e_{\max}}$$

Expoente enviesado

O sistema de numeração IEEE admite ainda números desnormalizados

$$x = (-1)^s (0.d_{-1} d_{-2} \cdots d_{-(t-1)})_2 2^{e_{\min}},$$

e os "números" especiais

- $+0$ e -0

que correspondem a duas representações diferentes do mesmo número 0;

- $+\infty$ e $-\infty$ ($\mathbf{Inf}$ e $-\mathbf{Inf}$)

para representar, por exemplo, o resultado da divisão de um número por zero;

- $\mathbf{NaN}$ (Not a Number),

para representar o resultado de operações não definidas matematicamente, tais como $0/0$, $\infty - \infty$, etc.

$\pm$	$e_1e_2e_3\cdots e_7e_8$	$d_1d_2d_3\cdots d_{22}d_{23}$

$e_1e_2\cdots e_8$	valor representado
$(00000000)_2 = (0)_{10}$	$\pm(0.d_1d_2d_3\cdots d_{22}d_{23})_2 \times 2^{-126}$
$(00000001)_2 = (1)_{10}$	$\pm(1.d_1d_2d_3\cdots d_{22}d_{23})_2 \times 2^{-126}$
$(00000010)_2 = (2)_{10}$	$\pm(1.d_1d_2d_3\cdots d_{22}d_{23})_2 \times 2^{-125}$
$\vdots$	$\vdots$
$(01111111)_2 = (127)_{10}$	$\pm(1.d_1d_2d_3\cdots d_{22}d_{23})_2 \times 2^{0}$
$(10000000)_2 = (128)_{10}$	$\pm(1.d_1d_2d_3\cdots d_{22}d_{23})_2 \times 2^{1}$
$\vdots$	$\vdots$
$(11111101)_2 = (253)_{10}$	$\pm(1.d_1d_2d_3\cdots d_{22}d_{23})_2 \times 2^{126}$
$(11111110)_2 = (254)_{10}$	$\pm(1.d_1d_2d_3\cdots d_{22}d_{23})_2 \times 2^{127}$
$(11111111)_2 = (255)_{10}$	$\pm\infty$ se $d_1 = d_2 \cdots = d_{23} = 0$ NaN, nos outros casos

➲ O formato simples corresponde ao sistema $F(2, 24, -125, 128)$ e o formato duplo corresponde a $F(2, 53, -1021, 1024)$.

	formato simples	formato duplo
nível overflow Ω	$\approx 3.4028 \times 10^{38}$	$\approx 1.798 \times 10^{308}$
nível underflow ω	$\approx 1.1755 \times 10^{-38}$	$\approx 2.225 \times 10^{-308}$
menor positivo (desnormalizado)	$\approx 1.4013 \times 10^{-45}$	$\approx 4.941 \times 10^{-324}$
epsilon ε	$\approx 1.1921 \times 10^{-7}$	$\approx 2.220 \times 10^{-16}$

➲ A norma IEEE 754 especifica também as regras de arredondamento a utilizar. Por defeito, é utilizado o chamado arredondamento para par, isto é, dado $x \in \mathbb{R}$, $fl(x)$ é escolhido como o número de máquina (normalizado ou desnormalizado) mais próximo de x, sendo, em caso de "empate", escolhido aquele que tem o último *bit* da mantissa igual a zero.

Exceções:

- se $x \geq (2 - 2^{-t})2^{e_{\max}}$, $fl(x) = \mathbf{Inf}$;
- se $x \leq -(2 - 2^{-t})2^{e_{\max}}$, $fl(x) = -\mathbf{Inf}$;

Exemplos - precisão simples

1	1 0 0 0 0 0 0 1	1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

$-(1.111)_2 \times 2^{129-127} = -(7.5)_{10}$

0	0 0 0 0 0 0 0 0	1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

$(0.11)_2 \times 2^{-126}$

0	1 1 1 1 1 1 1 1	0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

NaN

1	1 1 1 1 1 1 1 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

$-\infty$

Funções do MATLAB

`realmax` `realmin` `epsilon` `Inf` `NaN`

`ceil` `fix` `floor` `round`

4. Erros

Erro absoluto e erro relativo

Seja $\tilde{x}$ um valor aproximado para a solução x de um dado problema.

- **Erro absoluto** do valor aproximado $\tilde{x}$ para x: $\mathcal{E}_{\tilde{x}} := x - \tilde{x}$

  $\hookrightarrow \quad \tilde{x} = x - \mathcal{E}_{\tilde{x}}$

- **Erro relativo** do valor aproximado $\tilde{x}$ para x $(x \neq 0)$: $\mathcal{R}_{\tilde{x}} := \frac{x-\tilde{x}}{x}$

  $\hookrightarrow \quad \tilde{x} = x(1 - \mathcal{R}_{\tilde{x}})$

  Estimativa: $|\mathcal{R}_{\tilde{x}}| \approx \frac{|x-\tilde{x}|}{|\tilde{x}|}$

Exemplo:

```
>> erro_abs=@(x,xtil) x-xtil; erro_rel=@(x,xtil) (x-xtil)./x;
>> E=erro_abs([1/3,1/3000],[.3333,.0003])
E =
   3.3333e-05   3.3333e-05

>> R=erro_rel([1/3,1/3000],[.3333,.0003])
R =
       0.0001          0.1
```

Erros de arredondamento

Sejam $\mathcal{F} := F(b,t,m,M)$ e $x = (-1)^s m_x b^{e} \in R_{\mathcal{F}}$ não nulo e normalizado (i.e. $b^{-1} \leq m_x < 1$, $m \leq e \leq M$).

➲ Erro absoluto de arredondamento: $|\mathcal{E}_{fl(x)}| \leq \mu\, b^{e-1}$

$$|\mathcal{E}_{fl(x)}| = |x - fl(x)| \leq \frac{1}{2} b^{-t} b^{e} = \mu b^{e-1}$$

➲ Erro relativo de arredondamento: $|\mathcal{R}_{fl(x)}| \leq \mu$

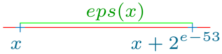
$$|\mathcal{R}_{fl(x)}| = \frac{|x - fl(x)|}{|x|} \leq \frac{\frac{1}{2} b^{-t}\, b^{e}}{b^{-1}\, b^{e}} = \frac{1}{2} b^{1-t} = \mu.$$

- O majorante do erro absoluto depende de e, logo de x. O majorante do erro relativo[6] depende apenas da unidade de erro de arredondamento da máquina usada.

[6] Muitas vezes estamos interessados apenas no valor absoluto dos erros (absoluto ou relativo), designado-os pelos mesmos nomes, caso tal seja claro pelo contexto.

- $fl(x) = x(1+\epsilon)$, com $|\epsilon| \leq \mu$
- $\mathcal{F} = F(2, 53, -1021, 1024)$, $x = (-1)^s (0.1d_2d_3\dots)_2\, 2^{e}$
  - $|\mathcal{E}_{fl(x)}| \leq 2^{e-54}$
  - $|\mathcal{R}_{fl(x)}| \leq 2^{-53} = \texttt{eps/2} \approx 1.1102 \times 10^{-16}$
  - Note que, se $x \in \mathcal{F}$, então o número de máquina que lhe sucede é

$$x + 2 \times 2^{e-54} = x + 2^{e-53}$$




```
>> x=0.5*2.^[0 1 53 54]
x =
      0.5           1        4.5036e+15    9.0072e+15

>> eps(x)
ans =
   1.1102e-16   2.2204e-16         1             2
```

Propagação dos erros nas operações aritméticas

Sejam $\tilde{x}$ e $\tilde{y}$ valores aproximados para x e y, respetivamente $(x, y \neq 0)$, e sejam $S = x + y$, $P = x.y$ e $Q = x/y$. Sejam $\widetilde{S}$, $\widetilde{P}$ e $\widetilde{Q}$ os valores aproximados para S, P e Q, obtidos usando os valores $\tilde{x}$ e $\tilde{y}$ em vez de x e y e admitindo que as operações são efetuadas exatamente.

- $\mathcal{E}_{\widetilde{S}} = \mathcal{E}_{\tilde{x}} + \mathcal{E}_{\tilde{y}}$

  $\mathcal{R}_{\widetilde{S}} = \frac{x}{x+y}\mathcal{R}_{\tilde{x}} + \frac{y}{x+y}\mathcal{R}_{\tilde{y}}$;

- $\mathcal{E}_{\widetilde{P}} = \tilde{y}\mathcal{E}_{\tilde{x}} + \tilde{x}\mathcal{E}_{\tilde{y}} + \mathcal{E}_{\tilde{x}}\mathcal{E}_{\tilde{y}}$ $\qquad |\mathcal{R}_{\tilde{x}}|, |\mathcal{R}_{\tilde{y}}| \ll 1$

  $\mathcal{R}_{\widetilde{P}} = \mathcal{R}_{\tilde{x}} + \mathcal{R}_{\tilde{y}} - \mathcal{R}_{\tilde{x}}\mathcal{R}_{\tilde{y}}$ $\qquad \mathcal{R}_{\widetilde{P}} \approx \mathcal{R}_{\tilde{x}} + \mathcal{R}_{\tilde{y}}$

- $\mathcal{E}_{\widetilde{Q}} = \dfrac{\tilde{y}\mathcal{E}_{\tilde{x}} - \tilde{x}\mathcal{E}_{\tilde{y}}}{\tilde{y}\left(\tilde{y} + \mathcal{E}_{\tilde{y}}\right)}$

  $\mathcal{R}_{\widetilde{Q}} = \dfrac{\mathcal{R}_{\tilde{x}} - \mathcal{R}_{\tilde{y}}}{1 - \mathcal{R}_{\tilde{y}}}$ $\qquad \mathcal{R}_{\widetilde{Q}} \approx \mathcal{R}_{\tilde{x}} - \mathcal{R}_{\tilde{y}}$.

Algarismos significativos/casas decimais de precisão

$\mathcal{F} := F(10, t, m, M)$ - sistema de vírgula flutuante;

$x = (-1)^s m_x 10^{\boldsymbol{e}} \in R_{\mathcal{F}}$;

$\tilde{x}$ - valor aproximado para x.

- Diz-se que $\tilde{x}$ é uma aproximação para x com precisão de $\boldsymbol{p}$ casas decimais (c.d.) ou que $\tilde{x}$ aproxima x com p casas decimais (corretas), se

$$|x - \tilde{x}| \leq 0.5 \times 10^{-\boldsymbol{p}}.$$

- Diz-se que $\tilde{x}$ é uma aproximação para x com precisão de $\boldsymbol{q}$ algarismos signficativos (a.s) ou que $\tilde{x}$ aproxima x com q algarismos significativos (corretos), se

$$|x - \tilde{x}| \leq 0.5 \times 10^{\boldsymbol{e}-\boldsymbol{q}}.$$

Exemplos:

- $x = 3.127,\ \tilde{x} = 3.123$ $\boxed{e = 1}$

  $|x - \tilde{x}| = 0.4 \times 10^{-2} < 0.5 \times 10^{-\mathbf{2}} = 0.5 \times 10^{\mathbf{1}-\mathbf{3}}$ $\boxed{\text{2 c.d.}}$ $\boxed{\text{3 a.s.}}$

- $x = 0.0003127,\ \tilde{x} = 0.0003123$ $\boxed{e = -3}$

  $|x - \tilde{x}| = 0.4 \times 10^{-6} < 0.5 \times 10^{-\mathbf{6}} = 0.5 \times 10^{-\mathbf{3}-\mathbf{3}}$ $\boxed{\text{6 c.d.}}$ $\boxed{\text{3 a.s.}}$

- $x = 3.127,\ \tilde{x} = 3.12$ $\boxed{e = 1}$

  $|x - \tilde{x}| = 0.7 \times 10^{-2} < 0.5 \times 10^{-\mathbf{1}} = 0.5 \times 10^{\mathbf{1}-\mathbf{2}}$ $\boxed{\text{1 c.d.}}$ $\boxed{\text{2 a.s.}}$

- $x = 3.127,\ \tilde{x} = 3.13$ $\boxed{e = 1}$

  $|x - \tilde{x}| = 0.3 \times 10^{-2} < 0.5 \times 10^{-\mathbf{2}} = 0.5 \times 10^{\mathbf{1}-\mathbf{3}}$ $\boxed{\text{2 c.d.}}$ $\boxed{\text{3 a.s.}}$

Se $\tilde{x}$ é uma aproximação para x com $\boldsymbol{p}$ casas decimais de precisão, então $\tilde{x}$ tem precisão de $\boldsymbol{q} = \boldsymbol{p} + \boldsymbol{e}$ algarismos significativos.

Algarismos significativos vs erro relativo

Seja $\tilde{x}$ uma aproximação para x.

- Se $|\mathcal{R}_{\tilde{x}}| \leq 0.5 \times 10^{-q}$, então $\tilde{x}$ tem precisão de q a.s.

$$\left|\frac{x-\tilde{x}}{x}\right| \leq 0.5 \times 10^{-q} \Rightarrow |x-\tilde{x}| \leq 0.5 \times 10^{-q}|m_x 10^e| < 0.5 \times 10^{-q} \times 10^e$$

- Se $\tilde{x}$ tem precisão de q a.s., então $|\mathcal{R}_{\tilde{x}}| \leq 0.5 \times 10^{1-q}$.

$$|\mathcal{R}_{\tilde{x}}| = \left|\frac{x-\tilde{x}}{x}\right| \leq \frac{0.5 \times 10^{-q} \times 10^e}{|x|} \leq \frac{0.5 \times 10^{-q} \times 10^e}{0.1 \times 10^e} = 0.5 \times 10^{1-q}$$

Como $|\mathcal{R}_{fl(x)}| \leq 2^{-53} \approx 1.1102 \times 10^{-16} < 0.5 \times 10^{-15}$ (ver pág. 30), $fl(x)$ tem (no mínimo) precisão de 15 algarismos significativos.

5. Condicionamento e estabilidade

Cancelamento subtrativo

- $x = 0.76545424 \times 10^1, \;\; y = 0.76544199 \times 10^1$
- $\tilde{x} = 0.76545421 \times 10^1, \;\; \tilde{y} = 0.76544200 \times 10^1$

$\tilde{x}$ aproxima x com 7 a.s.; $\tilde{y}$ aproxima y com 7 a.s.

- $z = x - y = 0.1225 \times 10^{-3}; \;\; \tilde{z} = \tilde{x} - \tilde{y} = 0.1221 \times 10^{-3}$

$\tilde{z}$ aproxima z com 3 algarismos significativos!

Erro relativo em $\tilde{z}$ pode ser $10\,000$ superior aos erros relativos em $\tilde{x}$ e $\tilde{y}$.

Cancelamento Subtrativo - Perda de algarismos significativos de precisão que resulta da subtração de números muito próximos. O erro relativo do resultado é muito maior que o erro relativo dos dados. ✘

Exemplo: $f(x) = \sqrt{x+1} - \sqrt{x}$ ou $g(x) = \frac{1}{\sqrt{x+1}+\sqrt{x}}$?

x	$f(x)$	$g(x)$
10^1	$1.543471301870229 \times 10^{-1}$	$1.543471301870205 \times 10^{-1}$
10^2	$4.987562112088946 \times 10^{-2}$	$4.987562112089027 \times 10^{-2}$
10^3	$1.580743742895763 \times 10^{-2}$	$1.580743742895582 \times 10^{-2}$
10^4	$4.999875006248544 \times 10^{-3}$	$4.999875006249609 \times 10^{-3}$
10^5	$1.581134877255863 \times 10^{-3}$	$1.581134877256878 \times 10^{-3}$
10^6	$4.999998750463419 \times 10^{-4}$	$4.999998750000625 \times 10^{-4}$
10^7	$1.581138790243131 \times 10^{-4}$	$1.581138790555721 \times 10^{-4}$
10^8	$5.000000055588316 \times 10^{-5}$	$4.999999987499999 \times 10^{-5}$
10^9	$1.581139076733962 \times 10^{-5}$	$1.581138829688905 \times 10^{-5}$
10^{10}	$4.999994416721165 \times 10^{-6}$	$4.999999999875000 \times 10^{-6}$

cancelamento subtrativo ✘ todos algarismos significativos ✔

Condicionamento de um problema

➲ Um problema diz-se mal condicionado se for muito sensível a pequenas alterações nos seus dados; caso contrário, diz-se bem condicionado.

Exemplo: Considere-se o problema da resolução do sistema de equações

$$\begin{cases} 1.01x + 0.99y = 2.00 \\ 0.99x + 1.01y = 2.00 \end{cases} \qquad \text{Sol} : x = 1; y = 1.$$

Modifiquemos ligeiramente o lado direito...

$$\begin{cases} 1.01x + 0.99y = 2.02 \\ 0.99x + 1.01y = 1.98 \end{cases} \qquad \text{Sol} : x = 2; y = 0.$$

$$\begin{cases} 1.01x + 0.99y = 1.98 \\ 0.99x + 1.01y = 2.02 \end{cases} \qquad \text{Sol} : x = 0; y = 2.$$

"Pequenas" alterações nos dados $\Longrightarrow$ "grandes alterações" nas soluções.

Problema mal condicionado!

Exemplo: Polinómio de Wilkinson $p(x) = (x-1)(x-2)\dots(x-20)$

$$p(x) = x^{20} - 210x^{19} + 20615x^{18} - 1256850x^{17} + \cdots + 20!$$

Seja q o polinómio que resulta de p, modificando o coeficiente de x^{19}:

$$q(x) = x^{20} - (210 + 2^{-23})x^{19} + 20615x^{18} - 1256850x^{17} + \cdots + 20!$$

As raízes destes polinómios obtidas usando o Mathematica (precisão *infinita*) e o Matlab estão na tabela da página seguinte.

- Os resultados obtidos pelo Matlab no cálculo das raízes do polinómio p, mostram como estas são extremamente sensíveis ao efeito dos erros de arredondamento.
- Uma perturbação relativa de $2^{-23}/210 \approx 5.7 \times 10^{-10}$ no coeficiente a_{19} de p provoca uma alteração muito grande nas raízes. Metade delas "tornam-se" complexas.

Problema mal condicionado!

Polinómio de Wilkinson (continuação)

1	1.0000	1.0000	$1.0000 + 0.0000i$
2	2.0000	2.0000	$2.0000 + 0.0000i$
3	3.0000	3.0000	$3.0000 + 0.0000i$
4	4.0000	4.0000	$4.0000 + 0.0000i$
5	5.0000	5.0000	$5.0000 + 0.0000i$
6	6.0000	6.0000	$6.0000 + 0.0000i$
7	7.0000	6.9997	$6.9994 + 0.0000i$
8	8.0003	8.0073	$8.0153 + 0.0000i$
9	8.9984	8.91725	$8.8533 + 0.0000i$
10	10.0061	$10.0953 - 0.6435i$	$9.9859 - 0.8088i$
11	10.9840	$10.0953 + 0.6435i$	$9.9859 + 0.8088i$
12	12.0334	$11.7936 - 1.6523i$	$11.6820 - 1.8654i$
13	12.9491	$11.7936 + 1.6523i$	$11.6820 + 1.8654i$
14	14.0653	$13.9924 - 2.5188i$	$13.9142 - 2.7966i$
15	14.9354	$13.9924 + 2.5188i$	$13.9142 + 2.7966i$
16	16.0483	$16.7307 - 2.8126i$	$16.7521 - 3.1418i$
17	16.9711	$16.7307 + 2.8126i$	$16.7521 + 3.1418i$
18	18.0112	$19.5024 - 1.9403i$	$19.6819 - 2.2103i$
19	18.9972	$19.5024 + 1.9403i$	$19.6819 + 2.2103i$
20	20.0003	20.8469	$21.0997 + 0.0000i$

raízes $p(x)$ $q(x)$

Número de condição de uma função

- Seja $\tilde{x}$ um valor aproximado para x com um erro relativo $|\mathcal{R}_{\tilde{x}}| \ll 1$.
- Seja f uma função continuamente diferenciável numa vizinhança de x (que contém $\tilde{x}$).

Como se "propaga" o erro em x ao cálculo de $f(x)$?

Sejam $y = f(x)$ e $\tilde{y} = f(\tilde{x})$. Pelo teorema do valor médio,

$$y - \tilde{y} = f(x) - f(\tilde{x}) = f'(\xi)(x - \tilde{x}), \quad \xi \in (\min\{x, \tilde{x}\}, \max\{x, \tilde{x}\}).$$

Temos, então

$$\left|R_{\tilde{y}}\right| = \left|\frac{y - \tilde{y}}{y}\right| = \left|\frac{f'(\xi)(x - \tilde{x})}{f(x)}\right| = \left|\frac{x f'(\xi)}{f(x)}\right| \cdot \left|\frac{x - \tilde{x}}{x}\right| = \left|\frac{x f'(\xi)}{f(x)}\right| \left|R_{\tilde{x}}\right|.$$

Como admitimos que x e $\tilde{x}$ estão próximos, será razoável substituir $f'(\xi)$ por $f'(x)$, tendo-se, então

$$|R_{f(\tilde{x})}| \approx \left|\frac{xf'(x)}{f(x)}\right| |R_{\tilde{x}}|$$

Assim, a quantidade $\left|\frac{xf'(x)}{f(x)}\right|$ é uma medida do condicionamento do cálculo do valor de f em x.

➲ Chama-se número de condição de f em x e denota-se por $\operatorname{cond} f(x)$ à quantidade dada por

$$\operatorname{cond} f(x) = \left|\frac{xf'(x)}{f(x)}\right|$$

- Se $\operatorname{cond} f(x)$ é "pequeno", o problema de calcular $f(x)$ é bem condicionado.
- Se $\operatorname{cond} f(x)$ é "grande", o problema de calcular $f(x)$ é mal condicionado.

Exemplo:

- $f(x) = \sqrt{x}$

  $f'(x) = \frac{1}{2\sqrt{x}}$

  $\operatorname{cond} f(x) = \frac{1}{2}$

  função bem condicionada para todo x.

- $f(x) = e^x$

  $f'(x) = e^x$

  $\operatorname{cond} f(x) = |x|$

  função mal condicionada para valores de x tais que $|x|$ é “grande”;

  bem condicionada para valores de x tais que $|x|$ é “pequeno”.

Estabilidade/instabilidade de um método

➲ Diz-se que um método é instável se os erros se amplificam no decurso dos cálculos, de forma inaceitável; caso contrário, o método diz-se estável.

Exemplo: Duas expressões para a mesma função:

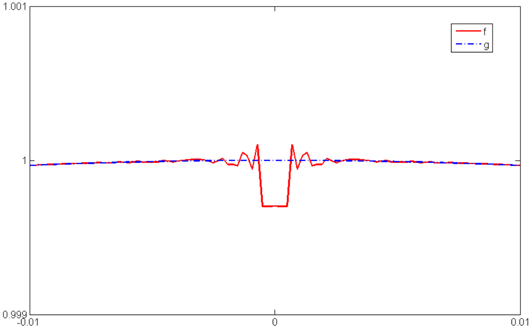
$$f(x) = \frac{1 - \cos^2 x}{x^2} \qquad g(x) = \frac{\operatorname{sen}^2 x}{x^2}$$

Os resultados de $\cos x$ e $\operatorname{sen} x$ foram arredondados para 10 a.s.; os restantes cálculos foram efetuados com a precisão do Matlab. Por exemplo,

```
>> f=@(x) (1-fl(cos(x),10).^2)./x.^2;
>> g=@(x) fl(sin(x),10).^2./x.^2;
>> [f(5*10^-5) g(5*10^-5)]
ans =
   0.959999990612914   0.999999999200000
```

$$f(5\times 10^{-5}) = g(5\times 10^{-5}) = 0.99999999916666666694\ldots$$

```
>> x=linspace(-0.01,0.01);
>> plot(x,f(x),'r-',x, g(x),'b-.')
```



As funções f e g, embora matematicamente equivalentes, são numericamente diferentes!