Armazenamento digital de números

Um nº N pode ser representado num sistema numérico de base β como:

$$N = a_n\beta^n + a_{n-1}\beta^{n-1} + \ldots + a_1\beta + a_0 + a_{-1}\,\beta^{-1} + \ldots + a_{-n}\beta^{-n}$$

onde a_i são inteiros positivos e menores que β.

Exemplo1: Conversão de binário para decimal

$$N = (110100010)_2$$

$$\begin{aligned} & 2^8 + 2^7 + 0 + 2^5 + 0 + 0 + 0 + 2^1 + 0 \\ = \; & 256 + 128 + 0 + 32 + 0 + 2 \\ = \; & (418)_{10} \end{aligned}$$

Exemplo2: Conversão de decimal para binário

N par: a_i=0
N impar: a_i=1

$$\begin{aligned}
N_0 &= 418 \\
&= a_n 2^n + a_{n-1} 2^{n-1} + \cdots + a_1 * 2 + a_0 \\
&\Rightarrow a_0 = 0 \\
N_1 &= \tfrac{N_0 - a_0}{2} = 209 \\
&= a_n 2^{n-1} + \cdots + a_2 * 2 + a_1 \\
&\Rightarrow a_1 = 1 \\
N_2 &= \tfrac{N_1 - a_1}{2} = 104 \\
&= a_n 2^{n-2} + \cdots + a_3 * 2 + a_2 \\
&\Rightarrow a_2 = 0, \text{etc.}
\end{aligned}$$

N=$(418)_{10}$ = $(110100010)_2$

Seja N=$(0.1)_{10}$ = $(?)_2$

$$z_1 = (0.1)_{10} = a_{-1} 2^{-1} + a_{-2} 2^{-2} + \cdots$$

$$2z_1 = a_{-1} + a_{-2} 2^{-1} + a_{-3} 2^{-2} + \cdots$$

$$\begin{aligned}
&z_1 = (0.1)_{10}; && 2z_1 = 0.2 < 1 \Rightarrow a_{-1} = 0 \\
&z_2 = 2z_1 - a_{-1} = 0.2; && 2z_2 = 0.4 < 1 \Rightarrow a_{-2} = 0 \\
&z_3 = 2z_2 - a_{-2} = 0.4; && 2z_3 = 0.8 < 1 \Rightarrow a_{-3} = 0 \\
&z_4 = 2z_3 - a_{-3} = 0.8; && 2z_4 = 1.6 > 1 \Rightarrow a_{-4} = 1 \\
&z_5 = 2z_4 - a_{-4} = 0.6; && 2z_5 = 1.2 > 1 \Rightarrow a_{-5} = 1 \\
&z_6 = 2z_5 - a_{-5} = 0.2; && 2z_6 = 0.4 < 1 \Rightarrow a_{-6} = 0 \\
&\vdots
\end{aligned}$$

N=$(0.1)_{10}$ = $(0.0\ 0011\ 0011\ 0011\ldots)_2$

Representação de números reais.

Representação de Ponto fixo.

Um número é representado com um nº fixo de digitos após o ponto decimal.

$$(101.011)_2=1*2^2+0+1*2^0+0*2^{-1}+2^{-2}+2^{-3}$$

- Gama de representação de números limitada.
- Precisão da representação limitada.Quanto maior menor a gama de representação
- Operações aritméticas usadas são as mesmas da aritmética inteira, logo muito eficientes.

Representação em virgula flutuante.

Um número é representado com um nº fixo de digitos significativos e redimensionado usando um expoente de uma base fixa.

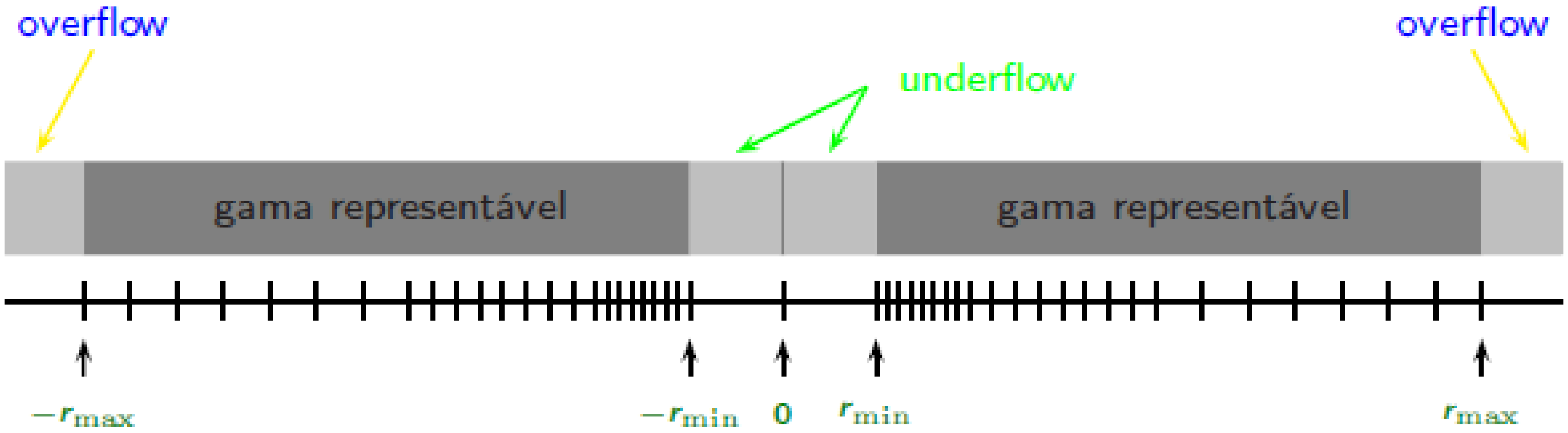
Números representáveis: $x = \pm(0.d_1 d_2 \cdots d_n) \times \beta^e$

β base de representação

n número de dígitos da mantissa (precisão)

m, M expoentes mínimo e máximo (gama representável)

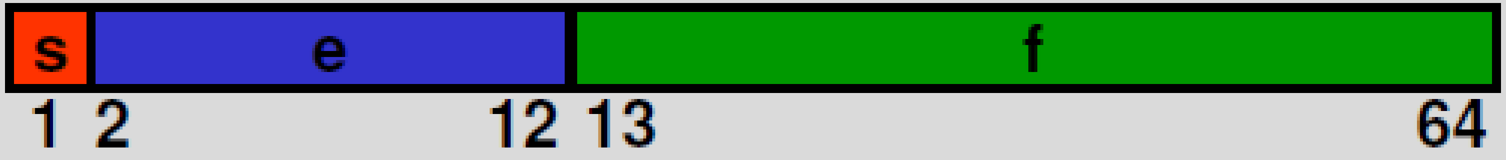
Sistema normalizado: $x = 0 \ \vee \ d_1 \neq 0$



Representação em virgula flutuante. Precisão simples e dupla.

Em precisão simples são usados 32 bits para a representação: sinal (1), expoente(8), mantissa(23).
Em precisão dupla 64 bits: sinal (1), expoente(11), mantissa(52)

IEEE 754 Standard para números de precisão dupla



$$x = \pm(1 + f) \cdot 2^e$$

$$f = \frac{d_1}{2} + \frac{d_2}{2^2} + \cdots + \frac{d_{52}}{2^{52}} \qquad d_k = 0 \text{ or } 1$$

$$-1022 \leq e \leq 1023$$

Round-off: eps = 2^{-52}

Underflow: realmin = 2^{-1022}

Overflow: realmax = (2-eps) $\cdot 2^{1023}$

Representação em virgula flutuante. Precisão simples e dupla.

Representation Scheme for IEEE Doubles

Number Name	*Values of s, e, and f*	*Value of Double*
Normal	$0 < e < 2047$	$(-1)^s \times 2^{e-1023} \times 1.f$
Subnormal	$e = 0,\ f \neq 0$	$(-1)^s \times 2^{-1022} \times 0.f$
Signed zero	$e = 0,\ f = 0$	$(-1)^s \times 0.0$
$+\infty$	$s = 0,\ e = 2047,\ f = 0$	+INF
$-\infty$	$s = 1,\ e = 2047,\ f = 0$	−INF
Not a number	$s = u,\ e = 2047,\ f \neq 0$	NaN

Representation Scheme for Normal and Abnormal IEEE Singles

Number Name	*Values of s, e, and f*	*Value of Single*
Normal	$0 < e < 255$	$(-1)^s \times 2^{e-127} \times 1.f$
Subnormal	$e = 0,\ f \neq 0$	$(-1)^s \times 2^{-126} \times 0.f$
Signed zero (±0)	$e = 0,\ f = 0$	$(-1)^s \times 0.0$
$+\infty$	$s = 0,\ e = 255,\ f = 0$	+INF
$-\infty$	$s = 1,\ e = 255,\ f = 0$	−INF
Not a number	$s = u,\ e = 255,\ f \neq 0$	NaN

Descrição do erro na representação finita de números reais.

absolute & relative

a : exact, $\bar{a}$: approximate

absolute error of $\bar{a}$: $\bar{a} - a \quad (=: \Delta a)$

relative error of $\bar{a}$: $\Delta a / a \quad$ (provided $a \neq 0$)

bounding interval

$a \in [\bar{a} - \varepsilon, \bar{a} + \varepsilon]$ is written as $a = \bar{a} \pm \varepsilon$

correct decimals

$\bar{a}$ has t correct decimals if $|\Delta a| \leq 0.5 \cdot 10^{-t}$

significant digits

$\bar{a} = d_0.d_1 d_2 \cdots \times 10^E \quad (d_0 \neq 0)$

has s significant digits if $|\Delta a| \leq 0.5 \cdot 10^{1+E-s}$

for example:

$a = \sqrt{200} = 14.14213562\cdots$

$\bar{a} = 14.14 = 1.414 \cdot 10^1$

$\Delta a = -0.00213562\cdots$

$\frac{\Delta a}{a} = -0.0001510\cdots$

$|\Delta a| \leq 0.005$

$a = 14.14 \pm 0.005$

$\bar{a}$ has 2 correct decimals

$\bar{a}$ has 4 significant digits

$a = \pi = 3.14159265\cdots$

$\bar{a} = \frac{355}{113}$

$\Delta a = 2.667 \cdot 10^{-7}$

$\frac{\Delta a}{a} = 8.49 \cdot 10^{-8}$

$|\Delta a| \leq 0.5 \cdot 10^{\boxed{-6}}$

$a = \frac{355}{113} \pm 0.5 \cdot 10^{\boxed{-6}}$

$\bar{a}$ has $\boxed{6}$ correct decimals

$\bar{a}$ has $\boxed{7}$ significant digits

Aritmética em representações finitas de números reais.

Na adição/subtração acumulam erros absolutos, na multiplicação/divisão são os erros relativos.

Addition

$$|\Delta(x_1 + x_2)| \leq |\Delta x_1| + |\Delta x_2|$$

Subtraction

$$|\Delta(x_1 - x_2)| \leq |\Delta x_1| + |\Delta x_2|$$

Multiplication

$$\left|\frac{\Delta(x_1 x_2)}{x_1 x_2}\right| \leq \left|\frac{\Delta x_1}{x_1}\right| + \left|\frac{\Delta x_2}{x_2}\right| + \left|\frac{\Delta x_1}{x_1}\right| \cdot \left|\frac{\Delta x_2}{x_2}\right|$$

$$\approx \left|\frac{\Delta x_1}{x_1}\right| + \left|\frac{\Delta x_2}{x_2}\right|$$

Division

$$\left|\frac{\Delta(x_1/x_2)}{x_1/x_2}\right| \leq \left|\frac{\Delta x_1}{x_1}\right| + \left|\frac{\Delta x_2}{x_2}\right|$$

proof

$$\begin{aligned}|\Delta(x_1 + x_2)| &= |(\bar{x}_1 + \bar{x}_2) - (x_1 + x_2)| \\ &= |(\bar{x}_1 - x_1) + (\bar{x}_2 - x_2)| \\ &\leq |\bar{x}_1 - x_1| + |\bar{x}_2 - x_2|\end{aligned}$$

Example

$$x_1 = 123.4 \pm 0.05 = 123.4 \cdot (1 \pm 0.0004052)$$

$$x_2 = 122.1 \pm 0.05 = 122.1 \cdot (1 \pm 0.0004095)$$

$$x_1 - x_2 = 1.3 \pm 0.1$$

$$\frac{x_1}{x_2} = 1.010647 \cdot (1 \pm 0.0008147)$$

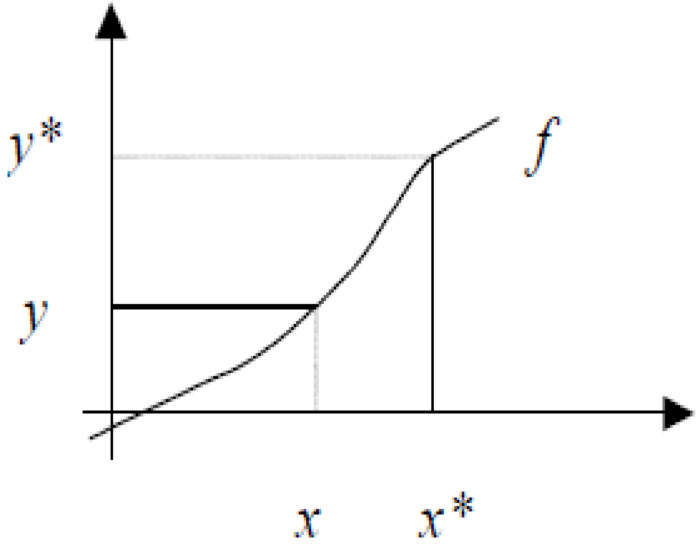
$$= 1.010647 \pm 0.00082$$

Propagação de erros no cálculo de $y = f(x)$

x^* valor aproximado de x. Como aproximar $y = f(x)$?

Será $\boxed{y^* = f(x^*)}$ uma boa aproximação?

f contínua: $\boxed{x^* \text{ próximo de } x} \Rightarrow \boxed{y^* \text{ próximo de } y}$



Majorante para o erro absoluto da aproximação y^* de y

$$\boxed{\varepsilon_y = |f'|\big|_{\max} \cdot \varepsilon_x}$$

Majorante para o erro relativo de $y^* = f(x^*)$

$$\boxed{\varepsilon'_y = \left| f'(x) \cdot \frac{x}{f(x)} \right|_{\max} \cdot \varepsilon'_x}$$

$\left|\frac{xf'(x)}{f(x)}\right|$ designa-se **número de condição** de f em x.

$\rightarrow$ $\left|\frac{xf'(x)}{f(x)}\right|$ reduzido: a função diz-se **bem condicionada**

$\rightarrow$ $\left|\frac{xf'(x)}{f(x)}\right|$ elevado: a função diz-se **mal condicionada**

Catastrophic Cancellation Errors (1)

The errors in

$$c = a + b \qquad \text{and} \qquad c = a - b$$

will be large when $a \gg b$ or $a \ll b$.

Consider $c = a + b$ with

$$a = x.xxx \ldots \times 10^0$$
$$b = y.yyy \ldots \times 10^{-8}$$

where x and y are decimal digits.

$$\begin{array}{cl} & \overbrace{\text{x.xxx xxxx xxxx xxxx}}^{\text{available precision}} \\ + & \text{0.000 0000 yyyy yyyy yyyy yyyy} \\ \hline = & \text{x.xxx xxxx zzzz zzzz } \underbrace{\text{yyyy yyyy}}_{\text{lost digits}} \end{array}$$

The most significant digits of a are retained, but the least significant digits of b are lost because of the mismatch in magnitude of a and b.

For subtraction: The error in

$$c = a - b$$

will be large when $a \approx b$.

Consider $c = a - b$ with

$$a = x.xxxxxxxxxxxx1ssssss$$
$$b = x.xxxxxxxxxxxx0tttttt$$

$$
\begin{array}{rl}
 & \overbrace{\text{x.xxx xxxx xxxx 1}}^{\text{available precision}} \\
- & \text{x.xxx xxxx xxxx 0} \\
\hline
= & \text{0.000 0000 0000 1} \;\; \underbrace{\text{uuuu uuuu uuuu}}_{\text{unassigned digits}} \\
= & \text{1.uuuu uuuu uuuu} \times 10^{-12}
\end{array}
$$

The result has only one significant digit. Values for the uuuu digits are not necessarily zero. The *absolute* error in the result is small compared to either a or b. The *relative* error in the result is large because $ssssss - tttttt \neq uuuuuu$ (except by chance).

Implications for Routine Calculations

- Floating point comparisons should test for "close enough" instead of exact equality.
- Express "close" in terms of

  absolute difference, $|x - y|$

  or

  relative difference, $\dfrac{|x - y|}{|x|}$

Floating Point Comparison

Don't ask "is x equal to y".

```
if x==y       %  Don't do this
   ...
end
```

Instead ask, "are x and y 'close enough' in value"

```
if abs(x-y) < tol
   ...
end
```

Regras para não perder precisão

- Trabalhar sempre com números da ordem de 1 (unidades “adaptadas”)
- Não somar números de ordens de grandeza muito diferentes
- Não subtrair números próximos e “grandes”
- Não dividir por números pequenos

Truncation Error

Consider the series for $\sin(x)$

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \cdots$$

For small x, only a few terms are needed to get a good approximation to $\sin(x)$. The $\cdots$ terms are "truncated"

$$f_{\text{true}} = f_{\text{sum}} + \text{truncation error}$$

The size of the truncation error depends on x *and* the number of terms included in f_{sum}.

Roundoff and Truncation Errors (3)

To study the roles of roundoff and truncation errors, compute the finite difference[2] approximation to $f'(x)$ when $f(x) = e^x$.

$$f'_{fd}(x) = \frac{f(x+h) - f(x)}{h} \quad \text{or} \quad f'_{fd}(x) = f'(x) + \mathcal{O}(h)$$

Evaluate E_{rel} at $x = 1$ for a range of h.

Truncation error dominates at large h.

Roundoff error in $f(x+h) - f(h)$ dominates as $h \to 0$.

$$E_{\mathrm{rel}} = \frac{f'_{fd}(x) - f'(x)}{f'(x)} = \frac{f'_{fd}(x) - e^x}{e^x}$$