JI 2022

$0.1 + 0.2 \neq 0.3$?

$\mathbb{F}$, ou les malheurs du calcul

Roméo Molina Vincent Lafage

jeudi 15 novembre 2022

$0.1 + 0.2 \neq 0.3$?

$$\Sigma = a + b =^{?} c \qquad \Delta = a + b - c$$

avec

$$a = 0.1 \qquad b = 0.2 \qquad c = 0.3$$

$0.1 + 0.2 \neq 0.3$?

$$\Sigma = a + b =^{?} c \qquad \Delta = a + b - c$$

avec

$$a = 0.1 \qquad b = 0.2 \qquad c = 0.3$$

	a	b	c	Σ	Δ
S	0.100000001	0.200000003	0.300000012	0.300000012	0
D	0.10000000000000001	0.20000000000000001	0.29999999999999999	0.30000000000000004	$5.551\cdots10^{-17}$
X	0.10000000000000000001	0.20000000000000000003	0.30000000000000000011	0.30000000000000000011	0

$0.1 + 0.2 \neq 0.3$?

$$\Sigma = a + b =^{?} c \qquad \Delta = a + b - c$$

avec

$$a = 0.1 \qquad b = 0.2 \qquad c = 0.3$$

	a	b	c	Σ	Δ
S	0.100000001	0.200000003	0.300000012	0.300000012	0
D	0.10000000000000001	0.20000000000000001	0.29999999999999999	0.30000000000000004	$5.551 \cdots 10^{-17}$
X	0.10000000000000000001	0.20000000000000000003	0.30000000000000000011	0.30000000000000000011	0

$\Rightarrow \mathbb{D} \not\subset \mathbb{B}$: tout décimal n'est pas un binaire

donc la conversion en binaire repose sur un arrondi

$\frac{1}{5} = 0.2_{10} = 0.00\overline{1100}_2 \cdots \ominus 13421773 \times 2^{-26} = 0.2 + 2,98 \times 10^{-9}$

Décimaux *vs.* binaires
...et binaires *vs.* flottants

$\mathbb{D} = \left\{ \frac{n}{10^p}, n \in \mathbb{Z}, p \in \mathbb{N} \right\} = \mathbb{Z}[1/10]$ (décimal)
$\mathbb{B} = \left\{ \frac{n}{2^p}, n \in \mathbb{Z}, p \in \mathbb{N} \right\} = \mathbb{Z}[1/2]$ (binaire)
$\mathbb{B} \subset \mathbb{D}$ mais $\mathbb{D} \not\subset \mathbb{B}$: $\frac{1}{5} \in \mathbb{D}$, $\frac{1}{5} \notin \mathbb{B} \Rightarrow 0.1 + 0.2 \neq 0.3$ $\left(\frac{1}{5} = 0.00\overline{1100}_2 \ldots \right) \Rightarrow$ Pas de calcul financiers...

- clôture :
  $\forall (x, y) \in \mathbb{B}^2, \quad x + y \in \mathbb{B},$
  $\forall (x, y) \in \mathbb{B}^2, \quad x \times y \in \mathbb{B}$
- commutativité :
  $\forall (x, y) \in \mathbb{B}^2, \quad x + y = y + x,$
  $\forall (x, y) \in \mathbb{B}^2, \quad x \times y = y \times x$
- associativité :
  $\forall (x, y, z) \in \mathbb{B}^3, \quad x + (y + z) = (x + y) + z,$
  $\forall (x, y, z) \in \mathbb{B}^3, \quad x \times (y \times z) = (x \times y) \times z$
- distributivité :
  $\forall (x, y, z) \in \mathbb{B}^3, \quad x \times (y + z) = x \times y + x \times z$
- ordre total :
  $\forall (x, y, z) \in \mathbb{B}^3, \quad x \leq y$ et $y \leq z \Rightarrow x \leq z$ (transitivité) ;
  $\forall (x, y) \in \mathbb{B}^2, \quad x \leq y$ et $y \leq x \Rightarrow x = y$ (antisymétrie) ;
  $\forall x \in \mathbb{B}, \quad x \leq x$ (réflexivité) ;
  $\forall (x, y) \in \mathbb{B}^2, \quad x \leq y$ ou $y \leq x$ (totalité).
- topologie :
  $\mathbb{B} \subset \mathbb{D} \subset \mathbb{Q}$ sont denses dans $\mathbb{R} \Rightarrow$ approximations arbitrairement proches des réels

Décimaux *vs.* binaires
...et binaires *vs.* flottants

- clôture :
  $\exists (x, y) \in \mathbb{F}^2, \quad x + y \notin \mathbb{F}$,
  $\exists (x, y) \in \mathbb{F}^2, \quad x \times y \notin \mathbb{F}$
  $\Rightarrow$ arrondi (inexact) et extension $\overline{\mathbb{F}} = \mathbb{F} \cup \{\pm\texttt{Inf}\} \cup \{\texttt{NaN}\} \cup \{\texttt{0\_}\}$ overflow, underflow, invalid
- commutativité :
  $\forall (x, y) \in \mathbb{F}^2, \quad x + y = y + x$,
  $\forall (x, y) \in \mathbb{F}^2, \quad x \times y = y \times x$
- associativité :
  $\exists (x, y, z) \in \mathbb{F}^3, \quad x + (y + z) \neq (x + y) + z$,
  $\exists (x, y, z) \in \mathbb{F}^3, \quad x \times (y \times z) \neq (x \times y) \times z$
- distributivité :
  $\exists (x, y, z) \in \mathbb{F}^3, \quad x \times (y + z) \neq x \times y + x \times z$
- ordre total :
  $\forall (x, y, z) \in \mathbb{F}^3, \quad x \leq y$ et $y \leq z \Rightarrow x \leq z$ (transitivité) ;
  $\forall (x, y) \in \mathbb{F}^2, \quad x \leq y$ et $y \leq x \Rightarrow x = y$ (antisymétrie) ;
  $\forall x \in \mathbb{F}, \quad x \leq x$ (réflexivité) ;
  $\forall (x, y) \in \mathbb{F}^2, \quad x \leq y$ ou $y \leq x$ (totalité).
  $\exists (x, y) \in \overline{\mathbb{F}}^2, \quad x \leq y$ et $y \leq x$ (NaN).
- topologie :
  $\mathbb{B} \subset \mathbb{D} \subset \mathbb{Q}$ sont denses dans $\mathbb{R} \Rightarrow$ approximations arbitrairement proches des réels
  mais
  $\mathbb{F}$: nombres à virgule flottante, les parties finies de $\mathbb{B}$ (ou $\mathbb{D}$) sont denses nulle part

(Images du) monde flottant

Revisiting "What Every Computer Scientist Should Know About Floating-point Arithmetic"

Impact des erreurs d'arrondi

$$P = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

with $x = 77617$ and $y = 33096$

[S.M. Rump, 1983, *"How reliable are results of computers"*]

Impact des erreurs d'arrondi

$$P = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

with $x = 77617$ and $y = 33096$

`float` : $P = -6.33825300e + 29$

[S.M. Rump, 1983, *"How reliable are results of computers"*]

Impact des erreurs d'arrondi

$$P = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

with $x = 77617$ and $y = 33096$

`float` : $P = -6.33825300e + 29$

`double` : $P = -1.1805916207174113e + 021$

[S.M. Rump, 1983, *"How reliable are results of computers"*]

Impact des erreurs d'arrondi

$$P = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

with $x = 77617$ and $y = 33096$

`float` :	$P = -6.33825300e + 29$
`double` :	$P = -1.1805916207174113e + 021$
`long double` :	$P = +5.76460752303423489188e + 17$

[S.M. Rump, 1983, *"How reliable are results of computers"*]

Impact des erreurs d'arrondi

$$P = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

with $x = 77617$ and $y = 33096$

`float` :	$P = -6.33825300e + 29$
`double` :	$P = -1.18059162071741 13e + 021$
`long double` :	$P = +5.76460752303423489188e + 17$
`quad` :	$P = +1.17260394005317863185883490452018380$

[S.M. Rump, 1983, *"How reliable are results of computers"*]

Impact des erreurs d'arrondi

$$P = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y)$$

with $x = 77617$ and $y = 33096$

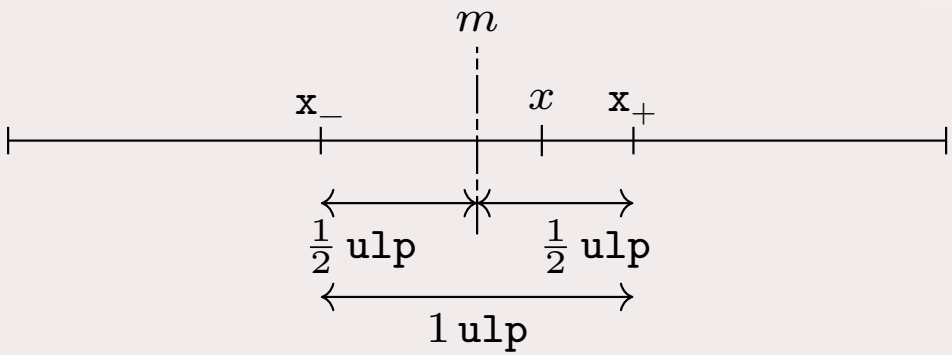
`float` :	$P = -6.33825300e + 29$
`double` :	$P = -1.1805916207174113e + 021$
`long double` :	$P = +5.76460752303423489188e + 17$
`quad` :	$P = +1.17260394005317863185883490452018380$
exact :	$P \approx -0.82739605994682136814116509547981629 2$
	$P = -\frac{54767}{66192}$

[S.M. Rump, 1983, *"How reliable are results of computers"*]

Comment contrôler les erreurs d'arrondi ?

Arrondi

— arrondir au 23^{e} ou 53^{e} bit : *what could go wrong ?*



— performance de l'arrondi $\delta = \frac{|x-\mathtt{x}|}{\max(|x|,|\mathtt{x}|)}$:

- $\delta < \frac{1}{2}\,\mathtt{ulp}$: arrondi **correct**
- $\delta < 1\,\mathtt{ulp}$: arrondi **fidèle**
- $\delta >= 1\,\mathtt{ulp}$...

— choix de l'arrondi :

- au plus près (avec la nuance de l'arrondi des financiers)
- vers $+\infty$
- vers $-\infty$
- vers 0

— l'arrondi est **déterministe**, mais **non-linéaire** (hautement) :

— l'arrondi est **déterministe**, mais **sensible aux perturbations** :

— l'arrondi est **déterministe**, mais **dépendant de l'ordre des opérations** :
(pas d'associativité en virgule flottante)

$\Rightarrow$ effet chaotique

$\Rightarrow$ ... surtout dans un enchaînement de calculs

Catastrophic Cancellation ?
Compensation Calamiteuse ?

Exceptionnellement base 10 (pas binaire) ! mantisse : 3 chiffres
Pour $a = 3.34$ et $b = 3.33$

- $a \ominus b = 0.01 \Rightarrow$ **compensation** (réduction de la précision relative)
  mais **bénigne** (le résultat flottant est exact : $a \ominus b = a - b$)
- $\begin{cases} a^2 - b^2 & = 0.0667 = 6.67 \times 10^{-2} \\ a \otimes a \ominus b \otimes b & = 0.1 = 1.00 \times 10^{-1} \end{cases}$
  50% d'erreur relative du résultat, ou 333 `ulp`, aucun chiffre n'est correct :
  compensation calamiteuse
- Quand advient-elle ?
- Combien de chiffres sont perdus ?

Plus, le risque d'***overflow***
$\Rightarrow$ Factorisons !

$$(a \oplus b) \otimes (a \ominus b) = 6.67 \otimes 0.01 = 6.67 \times 10^{-2} \qquad \text{exact}$$

$\Rightarrow$ The Right Way™

Quadratique

$$\begin{aligned} ax^2 + bx + c &= 0 \qquad (a \neq 0) \\ \Delta &= b^2 - 4ac \\ x_\pm &= \frac{-b \pm \sqrt{\Delta}}{2a} \end{aligned}$$

compensations calamiteuses (« catastrophic cancelation ») :

Quadratique

$$\begin{aligned} ax^2 + bx + c &= 0 \qquad (a \neq 0) \\ \Delta &= b^2 - 4ac \\ x_\pm &= \frac{-b \pm \sqrt{\Delta}}{2a} \end{aligned}$$

compensations calamiteuses (« catastrophic cancelation ») :

- entre $-b$ et $\sqrt{\Delta}$

$$\Rightarrow q = -b - \mathrm{sgn}(b)\sqrt{\Delta} = -\mathrm{sgn}(b)\left(|b| + \sqrt{\Delta}\right)$$

$$\begin{cases} x_1 = \frac{q}{2a} \\ x_2 = \frac{2c}{q} = \frac{c}{a x_1} \end{cases}$$

Quadratique

$$\begin{aligned} ax^2 + bx + c &= 0 \qquad (a \neq 0) \\ \Delta &= b^2 - 4ac \\ x_\pm &= \frac{-b \pm \sqrt{\Delta}}{2a} \end{aligned}$$

compensations calamiteuses (« catastrophic cancelation ») :

- entre $-b$ et $\sqrt{\Delta}$

$$\Rightarrow q = -b - \mathrm{sgn}(b)\sqrt{\Delta} = -\mathrm{sgn}(b)\left(|b| + \sqrt{\Delta}\right)$$

$$\begin{cases} x_1 = \frac{q}{2a} \\ x_2 = \frac{2c}{q} = \frac{c}{a x_1} \end{cases}$$

- discriminant $\Delta = b^2 - 4ac$ $\Rightarrow$ `fma`

Baker, Henry G., *"You Could Learn a Lot from a Quadratic : Overloading Considered Harmful"* 1998

Aire du triangle

Heron d'Alexandrie, aire S en fonction des longueurs a, b et c des cotés

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

$p = \frac{a+b+c}{2}$ demi-périmètre

Symétrique, mais instable numériquement, pour les triangles en épingle (confrontation de grandes et de petites valeurs)

Kahan Ré-étiquetage : $a > b > c$

$$\frac{1}{4}\sqrt{[a+(b+c)]\,[c-(a-b)]\,[c+(a-b)]\,[a+(b-c)]}$$

Symétrie apparente perdue, mais formule beaucoup plus robuste

Origine déterminantale

$$S = \frac{1}{4}\sqrt{\begin{vmatrix} 0 & a^2 & b^2 & 1 \\ a^2 & 0 & c^2 & 1 \\ b^2 & c^2 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{vmatrix}}$$

Volume du tétraèdre

$$V = \sqrt{\frac{1}{288}\begin{vmatrix} 0 & a^2 & b^2 & c^2 & 1 \\ a^2 & 0 & C^2 & B^2 & 1 \\ b^2 & C^2 & 0 & A^2 & 1 \\ c^2 & B^2 & A^2 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{vmatrix}}$$

$$X = (c - A + b)(A + b + c) \qquad x = (A - b + c)(b - c + A)$$
$$Y = (a - B + c)(B + c + a) \qquad y = (B - c + a)(c - a + B)$$
$$Z = (b - C + a)(C + a + b) \qquad z = (C - a + b)(a - b + C)$$

$$\xi = \sqrt{xYZ} \qquad \eta = \sqrt{yZX} \qquad \zeta = \sqrt{zXY} \qquad \lambda = \sqrt{xyz}$$

$$V = \frac{1}{192abc}\sqrt{(\xi + \eta + \zeta - \lambda)(\lambda + \xi + \eta - \zeta)(\eta + \zeta + \lambda - \xi)(\zeta + \lambda + \xi - \eta)}$$

Stable moyennant un ré-étiquetage : ordonner les paires de côtés de sorte à ce que les 3 plus petites des 12 différences faciales soient prises en compte parmi les 9 utilisées.

Discrete Stochastic Arithmetic (DSA)

[Vignes'04]

arithmétique classique

$$A \oplus B \to R$$

$$R = 3.14237654356891$$

DSA

arrondi aléatoire

$$A_1 \oplus B_1 \to R_1$$

$$A_2 \oplus B_2 \to R_2$$

$$A_3 \oplus B_3 \to R_3$$

$$R_1 = \mathbf{3.14}1354786390989$$
$$R_2 = \mathbf{3.14}3689456834534$$
$$R_3 = \mathbf{3.14}2579087356598$$

- chaque opération est exécutée 3 fois avec arrondi aléatoire
- nombre de chiffres corrects du résultat (test de Student à $95\,\%$ CL)
  $\Rightarrow$ détection d'instabilités numériques

PSA instrumenté avec CADNA

Avant de jouer avec la résolution utilisée, explorons la précision effective

L'instrumentation du PSA par CADNA, est passée essentiellement par un changement de types

Cette exécution a révélé de nombreuses instabilités numeriques, et potentiellement des pertes de précision massives

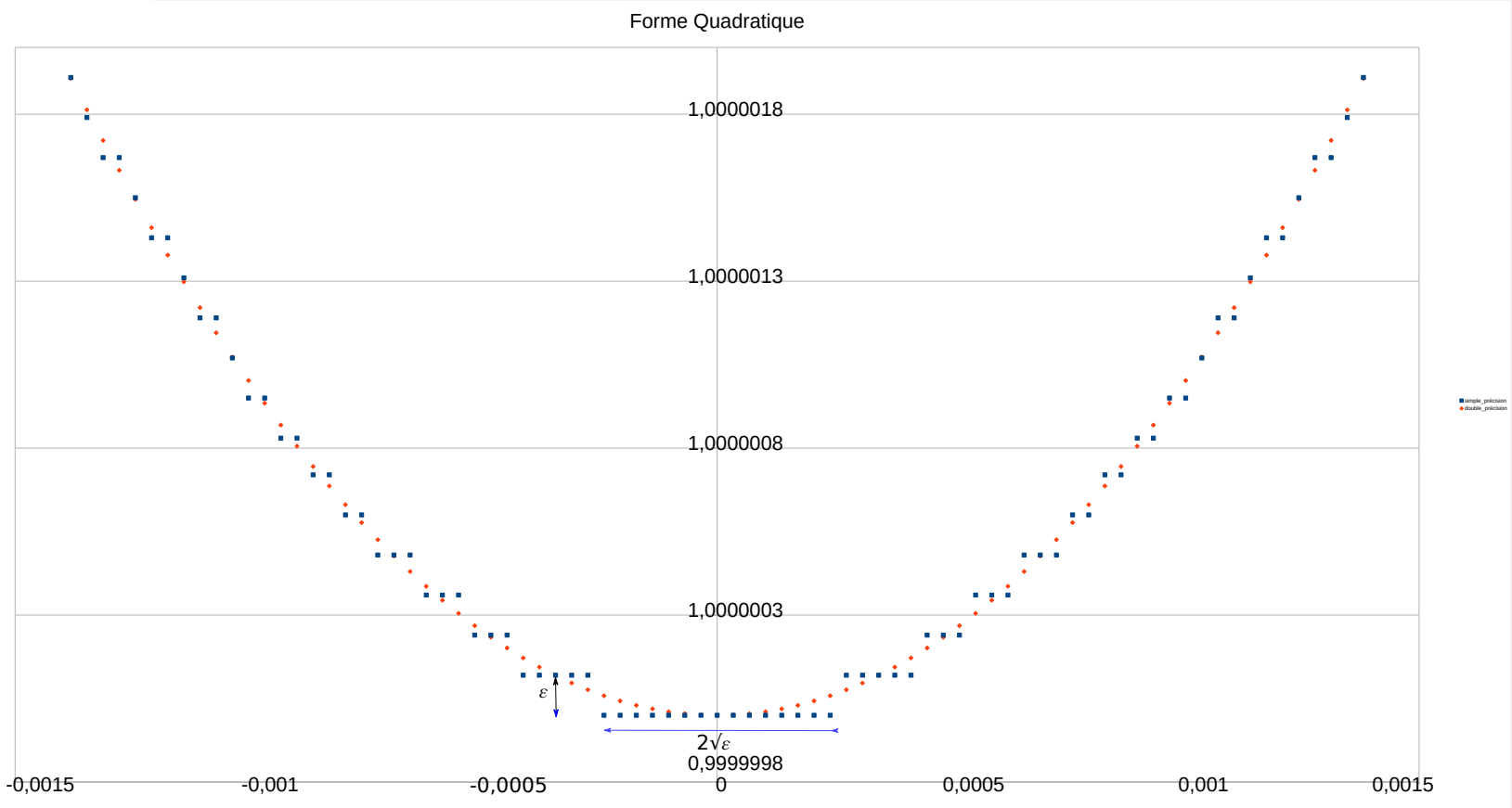
```
------------------------------------------------------------
CADNA_C 3.1.11 software

CRITICAL WARNING: the self-validation detects major problem(s).
The results are NOT guaranteed.

There are 2803679 numerical instabilities
124420 UNSTABLE MULTIPLICATION(S)
127753 UNSTABLE BRANCHING(S)
323243 UNSTABLE INTRINSIC FUNCTION(S)
266 UNSTABLE MATHEMATICAL FUNCTION(S)
2227997 LOSS(ES) OF ACCURACY DUE TO CANCELLATION(S)
------------------------------------------------------------
```

Minimisation

$$x \mapsto 1 + (x-1)^2 \quad \Rightarrow \quad f(x = x_0 + h) = f(x_0) + \underbrace{h \cdot \frac{\partial f}{\partial}}_{=0 \text{ à l'extremum}} + {}^t h \cdot \frac{\partial^2 f}{\partial \partial} \cdot h + o(h^2) \cdots \text{Taylor}$$



5793 `float` distincts au fond ; $\approx 190 \times 10^6$ `double` distincts

Conclusion

- changer la résolution (`float`, `double`) peut accélérer les calculs…
- …mais à quel prix en précision (et en sens) ?
- choisir une résolution plus élevée ne nous garantit pas la précision correspondante
  mais nous donne plus de marge de perte
- mesurez ou faites mesurer la performance de vos codes
- déterminez s'ils sont *CPU bound* ou *memory bound*
- mesurez ou faites mesurer la précision de vos calculs
- vérifiez la stabilité de vos algorithmes
- *« Attention, ces calculs ont été réalisés par des professionnels, n'essayez surtout pas de les reproduire chez vous ! »*