# Representação e erros numéricos

Marina Andretta

ICMC-USP

29 de fevereiro de 2012

Baseado no livro Análise Numérica, de R. L. Burden e J. D. Faires.

# Resolução computacional de problemas

Temos interesse em resolver problemas reais, envolvendo cálculos, usando um computador. Isso porque o problema pode ser muito complexo ou muito grande para ser resolvido "na mão".

Para isso, transformamos o problema em uma formulação matemática. Este processo é conhecido como modelagem.

Depois da modelagem do problema, usamos o computador para resolver o problema matemático.

# Resolução computacional de problemas

Neste processo, muitos erros podem ser introduzidos: o modelo pode não representar exatamente o problema, a medidas de dados podem conter erros, a resolução pelo computador pode apresentar erros numéricos, etc.

Estes tipos de erros devem ser controlados para que a resposta obtida tenha alguma serventia.

Nos concentraremos mais nos erros que podem ser produzidos durante a resolução do problema matemático.

# Representação numérica

A aritmética usada por nós é diferente da aritmética usada pelas calculadoras e computadores. Estamos acostumados a verdades como $3 + 5 = 5 + 3 = 8$, $\sqrt{(7)^2} = 7$ e $\pi/\pi = 1$.

Estamos supondo, aqui, que os números que usamos tem precisão infinita. Que todos os números podem ser representados.

No entanto, quando usamos um computador para representar um número, usamos um número finito de casas decimais. Isso já limita a representação a números racionais. E, mesmo assim, nem todo número racional pode ser representado.

# Representação numérica

O que acontece na prática é que substituímos um número não representável por um número próximo dele. Isso pode ser satisfatório em algumas situações.

Mas é preciso tomar cuidado e se lembrar sempre que estamos lidando com uma aritmética diferente quando fazemos contas no computador, já que estes erros estarão sempre presentes e devem ser controlados.

O erro produzido pelo computador para realizar cálculos com números reais é chamado de erro de arredondamento.

# Representação em ponto flutuante

Em 1985, o IEEE (Instituto de Engenheiros Elétricos e Eletrônicos) publicou um relatório chamado *Binary Floating Point Arithmetic Standard 754-1985*.

Neste relatório foram especificados formatos para precisão simples, dupla e extendida, que geralmente são seguidos pelos fabricantes de computadores.

# Representação em ponto flutuante

Por exemplo, em um sistema de 64 bits para representar um real longo, os 64 bits são distribuídos da seguinte maneira:

- O primeiro bit, denotado por $s$, é um indicador de sinal (0 para positivo e 1 para negativo).
- Em seguida, há 11 bits para um expoente, chamados de característica (denotada por $c$). Este é um número inteiro.
- Os 52 bits restantes representam a mantissa (denotada por $f$), que é uma fração binária.

A base do sistema é sempre 2.

# Representação em ponto flutuante

Como 52 algarismos binários correspondem a 15 ou 16 algarismos decimais, podemos dizer que este sistema tem pelo menos 15 algarismos decimais de precisão.

O exponte, com 11 algarismos binários, fornece uma faixa de 0 a $2^{11} - 1 = 2047$. Mas, para permitir expoentes negativos e uma melhor representação de números de módulo pequeno, é subtraído 1023 do expoente. Desta forma, na realidade, a faixa de valores para o expoente vai de -1023 a 1024.

# Representação em ponto flutuante

Para economizar armazenamento e obter uma representação única dos números em ponto flutuante, é imposta uma normalização.

A utilização deste sistema fornece um número em ponto flutuante da forma

$$(-1)^s 2^{c-1023}(1+f)$$

# Representação em ponto flutuante - exemplo

Considere o seguinte número, representado no sistema mencionado:

0 10000000011 1011100100010000000000000000000000000000000000000000

O primeiro bit representa o sinal $s$. Neste caso, 0 ou positivo.

# Representação em ponto flutuante - exemplo

Os 11 bits seguintes, 10000000011, fornecem a característica. Este número, no sistema decimal, é

$$c = 1 \times 2^{10} + 1 \times 2^{1} + 1 \times 2^{0} = 1027.$$

Portanto, a parte exponencial do número é dada por

$$2^{1027-1023} = 2^{4}.$$

# Representação em ponto flutuante - exemplo

Os últimos 52 bits representam a mantissa

$$f = 1 \times 2^{-1} + 1 \times 2^{-3} + 1 \times 2^{-4} + 1 \times 2^{-5} + 1 \times 2^{-8} + 1 \times 2^{-12}.$$

Assim, o número representado é

$$(-1)^s 2^{c-1023}(1+f) =$$

$$(-1)^0 2^{1027-1023}\left(1 + \frac{1}{2} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{256} + \frac{1}{4096}\right) = 27,56640625.$$

# Representação em ponto flutuante - exemplo

Note que o número de máquina anterior a este é dado por

0 10000000011 10111001000011111111111111111111111111111111111111

E o número de máquina posterior é dado por

0 10000000011 10111001000100000000000000000000000000000000000001

# Representação em ponto flutuante - exemplo

Isso significa que o número de máquina original representa não somente o número 27,56640625, mas também metade dos números reais que estão entre ele e seus números de máquina mais próximos.

Ou seja, ele representa qualquer número real no intervalo

[27,566406249999998223643160599749535322189331054687 5,
27,566406250000001776356839400250464677810668945312 5)

# Representação em ponto flutuante

O menor número em módulo que pode ser representado neste sistema é dado por $s = 0$, $c = 1$ e $f = 0$.

Ou seja,

$$(-1)^0 2^{1-1023}(1+0) \approx 0,2225 \times 10^{-307}.$$

Números que ocorrem em cálculos com módulos menores do que este valor resultam em underflow, e são, geralmente, arredondados para 0.

# Representação em ponto flutuante

O maior número em módulo que pode ser representado neste sistema é dado por $s = 0$, $c = 2046$ e $f = 1 - 2^{-52}$.

Ou seja,

$$(-1)^0 2^{2046-1023}(1 + 1 - 2^{-52}) \approx 0,17977 \times 10^{309}.$$

Números que ocorrem em cálculos com módulos maiores do que este valor resultam em overflow. Isso geralmente acarreta em parada do cálculo, a menos que o programa tenha sido projetado para detectar este tipo de erro.

# Representação em ponto flutuante

Note que há duas representações possíveis para o zero:

- uma positiva, com $s = 0$, $c = 0$ e $f = 0$;
- e uma negativa, com $s = 1$, $c = 0$ e $f = 0$.

# Representação em ponto flutuante

Para facilitar os cálculo daqui em diante, usaremos a forma normalizada de ponto flutuante decimal

$$\pm 0, d_1 d_2 ... d_k \times 10^n,$$

com $1 \leq d_1 \leq 9$ e $0 \leq d_i \leq 9$, para $i = 2, ..., k$.

Os números desta forma são chamados de números de máquina decimais de $k$ algarismos.

# Representação em ponto flutuante

Qualquer número real positivo dentro do intervalo numérico da máquina pode ser normalizado na forma

$$y = 0, d_1 d_2 ... d_k d_{k+1} d_{k+2} \times 10^n.$$

A forma em ponto flutuante $y$, denotada por $fl(y)$, é obtida terminando a mantissa de $y$ em $k$ algarismos decimais.

Há duas maneiras disto ser realizado: truncamento e arredondamento.

# Truncamento

O truncamento consiste em, simplesmente, descartar os dois últimos algarismos $d_{k+1}d_{k+2}$ de $y$.

Isso produz a forma em ponto flutuante de $y$

$$fl(y) = 0, d_1 d_2 ... d_k \times 10^n.$$

# Arredondamento

O arredondamento consiste em somar $5 \times 10^{n-(k+1)}$ a $y$ e, então, truncar o resultado.

Isso produz a forma em ponto flutuante de $y$

$$fl(y) = 0, \delta_1 \delta_2 ... \delta_k \times 10^n.$$

# Arredondamento

Deste modo, se $d_{k+1} \geq 5$, adicionamos 1 a $d_k$ para obter $fl(y)$.

Isto é o que chamamos arredondamento para cima.

Mas, se $d_{k+1} < 5$, simplesmente truncamos o número.

Isto é o que chamamos arredondamento para baixo.

Note que, quando arredondamos para baixo, $d_i = \delta_i$ para todo $1 \leq i \leq k$, mas isso não acontece quando arredondamos para cima.

# Erros absoluto e relativo

Existem duas maneiras muito usadas para medir erros de aproximação. São elas: erro absoluto e erro relativo.

Se $\bar{p}$ é uma aproximação de $p$, o erro absoluto é dado por $|\bar{p} - p|$.

O erro relativo é dado por $\frac{|\bar{p}-p|}{|p|}$, contanto que $p \neq 0$.

Como medida de precisão, o erro absoluto pode ser enganoso, já que não leva em consideração o tamanho do número que está sendo usado. Neste caso, o erro relativo pode ser mais significativo.

# Algarismos significativos

Diz-se que o número $\bar{p}$ aproxima $p$ até $t$ algarismos significativos se $t$ for o maior inteiro não-negativo para o qual

$$\frac{|\bar{p}-p|}{|p|} \leq 5 \times 10^{-t}.$$

É possível mostrar que, usando a representação em ponto flutuante $fl(y)$ para um número $y$, com $k$ algarismos decimais, o número de dígitos significativos é:

- $k-1$, quando o truncamento é usado e
- $k$, quando o arredondamento é usado.

# Aritmética de ponto flutuante

Já vimos que, somente a representação de números reais no computador já introduz erros de arredondamento.

Além disso, as contas feitas por ele introduz mais alguns erros.

A aritmética feita pelo computador pode ser aproximada com as 4 operações básicas definidas da seguinte maneira:

- $x \oplus y = fl(fl(x) + fl(y))$
- $x \ominus y = fl(fl(x) - fl(y))$
- $x \otimes y = fl(fl(x) \times fl(y))$
- $x \oslash y = fl(fl(x)/fl(y))$

# Aritmética de ponto flutuante

Com esta aritmética, muitos erros podem ser introduzidos.

Os mais comuns aparecem quando os números envolvidos tem ordem de grandeza muito diferentes.

Ou o cancelamento de dígitos significativos quando são subtraídos números muito parecidos.

E algumas coisas estranhas acontecem, como, por exemplo, uma conta $(a \oplus b) \oplus c \neq a \oplus (b \oplus c)$.

# Aritmética de ponto flutuante - exemplo

A fórmula quadrática afirma que as raízes de $ax^2 + bx + c = 0$, quando $a \neq 0$, são

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{e} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.$$

Usando a aritmética de arredondamento com quatro algarismos, considere esta fórmula aplicada à equação $x^2 + 62,1x + 1 = 0$, cujas raízes são, aproximadamente,

$$x_1 = -0,01610723 \quad \text{e} \quad x_2 = -62,0839.$$

# Aritmética de ponto flutuante - exemplo

Nesta equação, $b^2$ é muito maior do que $4ac$. Então, o numerador o cálculo de $x_1$ envolve a subtração de dois números quase iguais.

Como

$$\sqrt{b^2 - 4ac} = \sqrt{(62,1)^2 - 4 \times 1 \times 1} = \sqrt{3856 - 4} = \sqrt{3852} = 62,06,$$

temos

$$fl(x_1) = \frac{-62,10 + 62,06}{2} = \frac{-0,04}{2} = -0,02.$$

# Aritmética de ponto flutuante - exemplo

Esta é uma aproximação insatisfatória para $x_1 = -0,01611$, com grande erro relativo

$$\frac{|-0,02+0,01611|}{|-0,01611|} \approx 2,4 \times 10^{-1}.$$

# Aritmética de ponto flutuante - exemplo

No entanto, se racionalizarmos o numerador da fórmula para cálculo de $x_1$, obtemos

$$x_1 = \frac{-2c}{b + \sqrt{b^2 - 4ac}}.$$

Usando esta reformulação, temos

$$fl(x_1) = \frac{-2}{62,10 + 62,06} = \frac{-2}{124,2} = -0,01610,$$

com pequeno erro relativo $6,2 \times 10^{-4}$ pequeno.