Aritmética de precisión finita

Introducción al Análisis Numérico
MA-1006

UCR

Temas de la clase

a) Punto flotante.
b) Error absoluto, error relativo.
c) Operaciones aritméticas en precisión finita.
d) Propagación del error.

Ejemplo

Considere el número $\xi = -1.7245 \times 10^3$

Ejemplo

Considere el número $\xi = 1.125 \times 10^{-2}$ *¿Cómo se expresa este número cotidianamente?*

Definición (Representación punto flotante)

Sea $\beta \geq 2$ una base (usualmente par), t la precisión (número de dígitos en base β). Un número **punto flotante** *se representa en esta base como*

$$\xi = \pm(d_1.d_2...d_t)_\beta \times \beta^e = \pm m \times \beta^e$$

en donde $d_k \in \{0, 1, ..., \beta - 1\}$ son los **dígitos significativos,** *m* **mantisa** *y tiene t dígitos; $e \in \mathbb{Z}$ es el* **exponente**. *Si $d_1 \neq 0$ el número se dice* **normalizado**. *Un número punto flotante normalizado tal que $d_1 = 0$ implica entonces que $d_2 = d_3 = ... = d_t = 0$*

Note que dicho número es igual a

$$\pm(d_1 + d_2\beta^{-1} + d_3\beta^{-2} + ... + d_t\beta^{-(t-1)}) \times \beta^e, \quad 0 \leq d_i < \beta$$

Si el exponente de dos números punto flotante es el mismo, se dice que tienen la misma *magnitud*.
Los exponentes máximo y mínimo se denotan como e_{max} y e_{min}, respectivamente, y se tiene que (usualmente) $e_{\text{min}} < 0 < e_{\text{max}}$.
Entonces hay $e_{\text{max}} - e_{\text{min}} + 1$ posibles exponentes. El $+1$ es para el exponente cero.

Definición

Sean $\beta, t \in \mathbb{N}$, $0 \leq d_i < \beta$ *para* $i > 1$, $d_1 \neq 0$ *y* $L = e_{min}$, $U = e_{max}$. *Se define el conjunto normalizado de números reales de precisión finita*[a] *como*

$$\mathbb{F}(\beta, t, L, U) := \{0\} \cup \left\{ \xi \in \mathbb{R}^* : \xi = \pm \beta^e \times \sum_{i=1}^{t} d_i \beta^{1-i} \right\}$$

Cuando no haya ambigüedad dicho conjunto se denotará como $\mathbb{F}_{\beta,t}$.

[a] O *machine numbers set*.

Teorema

La cantidad de elementos en $\mathbb{F}(\beta, t, L, U)$ *es exactamente*

$$\operatorname{card}(\mathbb{F}(\beta, t, L, U)) = 2(\beta - 1)\beta^{t-1}(U - L + 1) + 1$$

Además, $x_{min} = \beta^{L}$ *y* $x_{max} = (1 - \beta^{-t})\beta^{U+1}$*, siendo estos elementos el valor mínimo y máximo positivos de* $\mathbb{F}_{\beta,t}$*, respectivamente.*

Comentarios:

Hay dos razones por las cuales algunos números no son representados de forma exacta en punto flotante:

1. La primera se ilustra con el número en base diez 0.1, el cual tiene una representación decimal finita, pero en binario su representación es $(0.0\overline{0011})_2$, luego dicho número en binario se encuentra estrictamente entre dos números punto flotante y a la vez no es ninguno de ellos.
2. La segunda razón es que el número $x \in \mathbb{R}$ podría ser muy grande/pequeño en valor absoluto y se escape del rango. A esto se le conoce como *overflow* (sobredesbordamiento) o *underflow* (subdesbordamiento) respectivamente.
   Por defecto, las computadoras atrapan esos casos limítrofes asignando los valores $\pm\infty$ y 0 sin previo aviso.

IEEE

La IEEE 754 reconoce los siguientes símbolos:

- $\pm\infty$, i.e. como el valor de $\pm 1/0$;
- `NaN` *not a number*, i.e. como el resultado de $0/0$ o $\infty - \infty$.

Esencialmente cada computadora desde 1985 implementa el estandard IEEE 754, el cual ofrece dos formatos binarios para la *hardware arithmethic* (por defecto `MATLAB` usa doble precisión, pero también permite el uso de precisión simple).

Precisión simple

Precisión simple: utiliza 32 bits (4 bytes) distribuidos así: 1 bit para el signo, 8 bits para el exponente y 23 bits para la mantisa. Tiene una precisión de 24 dígitos binarios. El exponente toma valores en $[-126, 127]$ con sesgo 127.

$\pm$	$a_1 a_2 a_3 \ldots a_8$	$b_1 b_2 b_3 \ldots b_{23}$

Ejemplo

Determine la representación del número $(1)_{10}$ *en el sistema de punto flotante con precisión simple*

Solución:

$$\color{blue}{0}\ \ 01111111\ \ \color{brown}{000\ \ 0000\ \ 0000\ \ 0000\ \ 0000\ \ 0000}$$

Ejemplo

Determine la representación del número $(3)_{10}$ *en el sistema de punto flotante con precisión simple*

Solución:

$$0 \quad 10000000 \quad 100 \quad 0000 \quad 0000 \quad 0000 \quad 0000 \quad 0000$$

Ejemplo

Convertir $(32995)_{10}$ *a su representación de precisión simple.*

Haciendo las respectivas operaciones se obtiene que

$$m = (32995)_{10} = (1000000011100011)_2$$

Normalizando, $m = (1.000000011100011)_2 \times 2^{15}$, entonces, $e = 15$.
El exponente que se guarda (sesgado) en precisión simple es:

$$e + 127 = 142 = (10001110)_2$$

Entonces, el número que se almacena en precisión simple como

$$0 \quad 10001110 \quad 000 \quad 0000 \quad 1110 \quad 0011 \quad 0000 \quad 0000$$

Ejercicio

Determine como convertir números en formato de precisión simple a números en base 10. *Muestre que*

0	01111100	01000000000000000000000

$= 0.15625$

Solución: precisión simple utiliza 32 bits (4 bytes) distribuidos así: 1 bit para el signo, 8 bits para el exponente y 23 bits para la mantisa. Tiene una precisión de 24 dígitos binarios. El exponente toma valores en $[-126, 127]$ con sesgo 127.

s	$a_1a_2a_3\ldots a_8$	$b_1b_2b_3\ldots b_{23}$

Sea N la representación en base 10 del número anterior, entonces se tiene que

$$N = (-1)^s \times (1.b_1b_2b_3...b_{23})_2 \times 2^{E-127}$$

donde $s \in \{0, 1\}$ es el signo, $E = (a_1a_2a_3...a_8)_2$ es el exponente sesgado. En este caso,

$$E = (01111100)_2 = 2^6 + 2^5 + 2^4 + 2^3 + 2^2 = 124$$

Usando lo anterior se tiene que el número representado es

$$N = (-1)^0 \times (1.01)_2 \times 2^{124-127} = (1 + 2^{-2}) \times 2^{-3} = 0.15625$$

Precisión doble

Utiliza 64 bits (8 bytes) distribuidos así:
1 bit para el signo,
11 bits para el exponente,
52 bits para la mantisa.
Tiene una precisión de 53 dígitos binarios. El exponente toma valores en $[-1022, 1023]$ con sesgo 1023.

$\pm$	$a_1a_2a_3\ldots a_{11}$	$b_1b_2b_3\ldots b_{52}$

Supongamos que se está utilizando el sistema de precisión doble de la IEEE. Sea $\mathbb{F}$ como el conjunto de números en punto flotante del sistema de precisión doble.

Definición (Función fl)

Considere la función $\mathrm{fl} : \mathbb{R} \to \mathbb{F}$, *donde a cada número real* ξ *se le asigna el correspondiente número en punto flotante* $\mathrm{fl}(\xi)$, *utilizando redondeo al valor más cercano. Es decir, si* $\xi_- \leq \xi < \xi_+$, *donde* $\xi_-, \xi_+ \in \mathbb{F}$ *son dos números en punto flotante consecutivos, se define*

$$\mathrm{fl}(\xi) = \begin{cases} \xi_-, & \textit{si } |\xi - \xi_-| \leq |\xi - \xi_-| \\ \xi_+, & \textit{en caso contrario} \end{cases}$$

Definición

El número ε_m más pequeño tal que

$$\mathrm{fl}(1+\varepsilon_m) > 1$$

se denomina **épsilon máquina** *o* **macheps**.

Nota

El épsilon máquina es la diferencia entre 1 y el número siguiente $x > 1$ con $x \in \mathbb{F}$ que se puede almacenar de forma exacta.

- Épsilon máquina para precisión simple es $\varepsilon_m = 2^{-23} \approx 1.19 \times 10^{-7}$.
- Para precisión doble es $\varepsilon_m = 2^{-52} \approx 2.22 \times 10^{-16}$.

Teorema

Sea $x \in \mathbb{R}$ *un número que puede ser representado de manera normal en un sistema de punto flotante con precisión* t. *Entonces, existe* $\delta \in \mathbb{R}$ *tal que*

$$\mathrm{fl}(x) = x(1 + \delta)$$

donde $|\delta| < \varepsilon_m/2$

Definición

Sea $x \in \mathbb{R}$ *y sea* $x^* = \mathrm{fl}(x)$. *Se define el error absoluto* ε_{ab} *como*

$$\varepsilon_{ab} = |x - x^*|$$

y el error relativo δ_{rel} *como*

$$\delta_{rel} = \frac{|x - x^*|}{|x|}, \quad x \neq 0$$

En la aritmética punto flotante, el error relativo es muy apropiado porque cada número es representado con una precisión relativa similar. Cuando $x = 0$ o x es muy cercano a cero, es mejor considerar el error absoluto.

Dígitos significativos

Para una base β se tiene que el error relativo, al aproximar x por algún valor, con precisión t, está acotado de la siguiente manera:

$$|\delta| = \left|\frac{\varepsilon}{x}\right| < \frac{1/2 \times \beta^{e-t+1}}{1 \times \beta^{e}} = 1/2 \times \beta^{-t+1}$$

En el caso de $\beta = 10$, base 10, se tiene la siguiente definición:

Definición

Se dice que $\mathrm{fl}(x)$ aproxima a x con p dígitos significativos si p es el mayor entero no negativo para el cual se cumple que

$$\delta < \frac{1}{2} \times 10^{1-p} = 5 \times 10^{-p}$$

Ejercicio

Sea $x = 3.14159265$ *y sea* $x^* = 3.141591$ *una aproximación de* x *obtenida por medio de un algoritmo. Determine el error absoluto, el error relativo y la cantidad de dígitos significativos con los que* x^* *aproxima a* x.

Ejercicio

Sea $x = 3.14159265$ *y sea* $x^* = 3.141591$ *una aproximación de* x *obtenida por medio de un algoritmo. Determine el error absoluto, el error relativo y la cantidad de dígitos significativos con los que* x^* *aproxima a* x.

Solución:

$$\varepsilon_{ab} = |x - x^*| = |3.14159265 - 3.141591| = 1.65 \times 10^{-6}$$

$$\delta_{rel} = \frac{|x - x^*|}{|x|} = \frac{|3.14159265 - 3.141591|}{|3.141591|} \approx 5.2521 \times 10^{-7}$$

Note que $\varepsilon_{rel} \approx 5.2521 \times 10^{-7} < 5 \times 10^{-6}$, esto es, x^* aproxima a x con a lo sumo $p = 6$ dígitos significativos.

Ejercicio

Sea $x = 1.4142$ *y sea* $x^* = 1.414$*. Determine el error absoluto, el error relativo y la cantidad de dígitos significativos con los que* x^* *aproxima a* x*.*

Ejercicio

Sea $x = 1.4142$ *y sea* $x^* = 1.414$. *Determine el error absoluto, el error relativo y la cantidad de dígitos significativos con los que* x^* *aproxima a* x.

Solución:

$$\varepsilon_{ab} = |x - x^*| = |1.4142 - 1.414| = 2.0 \times 10^{-4}$$

$$\delta_{rel} = \frac{|x - x^*|}{|x|} = \frac{2.0 \times 10^{-4}}{|1.4142|} = \times 10^{-7}$$

Truncamiento y Redondeo

Definición

Truncamiento Sea $x = \pm(d_1.d_2 \cdots d_t d_{t+1} \cdots)_\beta \times \beta^e$*, entonces podemos aproximar* x *por truncamiento hasta el dígito* t *como* $\mathrm{fl}_T(x) = \pm(d_1.d_2 \cdots d_t)_\beta \times \beta^e$.

Definición

Redondeo Sea $x = \pm(d_1.d_2 \cdots d_t d_{t+1} \cdots)_\beta \times \beta^e$*, entonces podemos aproximar* x *por redondeo hasta el dígito* t *como*

$$\mathrm{fl}_R(x) = \begin{cases} \pm(d_1.d_2 \cdots d_t)_\beta \times \beta^e, & \textit{si } d_{t+1} < \dfrac{\beta}{2}, \\ \pm(d_1.d_2 \cdots (d_t + 1))_\beta \times \beta^e, & \textit{si } d_{t+1} \geq \dfrac{\beta}{2} \end{cases}$$

La operación de redondeo $\mathrm{fl}: \mathbb{R} \to \mathbb{F}$ mapea $\xi \in \mathbb{R}$ al número máquina más cercano $\mathrm{fl}(\xi) = \hat{\xi} \in \mathbb{F}$. Dicha operación posee las siguientes propiedades:

1. Monotonía: $\xi \leq \eta \Rightarrow \mathrm{fl}(\xi) \leq \mathrm{fl}(\eta)$.
2. Idempotencia: $\mathrm{fl}(\xi) = \xi$, para $\xi \in \mathbb{F}$.

Teorema

Sea $x \neq 0$ un número real en el rango y $\mathrm{fl}(x)$ su representación punto flotante. Entonces, se cumple que

$$\frac{|x - \mathrm{fl}(x)|}{|x|} \leq \mu = \begin{cases} \beta^{1-t} & \textit{(truncamiento)} \\ \frac{1}{2}\beta^{1-t} & \textit{(redondeo)} \end{cases}$$

Ejemplo

Considere el conjunto $\mathbb{F}_{10,3}$ en el cual se utiliza redondeo. Escriba la versión punto flotante de

$$a = 1.23456, \quad b = -0.1988, \quad c = 5062.2$$

Solución:

Ejemplo

Considere el conjunto $\mathbb{F}_{10,3}$ en el cual se utiliza redondeo. Escriba la versión punto flotante de

$$a = 1.23456, \quad b = -0.1988, \quad c = 5062.2$$

Solución: Primero escribimos la forma normalizada de los números:

$$a = 1.23456 \times 10^0, \quad b = -1.988 \times 10^{-1}, \quad c = 5.0622 \times 10^3$$

se tiene una precisión de $t = 3$, entonces

- $\mathrm{fl}(a) = \mathrm{fl}(1.23456 \times 10^0) = 1.23 \times 10^0$.
- $\mathrm{fl}(b) = \mathrm{fl}(-1.988 \times 10^{-1}) = -1.99 \times 10^{-1}$.
- $\mathrm{fl}(c) = \mathrm{fl}(5.0622 \times 10^3) = 5.06 \times 10^3$.

Operaciones punto flotante

La IEEE 754 define que las operaciones aritméticas y la operación raíz cuadrada son calculadas para números máquina por un redondeo correcto para el resultado exacto que se persigue: si se denota la realización de una operación como $\star$ por $\hat{\star}$, entonces, para $x, y \in \mathbb{F}$ se tiene que

$$x \hat{\star} y = \mathrm{fl}(x \star y), \quad \star \in \{+, -, \cdot, \div, \sqrt{}\}$$

Entonces, para $x, y \in \mathbb{F}$, $\star \in \{+, -, \cdot, \div, \sqrt{}\}$, existe δ, $|\delta| \leq \varepsilon_m$, tal que

$$x \hat{\star} y = (x \star y)(1 + \delta)$$

para la realización-máquina $\hat{\star}$ de la operación $\star$.

Operaciones punto flotante

La idea es evitar **overflow** o **underflow**. Así, las operaciones aritméticas en precisión finita se definen así:

Definición

- *Suma:* $x \oplus y = \mathrm{fl}(\mathrm{fl}(x) + \mathrm{fl}(y))$.
- *Resta:* $x \ominus y = \mathrm{fl}(\mathrm{fl}(x) - \mathrm{fl}(y))$.
- *Multiplicación:* $x \odot y = \mathrm{fl}(\mathrm{fl}(x) \cdot \mathrm{fl}(y))$.
- *División:* $x \oslash y = \mathrm{fl}(\mathrm{fl}(x) \div \mathrm{fl}(y))$.

Nota

La suma y el producto son conmutativas pero no asociativas. Tampoco se cumple la ley distributiva.

Ejemplo

Nuevamente en $\mathbb{F}_{10,3}$*, considere*

$$a = 1.23456 \times 10^{0}, \quad b = -1.988 \times 10^{-1}, \quad c = 5.0622 \times 10^{3}$$

Calcule el resultado de $a \oplus b$ *y* b/c*.*

Solución:

Ejemplo

Nuevamente en $\mathbb{F}_{10,3}$, considere

$$a = 1.23456 \times 10^0, \quad b = -1.988 \times 10^{-1}, \quad c = 5.0622 \times 10^3$$

Calcule el resultado de $a \oplus b$ y b/c.

Solución:

$$\begin{aligned}
a \oplus b &= \mathrm{fl}(\mathrm{fl}(a) + \mathrm{fl}(b)) \\
&= \mathrm{fl}(1.23 \times 10^0 + 5.06 \times 10^3) \\
&= \mathrm{fl}(1.23 + 5060) \\
&= \mathrm{fl}(5061.23) \\
&= \mathrm{fl}(5.06123 \times 10^3) \\
&= 5.06 \times 10^3 \\
&= 5060
\end{aligned}$$

Para el caso de b/c se tiene

$$
\begin{aligned}
b \oslash c &= \mathrm{fl}(\mathrm{fl}(b) \div \mathrm{fl}(c)) \\
&= \mathrm{fl}(\mathrm{fl}(-1.99 \times 10^{-1}) \div \mathrm{fl}(5.06 \times 10^{3})) \\
&= \mathrm{fl}(-0.199 \div 5060) \\
&= \mathrm{fl}(-3.932806 \times 10^{-5}) \\
&= -3.93 \times 10^{-5}
\end{aligned}
$$

Comentarios:

1. Sean $x_1, x_2 \in \mathbb{R}$ dos números que se pueden representar en el sistema de punto flotante, entonces se cumple que

$$\begin{aligned} \mathrm{fl}(x_1) + \mathrm{fl}(x_2) &= x_1(1+\delta_1) + x_2(1+\delta_2) \\ &= x_1 + x_2 + (x_1\delta_1 + x_2\delta_2) \\ &= x_1 + x_2 + (\epsilon_1 + \epsilon_2) \end{aligned}$$

note que el error cometido depende de ϵ_1 y ϵ_2. En el peor de los casos, ambos poseen el mismo signo.

Comentarios:

2. Considere un caso donde una computadora tiene una precisión de $t = 6$ y considere los números:

$$x_1 = 1.00000, \quad x_2 = 9.99999 \times 10^{-1}$$

entonces, es fácil ver que $x_1 - x_2 = 0.000001$. Sin embargo, cuando la computadora calcula la diferencia, primero ajusta la magnitud para que x_1 y x_2 tengan la misma magnitud. Entonces, x_2 se transforma en 0.99999. Note que se perdió un dígito en la expansión de x_2.
El resultado de la computadora será $1.00000 - 0.99999 = 0.00001$.
Aquí se tiene que:
El error absoluto es $|0.000001 - 0.00001| = 0.000009$.
El error relativo es $0.000009/0.000001 = 9$.

Gracias por la atención.