Curso de Métodos Numéricos

DEMAT, Universidad de Guanajuato

Clase 2: Error de redondeo y propagación de errores

- Épsilon de la máquina
- Error absoluto y relativo
- Error en la representación de números
- Propagación de errores por la aritmética

MAT–251

Dr. Joaquín Peña Acevedo
CIMAT A.C.
e-mail: joaquin@cimat.mx

Resumen (I)

Entre las propiedades que son deseables para la representación de los números en la computadora tenemos:

Deseamos que la computadora maneje los números con

- Rapidez
- Exactitud
- Rango: se necesita representar tanto números grandes como pequeños
- Portabilidad: Los programas escritos en una máquina dada deben correr sobre diferentes máquinas sin requerir modificaciones.
- Facilidad de implementación y uso

Vimos que:

- En la computadora sólo se puede representar una cantidad finita de números.
- Se adopta el estándar de la IEEE.
- La limitante es la cantidad de bits que se usan para almacenar las representaciones de los números.

Ejemplo

Al convertir 0.9 de decimal a binario vemos que

Pasos del algoritmo	**Resultado**
0.9	$(0.)_2$
$0.9 \times 2 = 1.8$	$(0.1)_2$
$0.8 \times 2 = 1.6$	$(0.11)_2$
$0.6 \times 2 = 1.2$	$(0.111)_2$
$0.2 \times 2 = 0.4$	$(0.1110)_2$
$0.4 \times 2 = 0.8$	$(0.11100)_2$
$0.8 \times 2 = 1.6$	$(0.111001)_2$
$\vdots$	

Ejemplo

Al convertir 0.9 de decimal a binario vemos que

Pasos del algoritmo	**Resultado**
0.9	$(0.)_2$
$0.9 \times 2 = 1.8$	$(0.1)_2$
$0.8 \times 2 = 1.6$	$(0.11)_2$
$0.6 \times 2 = 1.2$	$(0.111)_2$
$0.2 \times 2 = 0.4$	$(0.1110)_2$
$0.4 \times 2 = 0.8$	$(0.11100)_2$
$0.8 \times 2 = 1.6$	$(0.111001)_2$
$\vdots$	

De modo que $0.9 = (0.1\overline{1100})_2$

Así, algunos números que tienen una expresión finita en base 10, su expresión en base 2 no es finita.

Notación $fl(x)$ (I)

Supongamos que tenemos las cantidades

$$x = (1.101101)_2 \times 2^4, \qquad y = (1.0101)_2 \times 2^{-1}.$$

Para calcular la suma de estos valores tenemos que expresarlos de modo que la base tenga el mismo exponente:

$$\begin{array}{rcl} x & = & (1.1011010000)_2 \times 2^4 \\ y & = & (0.0000010101)_2 \times 2^4 \\ \hline x+y & = & (1.1011100101)_2 \times 2^4 \end{array}$$

Denotamos por $fl(x)$ a la representación de un número x en la computadora.

Entonces si tenemos una máquina que puede representar sólo 5 dígitos para la fracción y que realiza truncamiento, tenemos

$$fl(x) = (1.10110)_2 \times 2^4, \qquad fl(y) = (1.01010)_2 \times 2^{-1}.$$

Notación $fl(x)$ (II)

Y la suma de estas cantidades es

$$\begin{array}{rl} & (1.10110)_2 \times 2^4 \\ + & (0.00000)_2 \times 2^4 \\ \hline & (1.10110)_2 \times 2^4 \end{array}$$

Por lo que en este caso $fl(\, fl(x) + fl(y)\,) = fl(x)$.

Así, en la computadora si tenemos que $x + y = x$, esto no implica que $y = 0$.

Epsilon de la máquina (I)

Se define el *épsilon de la máquina* como el número $\epsilon_m > 0$ que se puede representar en la computadora y que es el más pequeño tal que es verdadero que

$$fl(1.0+\epsilon_m) = 1.0+\epsilon_m > 1.0 = fl(1.0)$$

Se puede decir que el épsilon de la máquina es la distancia entre el número 1.0 y el siguiente número de punto flotante más grande.

Si p es la cantidad de bits para representar la parte fraccionaria de la mantisa,

$$\begin{aligned} fl(1.0) &= (1.0000\cdots00)_2 \times 2^0 \\ fl(1.0+\epsilon_m) &= (1.\underbrace{0000\cdots01}_{p \text{ dígitos}})_2 \times 2^0 \end{aligned}$$

Por tanto,

Epsilon de la máquina (II)

$$\epsilon_m = fl(1.0+\epsilon_m) - fl(1.0) = (0.\underbrace{00\cdots01}_{p \text{ dígitos}})_2 \times 2^0 = (1.0)_2 \times 2^{-p}$$

Dado que

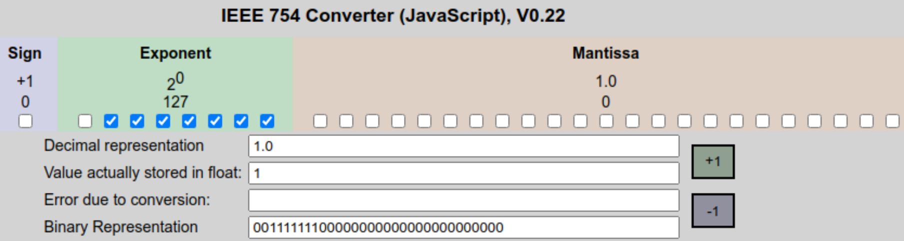
	signo	exponente	fracción
32 bits	1	8	$p = 23$
64 bits	1	11	$p = 52$

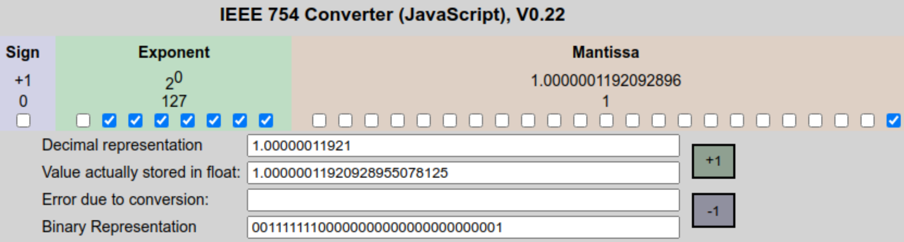
Para número de punto flotante de 32 bits, $\epsilon_m = 2^{-23} = 1.19 \times 10^{-7}$,

Para 64-bits, se tiene que $\epsilon_m = 2^{-52} = 2.2204 \times 10^{-16}$.

Epsilon de la máquina (III)

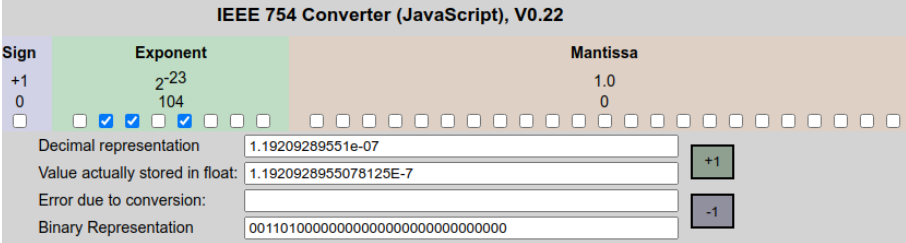
$$(-1)^0\,(1.00000000000000000000000)_2 \times 2^0$$





Para 32 bits el épsilon de la máquina es:

$$2^{-23} = (-1)^0\,(1.00000000000000000000000)_2 \times 2^{104-127}$$



El número p de dígitos determina la *precisión* de la representación.

En caso de desconocer p, se puede calcular el épsilon de la máquina con el siguiente algoritmo.

Epsilon de la máquina (V)

Algoritmo 1: Cálculo del epsilon de la máquina

```
epsilon = 0.5;
unidad  = 1.0;
valor   = unidad + epsilon;
while valor > unidad,
    epsilon = epsilon/2;
    valor   = unidad + epsilon;
end
epsilon = epsilon*2;
```

Observaciones

- Cuando se trabaja con precisión finita, 0 no es la única solución de la ecuación $1 + x = 1$.

Observaciones

- Cuando se trabaja con precisión finita, 0 no es la única solución de la ecuación $1 + x = 1$.
  Por ejemplo, cualquier número en la computadora en el intervalo $[0, \epsilon_m)$ cumple la igualdad.

Observaciones

- Cuando se trabaja con precisión finita, 0 no es la única solución de la ecuación $1 + x = 1$.
  Por ejemplo, cualquier número en la computadora en el intervalo $[0, \epsilon_m)$ cumple la igualdad.
- El orden en que se realizan las operaciones es importante. Ejemplo:

  $\frac{\epsilon_m}{2}$ es representable en la computadora y queremos calcular la suma

$$1 + \frac{\epsilon_m}{2} + \frac{\epsilon_m}{2}$$

  Entonces dependiendo del orden en que se realicen las operaciones se tienen diferentes resultados:

$$fl\left(fl\left(1 + \frac{\epsilon_m}{2}\right) + \frac{\epsilon_m}{2}\right) \neq fl\left(1 + fl\left(\frac{\epsilon_m}{2} + \frac{\epsilon_m}{2}\right)\right)$$

Error absoluto y error relativo

El *error absoluto* entre dos números reales x y y es

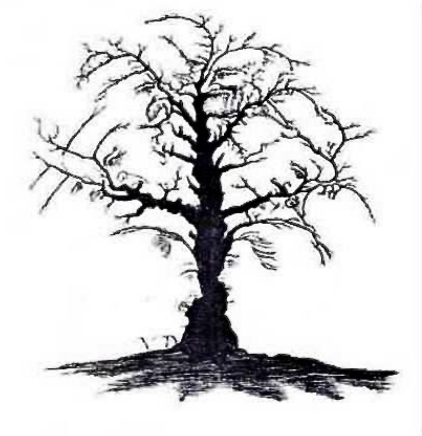
$$\boxed{|x-y|.}$$

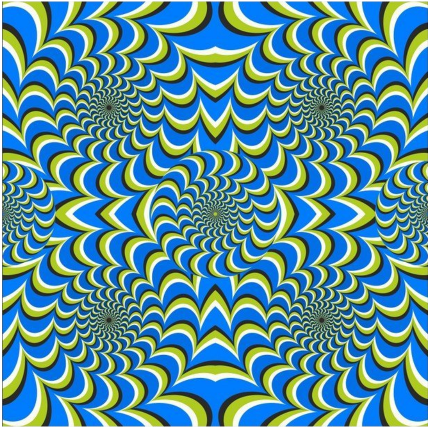
Supongamos que $y \neq 0$ es el verdadero valor y x es una aproximación. El *error relativo* es

$$\boxed{\frac{|x-y|}{|y|}}.$$

El *error de redondeo* es el error que se produce entre el valor real x y su representación en la computadora $fl(x)$, sin importar si se produce por redondeo o truncamiento.

Error absoluto y error relativo





¿Hay 10 u 11 rostros? ¿Hay 1200 ó 1201 arcos?

¿En caso se comete el mayor error?

Error en la representación numérica (I)

El número máquina anterior a *fl*(1.0) es

$$(1.\underbrace{11\cdots11}_{p \text{ dígitos}})_2 \times 2^{-1} = 2(1-2^{-p-1}) \times 2^{-1} = 1-2^{-p-1} = 1-\frac{\epsilon_m}{2}$$

De este modo las distancias entre el número que antecede a 1.0 y el que lo precede son

$$\boxed{\frac{\epsilon_m}{2} \qquad \text{y} \qquad \epsilon_m}$$

De este modo, lo que observamos es que la distancia entre 1.0 y los dos números máquina más cercanos es diferente.

Queremos ver lo que ocurre en el caso general:

Error en la representación numérica (II)

Sea x un número real positivo. Su representación en base 2 es:

$$x = (1.d_1 d_2 \ldots d_p d_{p+1} \ldots)_2 \times 2^e,$$

los números de máquina más cercanos a x son

$$\begin{aligned} x_- &= (1.d_1 d_2 \ldots d_p)_2 \times 2^e, \\ x_+ &= [(1.d_1 d_2 \ldots d_p)_2 + 2^{-p}] \times 2^e. \end{aligned}$$

Si x_- es el más cercano, el error absoluto es

$$|x - x_-| = (0.0 \ldots 0 d_{p+1} \ldots)_2 \times 2^e = (0.d_{p+1} \ldots)_2 \times 2^{e-p} \leq 2^{e-p}.$$

Entonces el error relativo es

$$\left| \frac{x - x_-}{x} \right| \leq \frac{2^{e-p}}{(1.d_1 \ldots d_{p+1} \ldots)_2 \times 2^e} = \frac{2^{-p}}{(1.d_1 \ldots d_{p+1} \ldots)_2} \leq \frac{2^{-p}}{1} = 2^{-p} = \epsilon_m.$$

Por otra parte, si x_+ es el más cercano, entonces

Error en la representación numérica (III)

$$|x - x_+| \leq \frac{1}{2}|x_+ - x_-| \leq \frac{1}{2} 2^{-p}\, 2^e,$$

por lo que

$$\frac{|x - x_+|}{|x|} \leq \frac{1}{2}\,\frac{2^{-p}\, 2^e}{1.0 \times\, 2^e} = \frac{2^{-p}}{2} = \frac{\epsilon_m}{2}$$

Definimos la *unidad de error de redondeo* u como

$$u = \begin{cases} \epsilon_m & \text{para redondeo hacia abajo} \\ \frac{\epsilon_m}{2} & \text{para redondeo hacia arriba} \end{cases}$$

Error de redondeo

La relación entre un número real y el número de máquina que lo representa está dada por $fl(x) = x(1 + \delta)$, donde $|\delta| \leq u$.

Operaciones con números de punto flotante (I)

Dados dos números máquina a y b, en el modelo estándar de aritmética de punto flotante se tiene que

$$\boxed{fl(a \circ b) = (a \circ b)(1+\delta)}$$

donde $\circ$ es uno de los operadores $\{+, -, \times, /\}$, y $|\delta| < u$.

Con este modelo podemos ver que

$$fl(a+b) = fl(b+a),$$

pero si queremos calcular la suma $a+b+c$, entonces

$$fl(fl(a+b)+c) \neq fl(a+fl(b+c))$$

Errores al realizar operaciones aritméticas (I)

- La asociatividad en la suma puede no ser válida.

- Hay que especificar el tipo de redondeo que se afectúa.

Al calcular 521000×0.0365 usando tres dígitos de precisión, tenemos que

Errores al realizar operaciones aritméticas (I)

- La asociatividad en la suma puede no ser válida.

- Hay que especificar el tipo de redondeo que se afectúa.

Al calcular 521000×0.0365 usando tres dígitos de precisión, tenemos que

$$a = 525000 = 0.525 \times 10^6, \qquad b = 0.365 \times 10^{-1}$$

$$fl(ab) = fl(0.191625 \times 10^5) = \begin{cases} 0.192 \times 10^5 & \text{Redondeo hacia arriba} \\ 0.191 \times 10^5 & \text{Redondeo hacia abajo} \end{cases}$$

Errores al realizar operaciones aritméticas (II)

- Errores por sustracción.

Sea $f(x) = (1 - \cos x)/x^2$. Para $x = 1.2 \times 10^{-5}$ y una precisión a 10 decimales, se tiene que

$$\begin{aligned} \cos x = 0.9999999999 \quad &\Longrightarrow \quad 1 - \cos x = 0.0000000001 \\ &\Longrightarrow \quad \frac{1 - \cos x}{x^2} = \frac{10^{-10}}{1.44 \times 10^{-10}} \approx 0.6944\ldots \end{aligned}$$

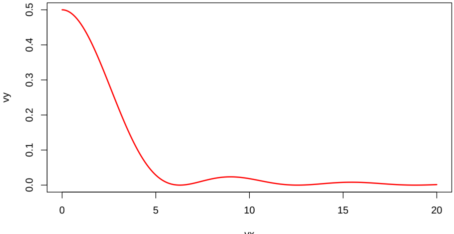
Errores al realizar operaciones aritméticas (II)

- Errores por sustracción.

Sea $f(x) = (1 - \cos x)/x^2$. Para $x = 1.2 \times 10^{-5}$ y una precisión a 10 decimales, se tiene que

$$\cos x = 0.9999999999 \quad \Longrightarrow \quad 1 - \cos x = 0.0000000001$$
$$\Longrightarrow \quad \frac{1-\cos x}{x^2} = \frac{10^{-10}}{1.44 \times 10^{-10}} \approx 0.6944\ldots.$$

El resultado es incorrecto. Resulta que $0 \leq f(x) < 0.5$ para todo $x \neq 0$.



Errores al realizar operaciones aritméticas (III)

Para evitarlo, podemos usar $\cos x = 1 - 2\sin^2(x/2)$.

$$f(x) = \frac{1}{2}\left(\frac{\sin(x/2)}{x/2}\right)^2.$$

$$x = 1.2 \times 10^{-5} \qquad \Longrightarrow \qquad f(x) = \frac{1}{2}\left(\frac{0.000006}{0.000006}\right)^2 = 0.5$$

Propagación del error en la suma

Supongamos que tenemos dos números reales x, y con el mismo signo, y que

$$fl(x) = x(1+\delta_x), \qquad fl(y) = y(1+\delta_y),$$

con $|\delta_x| \leq u$ y $|\delta_y| \leq u$.

El error relativo de la suma $x+y$ es

Propagación del error en la suma

Supongamos que tenemos dos números reales x, y con el mismo signo, y que

$$fl(x) = x(1+\delta_x), \qquad fl(y) = y(1+\delta_y),$$

con $|\delta_x| \leq u$ y $|\delta_y| \leq u$.

El error relativo de la suma $x+y$ es

$$\delta_{x+y} = \frac{[fl(x)+fl(y)]-(x+y)}{x+y} = \frac{fl(x)-x}{x+y} + \frac{fl(y)-y}{x+y} = \delta_x \frac{x}{x+y} + \delta_y \frac{y}{x+y}$$

$$\left|\delta_{x+y}\right| \leq u \frac{|x|+|y|}{|x+y|} = u$$

Propagación del error en la resta

Para la resta $x - y$, con $x \neq y$, se tiene lo siguiente:

$$\delta_{x-y} = \frac{[fl(x) - fl(y)] - (x - y)}{x - y} = \delta_x \frac{x}{x-y} - \delta_y \frac{y}{x-y}$$

$$|\delta_{x-y}| \leq |\delta_x| \frac{|x|}{|x-y|} + |\delta_y| \frac{|y|}{|x-y|} \leq u \frac{|x| + |y|}{|x-y|}$$