Sérgio Galdino

CÁLCULO NUMÉRICO

Caderno 01

1994

1

COMPUTAÇÕES NUMÉRICAS

1.0 - Representação

O sistema de numeração decimal é o sistema mais usado pelo homem nos dias de hoje. O número 10 tem papel fundamental, é chamado de base do sistema. Os símbolos 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, são usados para representar qualquer grandeza. O fato do sistema decimal ser largamente utilizado tem evidentemente razões históricas, pois na realidade qualquer número inteiro maior que 1 poderia ter sido escolhido. De fato, no mundo dos computadores digitais o sistema binário é o utilizado. O número 2 é a base do sistema e os símbolos 0 e 1 servem para representar uma grandeza qualquer. Ao lado do sistema binário, os sistemas octal e hexadecimal, base 8 e 16 respectivamente, são também utilizados. Isto ocorre pelo fato de que cada símbolo octal e hexadecimal representa um equivalente a três e quatro símbolos no sistema binário e vice-versa.

1.1 - Conversão

Dado um número x representado na base N, isto é, na N-representação, e nós queremos saber como representa-lo na base M, isto é, na M-representação. Temos então a equação: $x = a_m N^m + a_{m-1} N^{m-1} + \ldots = b_n M^n + b_{n-1} M^{n-1} + \ldots$ onde os coeficientes $a_m, a_{m-1}, \ldots$ são conhecidos e os coeficientes $b_n, b_{n-1}, \ldots$ devem ser determinados. Observe que $b_n, b_{n-1}, \ldots$ devem ser expressos com símbolos de dígitos da N-representação. Para realizar a conversão dividiremos x em uma parte inteira i e uma parte fracionário f. Nós temos $i = b_n M^n + b_{n-1} M^{n-1} + \ldots + b_1 M^1 + b_0$, e dividindo i por M nós obtemos um quociente q_1 e um resto $r_1 = b_0$. Continuando, dividiremos q_1 por M, nós conseguiremos q_2 e o resto $r_2 = b_1$, e, obviamente, $b_0, b_1, b_2, \ldots$ são os restos consecutivos quando i é dividido repetitivamente por M. De forma semelhante nós encontramos a parte fracionária como as partes inteiras consecutivas quando f é multiplicado repetitivamente por M e a parte inteira é removida. Os cálculos devem ser feitos na N-representação e M deve ser também dado nesta representação.

Exemplo: Converta o número decimal 261,359 para a representação binária, ternária e octal.

Conversão: Decimal para binário

Inteira: Divisão sucessiva do número decimal por 2

Divisão	Resto
261:2	1
130:2	0
65:2	1
32:2	0
16:2	0
8:2	0
4:2	0
2:2	0
1:2	1

O número inteiro binário é obtido através dos restos das divisões escritos na ordem inversa da sua obtenção. Então $(261)_{10} = (1.0000.0101)_2$

Fração: Multiplicação sucessiva da fração decimal por 2

Multiplicação	Sobra
0,359x2	0
0,718x2	1
0,436x2	0
0,872x2	1
0,774x2	1
0,488x2	0
0,976x2	1
0,952x2	1
0,904x2	1

A fração binária é obtida através das sobras, parte inteira, das multiplicações escritas na ordem direta de sua obtenção. Então $(0,359)_{10} = (0,0101.1011.1...)_2$

Somando-se a parte inteira e fracionárias dos binários obtidos têm-se

$$(261,359)_{10} = (1.0000.0101,0101.1011.1...)_2$$

Conversão: Decimal para ternário

Inteira: Divisão sucessiva do número decimal por 3

Divisão	Resto
261:3	0
87:3	0
29:3	2
9:3	0
3:3	0
1:3	1

O número inteiro ternário é obtido através dos restos das divisões escritos na ordem inversa da sua obtenção. Então $(261)_{10} = (100.200)_3$

Fração: Multiplicação sucessiva da fração decimal por 3

Multiplicação	Sobra
0,359x3	1
0,077x3	0
0,231x3	0
0,693x3	2
0,079x3	0
0,273x3	0
0,711x3	2
0,133x3	0
0,399x3	1

A fração ternária é obtida através das sobras, parte inteira, das multiplicações escritas na ordem direta de sua obtenção. Então

$$(0,359)_{10} = (0,100\,200\,201\ldots)_3$$

Somando-se a parte inteira e fracionárias dos ternários obtidos têm-se

$$(261,359)_{10} = (100.200,100\,200\,201\ldots)_3$$

Conversão: Decimal para hexadecimal

Inteira: Divisão sucessiva do número decimal por 16

Divisão	Resto
261:16	5
16:16	0
1:16	1

O número inteiro hexadecimal é obtido através dos restos das divisões escritos na ordem inversa da sua obtenção. Então $(261)_{10} = (105)_{16}$

Fração: Multiplicação sucessiva da fração decimal por 16

Multiplicação	Sobra
0,359x16	5
0,744x16	11
0,904x16	14
0,464x16	7
0,424x16	6
0,784x16	12
0,544x16	8
0,704x16	11
0,264x16	4

A fração hexadecimal é obtida através das sobras, parte inteira, das multiplicações escritas na ordem direta de sua obtenção. Então

$$(0,359)_{10} = (0,5\ \underline{11}\ \underline{14}\ 7\ 6\ \underline{12}\ 8\ \underline{11}\ 4...)_3$$

ou, utilizando-se os símbolos hexadecimais

$$(0,359)_{10} = (0,5BE.76C.8B4...)_{16}$$

Somando-se a parte inteira e fracionárias dos hexadecimais obtidos têm-se

$$(261,359)_{10} = (105,5BE.76C.8B4...)_{16}$$

Tabela 1.1 - Símbolos Hexadecimais

Grandeza	Símbolo hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

1.2 - Representação da vírgula-flutuante

Em geral, um número N vírgula-flutuante tem a forma seguinte

$$N = M \cdot \beta^K$$

onde

M = mantissa, um valor que deve ser entre +1 e -1;
β = base, 2 se o sistema de numeração for binário, 10 se o sistema de numeração for o decimal, etc.;
K = expoente, um inteiro.

Exemplo: $N = -19{,}2 \cdot 10^{-8}$

Rescrevendo o número para forma $N = -0{,}192 \cdot 10^{-6}$, então o expoente é igual a -6, o mantissa é igual a -0,192 e a base é 10, têm-se um número escrito na notação de vírgula flutuante.

Se além da limitação $-1 < M < 1$, M também satisfaz uma das condições seguintes

$\frac{1}{\beta} \leq |M| < 1$ (significa que devemos ter um dígito não nulo após a vírgula)

ou

$M = 0$,

nós podemos dizer que $N = M \cdot \beta^K$ é um número escrito na notação de vírgula-flutuante normalizada.

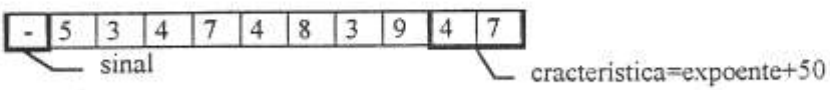
Exemplo:

$$\begin{array}{r} 0{,}27143247 \cdot 10^7 \\ -0{,}27072236 \cdot 10^7 \\ \hline 0{,}00071011 \cdot 10^7 \end{array}$$

Vemos que a diferença entre estes dois números vírgula-flutuantes normalizados, resulta num número em vírgula-flutuante não normalizado. Podemos, entretanto, normalizá-lo se deslocarmos a vírgula três lugares à direita e somar -3 ao expoente, obtendo-se $0{,}71011000 \cdot 10^4$ normalizado.

1.3 - Armazenamento na memória

Para começar vamos representar o número 0,00053474839 num computador decimal. A notação vírgula-flutuante normalizada deste número é $0{,}53474839 \cdot 10^{-3}$. Para evitar o expoente negativo, nós adicionamos, arbitrariamente, 50 ao expoente e o número agora é $0{,}53474839 \cdot 10^{47}$. O exponente é chamado de característica. O número pode ser representado, unicamente através da normalização da notação vírgula-flutuante, na memória do computador utilizando o esquema seguinte



Deve ser observado que a característica esta limitada a expressão seguinte: $-50 \leq k \leq 49$. O tem número máximo oito dígitos de precisão e a representação falha quando temos números muito grandes ou muito pequenos. De modo análogo, um número binário em vírgula-flutuante também pode ser armazenado na memória de um computador digital. Uma palavra armazenada tendo um bit de "sinal" e 35 bits regulares pode representar um número binário vírgula-flutuante na forma seguinte

0	1	8	9	35
sinal	característica	mantissa normalizada		

onde

sinal = sinal do número codificado, 0 se positivo e 1 se negativo;
característica = 128+expoente(resultado escrito em binário);
mantissa = fração binária normalizada.

Exemplo: Represente o número 12,625 numa palavra de 36 bits conforme descrito acima.

Divisão	Resto
12:2	0
6:2	0
3:2	1
1:2	1

Multiplicação	sobra
0,625x2	1
0,25x2	0
0,5x2	1

Divisão	Resto
132:2	0
66:2	0
33:2	1
16:2	0
8:2	0
4:2	0
2:2	0
1:2	1

$(12,625)_{10} = (1100,101)_2 = (0,1100101) \times 2^4$

Ajustando a característica $4+128 = 132$

$132 = 10000100_2$

0	1	2	3	4	5	6	7	8	9									35
0	1	0	0	0	0	1	0	0	1	1	0	0	0	1	1	0	0	0

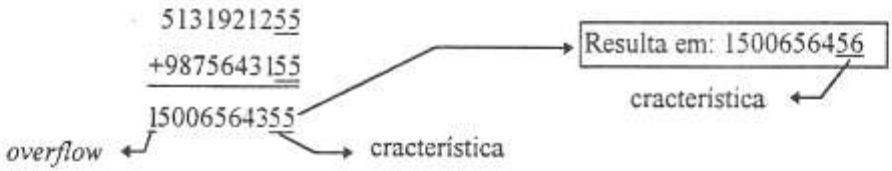
1.4 - Aritmética da vírgula-flutuante

Os princípios das operações aritméticas básicas de um computador serão discutidos agora. Para isto iremos considerar que estamos trabalhando num computador decimal com uma palavra de 10 dígitos de comprimento. Princípios semelhantes são utilizados em computadores binários (digitais). Na adição ou subtração de dois números o computador examina a característica ajustada dos números. Os seguintes casos são possíveis:

1-Características iguais: Adiciona-se as mantissas e mantém-se a característica

$$\begin{array}{r} 3210987654 \\ +1234012354 \\ \hline 4444999954 \end{array}$$

2- Quando existe *overflow* na adição das mantissas: o resultado será deslocado uma vez para direita



3- Carcterísticas distintas: mantém-se a de maior módulo e ajusta-se a de menor valor

$$\begin{array}{r} 3141112255 \\ +1234432153 \end{array} \longrightarrow \begin{array}{r} 3141112255 \\ +0012344355 \\ \hline 3153456555 \end{array}$$

4- Resultado com zero, ou zeros, a esquerda: Normaliza-se o resultado

$$\begin{array}{r} 3412222273 \\ -3400012273 \end{array}$$

0012210073 → resulta em: 121000071

Na multiplicação e divisão as mantissas e caracteristicas são tratadas separadamente.

Exemplo:

313131425(

×1231578265

mantissa = 0,31313142×0,12315782 = 0,038564583

exponencial = 51+65-50=66 , onde -50 é o desconto para compensar o ajuste +50 em cada ajuste do expoente da representação. A resposta é

0,31313142×0,12315782 = 0,03856458366

com a normalização teremos o resultado 3856458365

1.5 Erros

Os erros são definidos como absoluto e relativo. Se x é o número exato e x' uma aproximação, então temos

Erro absoluto: $\varepsilon = |x - x'|$, e

Erro relativo: $\left|\frac{\varepsilon}{x}\right| = \left|1 - \frac{x'}{x}\right|$

Um número decimal é arredondado na posição n desprezando-se todos os dígitos à direita desta posição. O dígito na posição n é deixado inalterado ou acrescido de uma unidade se o dígito da posição n+1 é um número menor que 5 ou maior que 5. Se o número na posição n+1 for igual a 5, o dígito na posição n é acrescido de uma unidade se ele for par e é deixado inalterado se for impar. Frequentemente é feito o trucamento par n decimais onde todos os dígitos além da posição n são simplesmente desprezados.

Exemplo: 3,1415926535

Arredondando: (2d) é 3,14;
(3d) é 3,141;
(4d) é 3,1416;
(7d) é 3,1415927;

onde (nd) = número de casas decimais.

Nós podemos dizer de forma somplória que digitos significativos são aqueles que tem informação sobre a dimensão do número sem contar com a porção exponencial. Naturalmente um digito d localizado mais a esquerda tem mais informação do que um mais a direita. Quando um número é escrito com somente seus dígitos significativos estes formam uma cadeia que começam com o primeiro dígito diferente de zero. Portanto se a parte fracionária termina com um ou vários zeros, eles são significativos por definição. Se o número é inteiro e termina com um ou vários zeros, eles são significtivos por definição. Se o número é inteiro e termina com zeros somente com o conhecimento da situação é que podemos decidir se eles são significativos ou não. Por exemplo, 8630574 escrito com 4 dígitos significativos é 8630000.

Em muitos casos nós estimamos o erro de uma função $f(x_1, x_2, \ldots, x_n)$ com erros individuais nas variáveis $\Delta x_1, \Delta x_2, \ldots, \Delta x_n$ conhecidos. Nós encontramos diretamente que

$$\Delta f = \frac{\partial f}{\partial x_1}\Delta x_1 + \frac{\partial f}{\partial x_2}\Delta x_2 + \ldots + \frac{\partial f}{\partial x_n}\Delta x_n$$

onde os termos de ordem superior foram desprezados. O erro máximo é dado por

$$|\Delta f| \le \left|\frac{\partial f}{\partial x_1}\right| \cdot |\Delta x_1| + \left|\frac{\partial f}{\partial x_2}\right| \cdot |\Delta x_2| + \ldots + \left|\frac{\partial f}{\partial x_n}\right| \cdot |\Delta x_n|$$

O limite superior do erro é geralmente bastante pessimista, em computações práticas, os erros tem uma tendência a cancelar. Por exemplo, se 20.000 números são arredondados com quatro casas decimais e adicionados, o erro máximo é

$$\frac{1}{2} \times 10^{-4} \times 20.000 = 1.$$

Do ponto de vista estatístico é esperado que em 99% de todos os casos o erro total não ultrapasse 0,005.

Normalmente nós classificamos os erros em computações numéricas para estudar suas fontes e os seus crescimentos individuais. Enquanto as fontes tem uma natureza que é essencialmente estática o crescimento é puramente dinâmico. Apesar de erros "grosseiros" terem frequentemente um papel destacado em cálculo numérico nós não trataremos dele aqui. Sendo assim, restam essencialmente três fontes de erro:

1. Erros iniciais;
2. Erros de trucamento;
3. Erros de arredondamento.

Os erros inicias são erros nos dados iniciais. Os erros de truncamento surjem quando um processo infinito (em algum sentido) é trocado por um finito. Erros de

arredondamento surgem do fato que durante uma computação numérica os números devem ser arredondados até um certo número de digitos. Geralmente uma computação numérica é feita em muitos passos. Cada passo nós temos as aproximações x' e y' dos números exatos x e y e nós obtemos a aproximação z' de z com o uso de uma das quatro operações aritiméticas. Por exemplo, se x'=x+δ e y'=y+γ e se calcula-se $z=\frac{x}{y}$, calcula-se na realidade $z'=\left(\frac{x'}{y'}\right)_{\text{arredondado}}=\frac{x+\delta}{y+\gamma}+\varepsilon$, então $z\cong' z+\left(\frac{1}{y}\right)\delta-\left(\frac{x}{y^2}\right)\eta+\varepsilon$ .O erro em z' é constituido dos erros propagados de x e y e um novo erro de arredondamento.

É interessante ilustrar diferentes tipos de erro mais explicitamente. Suponha que nós queremos computar f(x) para um dado x real. No cálculo prático x é aproximado por x' pois o computador tem uma palavra finita. A diferença x'-x é o erro inicial, enquanto $\varepsilon_1 = f(x') - f(x)$ é o erro propagado correspondente. Normalmente f é trocado por uma função mais simples f_1 (frequentemente uma série de potência truncada). A diferença $\varepsilon_2 = f_1(x') - f(x')$ é então o erro de truncamento. Os cálculos são feitos por um computador, portanto não são exatos. Na realidade calculamos $f_2(x')$ no lugar de $f_1(x')$, o qual é um valor calculado errado de uma função errada com argumento errado. A diferença $\varepsilon_3 = f_2(x') - f_1(x)$ erro de arredondamento propagado. O erro total é $\varepsilon = f_2(x') - f(x') = \varepsilon_1 + \varepsilon_2 + \varepsilon_3$.

Exemplo: Calcular $e^{1/3}$ fazendo todos os cálculos com 4 decimais.

$$\varepsilon_1 = e^{0.3333} - e^{1/3} = -0{,}000465196$$

$$e^x \cong f_1(x) = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}$$

para x' = 0,3333

$$\varepsilon_2 = f_1(x') - f(x') = -\left(\frac{0{,}3333^5}{5!} + \frac{0{,}3333^6}{6!} + \ldots\right) = -0{,}0000362750$$

$$f_2(x') = 1 + 0{,}3333 + 0{,}0555 + 0{,}0062 + 0{,}0005 = 1{,}3955$$

$f_1(x') = 1{,}3955296304$ obtidos com 10 decimais

$\varepsilon_3 = -0{,}0000296304$

$$\varepsilon = 1.3955 - e^{1/3} = \varepsilon_1 + \varepsilon_2 + \varepsilon_3 = -0{,}0001124250$$

1.6- Cancelamento numérico

Devido ao comprimento limitado das palavras em computadores, e em consequência do uso da aritmética da vírgula flutuante normalizada, as leis da aritmética elementar não são satisfeitas. Os efeitos do uso da aritimética da vírgula flutuante serão vistas em alguns exemplos que seguem.

Os exemplos a seguir violam a lei associativa da adição

Exemplo 1: (quando usamos uma máquina com quatro dígitos decimais na representação)

$$x = 9{,}909 \qquad y = 1{,}000 \qquad z = -0{,}990$$
$$(x + y) + z = 10{,}90 + (-0{,}990) = 9{,}910$$
$$x + (y + z) = 9{,}909 + (0{,}010) = 9{,}919$$

Exemplo 2: (quando usamos uma máquina com cinco dígitos decimais na representação)

$$x = 4561 \qquad y = 0{,}3472$$
$$(y + x) - x = (0{,}3472 + 4561) - 4561$$
$$= 4561 - 4561 = 0{,}0000$$
$$y + (x - x) = 0{,}3472 + (4561 - 4561)$$
$$= 0{,}3472 + 0{,}0000 = 0{,}3472$$

Vejamos agora um exemplo (quando usamos uma máquina com quatro dígitos decimais na representação) que viola a lei distributiva

$$x = 9909 \qquad y = -1{,}000 \qquad z = 0{,}999$$
$$(x \times y) + (x \times z) = -9909 + (9899) = -10{,}00$$
$$x \times (y + z) = 9909 + (-0{,}001) = -9{,}909$$

A equação do segundo grau $x^2 - bx + \varepsilon = 0$ tem duas soluções

$$x_1 = \frac{b + \sqrt{b^2 - 4\varepsilon}}{2} \qquad e \qquad x_2 = \frac{b - \sqrt{b^2 - 4\varepsilon}}{2}$$

Se $b < 0$ e $\varepsilon << b$, x_2 é expresso como a diferença de dois números praticamente iguais e poderá perder muitos digitos significativos. Se nós rescrevermos

$$x_2 = \frac{\varepsilon}{x_1} = \frac{2\varepsilon}{b + \sqrt{b^2 - 4\varepsilon}}$$

a raiz é aproximadamente $\frac{2\varepsilon}{b}$ sem perda de digitos significativos.

Exemplo: (quando usamos uma máquina com quatro dígitos decimais na representação)

$b = 300{,}0 \quad e \quad c = 1{,}000$

$\sqrt{90000 - 4{,}000} = 300{,}0$

$$x_1 = \frac{600{,}0}{2}$$

$$x_2 = \frac{0{,}000}{2{,}000} = 0{,}000$$

ou usando a relação $x_2 = \frac{\varepsilon}{x_1}$

$$x_2 = \frac{1{,}000}{300{,}0} = 0{,}003 \text{ é o resutado mais preciso.}$$

Sabe-se para x grande $\text{senh}(x) \equiv \cosh(x) \equiv \frac{e^{-x}}{2}$. Se queremos calcular e^{-x} podemos dizer que $e^{-x} = \cosh(x) - \text{senh}(x)$, o que conduz a um cancelamento perigoso. Por outro lado $e^{-x} = \frac{1}{\cosh(x) + \text{senh}(x)}$ fornece resultados bastante precisos.

1.7 - Exercícios:

1.1. Converta os números binários seguintes para a forma decimal.

a) 101110_2 b) 1100.01_2 c) 10101.1_2 d) 101.011_2

1.2. Converta os números decimais seguintes para a forma binária.

a) 95 b) 178 c) 250 d) 2000 e) 655 f) 722 $3,6\times10^{-1}$ g) 231 $2,5\times10^{-18}$

1.3. Rescreva os números seguintes na forma geral da vírgula-flutuante tal que a mantissa fique entre 1 e -1.

a) 27,534	b)-89,901	c)18×10^{21}	d)1.3657×10^{-7}
e) $11,0111_2$	f)$-111,0101_2$	g)$0,000101_2$	h)111010101_2

1.4. Qual o valor é o valor de cada expoente se os números vírgula-flutuante forem normalizados? Qual é o valor do expoente ajustado em cada caso se nós adicionamos arbitrariamente 25 ao expoente original?

1.5. Seja o número seguinte em ponto flutuante num computador de 32 bits:

00100101000000010001100111001110

Se o primeiro bit é o sinal do número, os oito seguintes a característica obtida com adição de 128 ao expoente do número vírgula flutuante, e o 23 restantes são a mantissa, responda às questões seguintes:

a) O número está normalizado? Se não, normalize-o.
b) Qual o sinal do número?
c) O valor absoluto do número é menor do que 1.

1.6. Repita a questão 1.5 com o número

10000000011011011010110110110110

1.7. Para a representação da questão 1.5, quais são aproximadamente o maior e o menor número, o menor número positivo e o próximo menor número positivo.

1.8. Use a aritmética de vírgula-flutuante para somar e subtraiar os seguintes pares de números?

a) 5,414234 e 2,27531 b)5,41234 e 22,7531

c) 54,67 e 0,328 d) $5,4\times10^{-8}$ e $3,14\times10^{-5}$

1.9) Use a aritmética da vírgula-flutuante para realizar as operações aritiméticas seguintes.

a) 3,14×7,47 b)75,81×8,15 c) 1,35:28,5 d) 4000:150

1.10. Calcular as cotas dos erros absolutos e relativos que se cometem ao se tomar como valores de π:

a) 22/7 b) 333/106 c) 355/113 d) $\sqrt{3}+\sqrt{2}$

1.11. Ao calcular a função $e^{-x}\cdot \text{sen}(x)$ para $x=2/3$, qual o erro máximo que se comete se o valor de x usado no cálculo possui um erro de 0,01?

1.12. Ao se calcular $\cos(x)\cong 1-\frac{x^2}{2!}-\frac{x^4}{4!}-\frac{x^6}{6!}$ para $x=5/7$, quais são os erros: inicial, propagado, de truncamento, de arredondamento e total cometidos quando se realiza os cálculos arredondados em duas decimais.

1.13. Repita os cálculos da questão 1.5 usando a aritmética da vírgula flutuante normalizada na base decimal, com mantissa de 4 dígitos.

Bibliografia

Kuo, Shan S., *Numerical Methods and Computers*. Addison-Wesley Publishing Company. pp. 25-9, 1965.

Frberg, Carl-Erik, *Introduction to Numerical Analysis*. Addison-Wesley Publishing Company. pp. 1-9, 1970.