

Chapitre 1 : REPRÉSENTATION EN VIRGULE FLOTTANTE, STABILITÉ ET CONDITIONNEMENT

1 Représentation en virgule flottante

- Motivation : accidents dus aux erreurs numériques
- Représentation en virgule flottante
- Erreurs d'arrondi
- Standard IEEE
- Modèle d'arithmétique
- Analyse d'erreurs : règles de base
- Standard IEEE : compléments et résumé

2 Stabilité et conditionnement

- Erreur directe
- Stabilité directe
- Conditionnement

ACCIDENTS DUS AUX ERREURS NUMÉRIQUES

SYSTÈME PATRIOT :



FUSÉE ARIANE 5 :



- système américain d'interception des missiles ;
- pendant la guerre du Golf (1991), les erreurs d'arrondi dans l'estimation du temps de 0.34 secondes dans un des systèmes anti-missile ont causé une erreur sur la position du missile irakien d'environ un demi-kilomètre ;
- cela a coûté la vie à 28 soldats ;
- premier lancement (1996), 30 secondes après le décollage la fusée devient incontrôlable et est détruite ;
- la perte de contrôle est due au dépassement de la valeur maximale du registre qui contenait la vitesse horizontale ;
- coût : environ 500 millions de dollars.

Plus d'information sur <https://www-users.cse.umn.edu/~arnold/disasters/>

REPRÉSENTATION DES NOMBRES RÉELS

Représentation en virgule flottante (notation à gauche, valeur à droite) :

$$\pm \overline{0.d_1d_2 \cdots d_t} \cdot \beta^e = \pm \beta^e \sum_{i=1}^t \frac{d_i}{\beta^i}$$

où

- β est la base ($\beta = 2$ – représentation binaire, $\beta = 10$ – décimale) ;
- t est le nombre de chiffres significatifs ;
- d_i est le i ème chiffre significatif ($0 \leq d_i \leq \beta - 1$) ;

l'ensemble des chiffres significatifs $\overline{0.d_1d_2 \cdots d_t}$ forment la **mantisso** ;

- e est l'**exposant** ($e_{\min} \leq e \leq e_{\max}$).

EXEMPLES : avec $t = 3$ chiffres significatifs

- 2 est $\overline{0.200} \cdot 10^1$ en décimale (mais aussi $\overline{0.020} \cdot 10^2$ et $\overline{0.002} \cdot 10^3$)
 - ▶ vérifiez : $10^1(\frac{2}{10} + \frac{0}{10^2} + \frac{0}{10^3}) = 2$
- $1/2$ est $\overline{0.500} \cdot 10^0$ en décimale et $\overline{0.100} \cdot 2^0$ en binaire
 - ▶ vérifiez : $10^0(\frac{5}{10} + \frac{0}{10^2} + \frac{0}{10^3}) = \frac{1}{2} = 2^0(\frac{1}{2} + \frac{0}{2^2} + \frac{0}{2^3})$

REPRÉSENTATION DES NOMBRES RÉELS (SUITE)

Représentation en virgule flottante (notation à gauche, valeur à droite) :

$$\pm \overline{0.d_1d_2 \cdots d_t} \cdot \beta^e = \pm \beta^e \sum_{i=1}^t \frac{d_i}{\beta^i}$$

Certains réels ont de multiples représentations (ex : $\overline{0.200} \cdot 10^1$ et $\overline{0.020} \cdot 10^2$).

La représentation avec

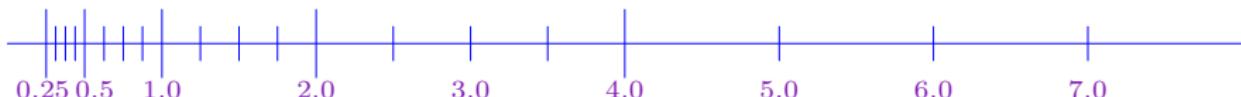
$$d_1 \neq 0$$

est **normalisée**. Dans une base binaire cette représentation implique $d_1 = 1$.

L'ensemble des réels possédant une représentation normalisée est noté

$$\mathbb{F} = \{ x \mid x = \pm \overline{0.d_1d_2 \cdots d_t} \cdot \beta^e, e_{\min} \leq e \leq e_{\max} \}.$$

EXEMPLE : représentation de la partie positive de \mathbb{F} pour $\beta = 2$, $t = 3$, $e_{\min} = -1$ et $e_{\max} = 3$.

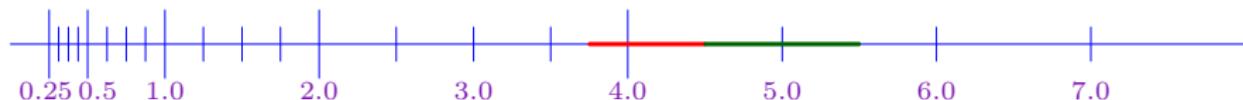


ERREURS D'ARRONDI : EXEMPLE

Comme l'ensemble \mathbb{F} des réels représentables est fini alors que \mathbb{R} est infini, les erreurs d'arrondi sont inévitables. Pour comprendre leurs effets, définissons

$$\text{fl}(x) = \text{le réel dans } \mathbb{F} \text{ le plus proche de } x \in \mathbb{R}.$$

EXEMPLE (SUITE) : $\text{fl}(x) = 4.0$ dans la région rouge et $\text{fl}(x) = 5.0$ dans la région verte



- La différence absolue $|\text{fl}(x) - x|$ peut être d'autant plus importante que x est grand ; elle vaut au plus la moitié de la distance entre deux éléments de \mathbb{F} qui entourent x .

Pour cet exemple cela donne

- ▶ pour $x \in]4, 7[$ on a $|\text{fl}(x) - x| \leq 0.5$
- ▶ pour $x \in]2, 4[$ on a $|\text{fl}(x) - x| \leq 0.25$
- ...
- ▶ pour $x \in]0.25, 0.5[$ on a $|\text{fl}(x) - x| \leq 2^{-5}$

- On constate par contre que différence relative est bornée ; pour cet exemple

$$\frac{|\text{fl}(x) - x|}{|x|} \leq \frac{1}{8}.$$

ERREURS D'ARRONDI : CAS GÉNÉRAL

ANALYSE : Considérons un $x \in \mathbb{R}$ tel que $|x|$ se trouve entre les valeurs positives minimale et maximale de \mathbb{F} . Soit

$$x = \pm \overline{0.d_1d_2\cdots} \cdot \beta^e$$

son expansion (potentiellement infinie) normalisée ($d_1 \neq 0$) en base β ; on a en particulier

$$|x| \geq \overline{0.1} \cdot \beta^e = \beta^{e-1}. \quad (1)$$

Par ailleurs, la différence $|\text{fl}(x) - x|$ vaut au plus la moitié de la distance entre deux éléments consécutifs x_+ , x_- de \mathbb{F} qui entourent x , et donc

$$|\text{fl}(x) - x| \leq \frac{1}{2} \cdot |x_+ - x_-| = \frac{1}{2} \cdot \overline{0.00\cdots 01} \cdot \beta^e = \frac{1}{2} \beta^{e-t}. \quad (2)$$

Inégalités (1) et (2) donnent une relation importante

$$\frac{|\text{fl}(x) - x|}{|x|} \leq \frac{\frac{1}{2}\beta^{e-t}}{\beta^{e-1}} = \frac{1}{2} \beta^{1-t} =: u, \quad (3)$$

où u est l'unité d'arrondi.

ERREURS D'ARRONDI : CAS GÉNÉRAL (SUITE)

On a donc l'erreur relative

$$\frac{|\text{fl}(x) - x|}{|x|} \leq u. \quad (3)$$

RAPPEL SUR LES ERREURS : Soient x et son approximation \hat{x} . Alors

- l'erreur absolue est $\epsilon_{\text{abs}} = |x - \hat{x}|$;
- l'erreur relative (pour $x \neq 0$) est $\epsilon_{\text{rel}} = \frac{|x - \hat{x}|}{|x|}$; en particulier

$$\hat{x} = x(1 + \epsilon), \quad |\epsilon| = \epsilon_{\text{rel}}$$

L'inégalité (3) est donc équivalente à

$$\boxed{\text{fl}(x) = x(1 + \epsilon), \quad |\epsilon| \leq u.} \quad (4)$$

De manière similaire à (4) on montre (cf. travaux pratiques) que

$$\text{fl}(x) = \frac{x}{1 + \epsilon'}, \quad |\epsilon'| \leq u. \quad (5)$$

STANDARD IEEE 754 (1985)

STANDARD IEEE :

- universellement accepté aujourd’hui
- deux principaux formats en virgule flottante pour $\beta = 2$:
single (simple précision) et **double** (double précision)
- représentation en mémoire



- spécifications principales :

single

1 bit	8 bits	23 bits
-------	--------	---------

$$e_{\min} = -125$$

$$e_{\max} = 128$$

$$x_{\min} \approx 1.2 \cdot 10^{-38}$$

$$x_{\max} \approx 3.4 \cdot 10^{38}$$

$$u \approx 6.0 \cdot 10^{-8}$$

double

1 bit	11 bits	52 bits
-------	---------	---------

$$e_{\min} = -1021$$

$$e_{\max} = 1024$$

$$x_{\min} \approx 2.2 \cdot 10^{-308}$$

$$x_{\max} \approx 1.8 \cdot 10^{308}$$

$$u \approx 1.1 \cdot 10^{-16}$$

- double précision est utilisée par défaut en Octave ;
regardez les commandes **realmax**, **realmin** et **eps** ($= 2u$).

MODÈLE D'ARITHMÉTIQUE

Modèle standard d'arithmétique en virgule flottante (satisfait avec IEEE) :
soient

- $x, y \in \mathbb{F}$
- $+, -, \cdot, /$ (symbolisées par \circ) les opérations habituelles dans \mathbb{R} ;
notez que $x \circ y$ n'est pas nécessairement dans \mathbb{F}
- $x_{\min} \leq |x \circ y| \leq x_{\max}$ (le résultat n'est ni trop grand, ni trop petit)
- $\oplus, \ominus, \odot, \oslash$ (symbolisées par \odot) les opérations en virgule flottante dans \mathbb{F} ;
avec donc $x \odot y \in \mathbb{F}$

alors

$$x \odot y = (x \circ y)(1 + \epsilon), \quad |\epsilon| \leq u. \quad (6)$$

INTERPRÉTATION : L'opération \odot mène aux erreurs d'arrondi comparables à celles de $\text{fl}(x \circ y)$.

Ce modèle permet de comprendre et prédire les effets des erreurs d'arrondi !

ANALYSE D'ERREURS : RÈGLES DE BASE

MODÈLE STANDARD D'ARITHMÉTIQUE :

$$x \odot y = (x \circ y)(1 + \epsilon), \quad |\epsilon| \leq u. \quad (6)$$

RÈGLES DE BASE :

Pour l'ensemble des règles, on suppose $|\epsilon_i|, |\epsilon'_i| \leq u$ et $\alpha, \beta \in \mathbb{R}$

$$\textcircled{1} \quad \alpha\epsilon_1 \pm \beta\epsilon_2 = (|\alpha| + |\beta|)\epsilon_3$$

$$\textcircled{2} \quad (1 + \alpha\epsilon_1)(1 + \beta\epsilon_2) = 1 + (|\alpha| + |\beta|)\epsilon_3 + |\alpha\beta|\mathcal{O}(u^2)$$

$$\textcircled{3} \quad \frac{1}{1 + \alpha\epsilon_4 + \mathcal{O}(u^2)} = 1 + \alpha\epsilon'_4 + |\alpha|^2\mathcal{O}(u^2)$$

(par développement en série de Taylor $1/(1+x) = 1 - x + \mathcal{O}(x^2)$,
avec $\epsilon'_4 = -\epsilon_4$)

NOTE : les facteurs devant $\mathcal{O}(u^2)$ sont omis dans la suite sous l'hypothèse
(souvent implicite) que $\alpha, \beta = \mathcal{O}(1)$.

MODÈLE D'ARITHMÉTIQUE : EXEMPLE 1

EXEMPLE 1 : Estimer l'erreur relative d'arrondi sur le résultat de

$$(1 \oslash 3) \odot 3$$

ANALYSE : (tous les ϵ_i satisfont $|\epsilon_i| \leq u$)

$$\begin{aligned}(1 \oslash 3) \odot 3 &= 1/3(1 + \epsilon_1) \odot 3 && (\text{par (6)}) \\&= (1/3(1 + \epsilon_1) \cdot 3)(1 + \epsilon_2) && (\text{par (6)}) \\&= 1 \cdot (1 + \epsilon_1)(1 + \epsilon_2) \\&= 1 \cdot (1 + 2\epsilon_3) + \mathcal{O}(u^2) && (\text{par } \textcircled{2})\end{aligned}$$

CONCLUSION : l'erreur relative en double précision est bornée (à $\mathcal{O}(u^2)$ près) par $2u \approx 2.2 \cdot 10^{-16}$.

MODÈLE D'ARITHMÉTIQUE : EXEMPLE 2

EXEMPLE 2 : soient x, y contaminés avec des erreurs d'arrondi :

- $\tilde{x} = x(1 + \epsilon_1) \in \mathbb{F}$, $|\epsilon_1| \leq u$,
 - $\tilde{y} = y(1 + \epsilon_2) \in \mathbb{F}$, $|\epsilon_2| \leq u$.

Quelle est l'erreur relative d'arrondi de $\tilde{x} \ominus \tilde{y}$ comme approximation de $x - y$?

ANALYSE : (avec comme avant $|\epsilon_i| \leq u$)

- $\widetilde{x} \ominus \widetilde{y} = (\widetilde{x} - \widetilde{y})(1 + \epsilon_3)$ (par (6))
 - $\widetilde{x} - \widetilde{y} = (x - y) + (x\epsilon_1 - y\epsilon_2) = (x - y)(1 + \delta u)$ avec

$$|\delta| = \left| \frac{x\epsilon_1 - y\epsilon_2}{x - y} \right| \cdot \frac{1}{u} \leq \frac{|x| + |y|}{|x - y|}$$
 - et donc

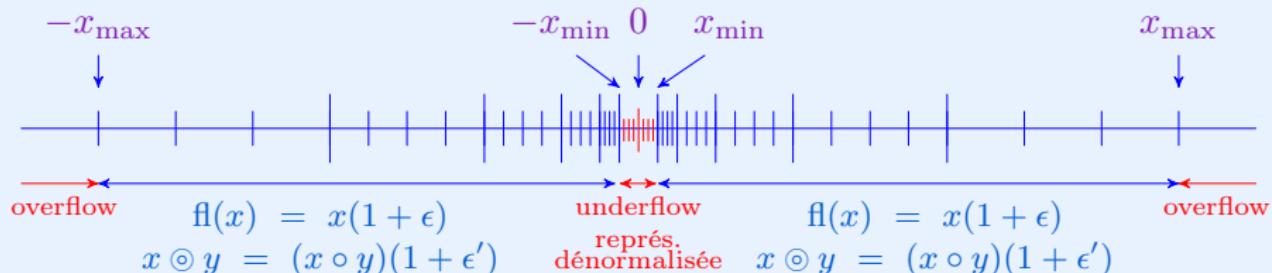
Si $x \approx y$, on a potentiellement $\delta \gg 1$ et donc un risque de perte de précision ; le phénomène est connu sous le nom d'**annulation**.

NOTE : c'est le phénomène qui s'est produit pour le calcul de la variance dans Exemple 6 du chapitre Motivation.

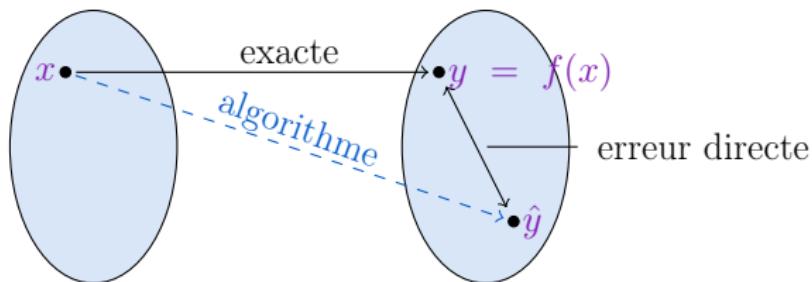
STANDARD IEEE : COMPLÉMENTS ET RÉSUMÉ

- Pour remédier à un changement brusque entre x_{\min} et 0 on utilise une représentation dénormalisée si $|x| < x_{\min}$ (pour 0 y compris) ; le dépassement de x_{\min} porte le nom d'**underflow**. En particulier, les relations (4), (6) ne sont plus valables et l'erreur relative peut facilement dépasser u .
- La représentation est aussi complétée avec
 - ▶ **$\pm\text{Inf}$** : provient du dépassement de x_{\max} ou $1/0$; c'est un **overflow** ;
 - ▶ **NaN** : résulte de $0/0$, $0 \cdot \infty$ ou (dans certains logiciels, pas dans Octave) $\sqrt{-1}$

RÉSUMÉ (avec $t = 3$, $e_{\min} = -1$, $e_{\max} = 3$) :



ERREUR DIRECTE



Soit un problème dont la solution (**exacte**) y est une fonction f des données x :

$$y = f(x).$$

Attention : x, y peuvent être des scalaires, vecteurs, matrices, etc...

EXEMPLES :

- évaluation d'une racine carrée : $f(x) = \sqrt{x}$
- soustraction de deux nombres : $f(x_1, x_2) = x_1 - x_2$

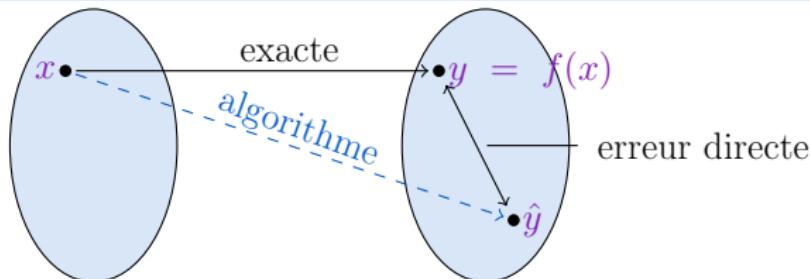
En arithmétique finie, un **algorithme** ne peut fournir qu'une **solution approchée** \hat{y} (qui dépend de cet algorithme !)

La différence $\hat{y} - y$ est appelée **erreur directe**.

INTUITION : Si cette erreur est petite (en norme), l'algorithme est stable.

Mais que veut dire «petite» ?

STABILITÉ DIRECTE



Mais que veut dire une «petite» erreur $\|\hat{y} - y\|/\|y\|$?

On considère qu'un algorithme a la **stabilité directe** en x si la norme de l'erreur directe $\|\hat{y} - y\| = \|\hat{y} - f(x)\|$ est comparable à la norme de l'erreur due aux effets d'arrondi.

En d'autres termes, si il existe $C_1, C_2 \geq 1$ (petits) tels que

$$\|\hat{y} - y\| \leq C_1 \|f(x + \delta x) - f(x)\|$$

pour au moins un δx tel que $\|\delta x\|/\|x\| \leq C_2 u$.

MOTIVATION :

- souvent x est déjà entaché d'erreurs d'arrondi (au moins) ;
- les premières opérations sur chaque élément de x introduisent des erreurs relatives au moins aussi grandes que u (cf. Exemple 1)

CONDITIONNEMENT

Un algorithme a la stabilité directe en x s'il existe $C_1, C_2 \geq 1$ tels que

$$\|\hat{y} - y\| \leq C_1 \|f(x + \delta x) - f(x)\|$$

pour au moins un δx tel que $\|\delta x\|/\|x\| \leq C_2 u$.

CONDITIONNEMENT

Un algorithme a la stabilité directe en x s'il existe $C_1, C_2 \geq 1$ tels que

$$\frac{\|\hat{y} - y\|}{\|y\|} \leq C_1 \frac{\|f(x + \delta x) - f(x)\|}{\|f(x)\|}$$

pour au moins un δx tel que $\|\delta x\|/\|x\| \leq C_2 u$.

Où on a utilisé $y = f(x)$.

CONDITIONNEMENT

Un algorithme a la stabilité directe en x s'il existe $C_1, C_2 \geq 1$ tels que

$$\frac{\|\hat{y} - y\|}{\|y\|} \leq C_1 \frac{\|f(x + \delta x) - f(x)\|}{\|f(x)\|}$$

pour au moins un δx tel que $\|\delta x\|/\|x\| \leq C_2 u$.

En particulier, nous nous intéresserons à

$$\frac{\|f(x + \delta x) - f(x)\|}{\|f(x)\|} = \underbrace{\frac{\|f(x + \delta x) - f(x)\| \|x\|}{\|f(x)\| \|\delta x\|}}_{\text{facteur d'amplification}} \cdot \underbrace{\frac{\|\delta x\|}{\|x\|}}_{\leq C_2 u}$$

Le pire des facteur pour $\|\delta x\|/\|x\|$ petit, à savoir

$$\kappa(x) := \lim_{\epsilon \rightarrow 0} \kappa_\epsilon(x), \quad \kappa_\epsilon(x) := \sup_{\|\delta x\| \leq \epsilon \|x\|} \frac{\|f(x + \delta x) - f(x)\| / \|f(x)\|}{\|\delta x\| / \|x\|}$$

est le conditionnement. En d'autres termes, le conditionnement est le pire des facteurs par lequel il faut multiplier les erreurs relatives dans les données x pour obtenir les erreurs relatives dans $f(x)$ (avec erreurs $\rightarrow 0$).

CONDITIONNEMENT (SUITE)

Un algorithme a la stabilité directe en x s'il existe un $C_1, C_2 \geq 1$ tel que

$$\frac{\|\hat{y} - y\|}{\|y\|} \leq C_1 \frac{\|f(x + \delta x) - f(x)\|}{\|f(x)\|}$$

pour au moins un δx tel que $\|\delta x\|/\|x\| \leq C_2 u$.

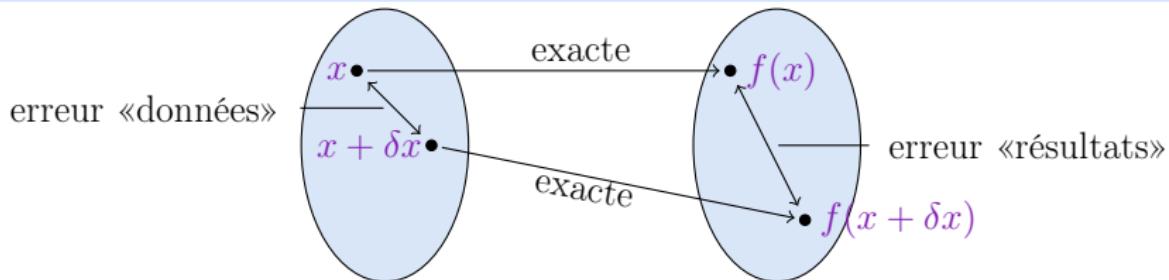
Pour revenir à la stabilité, on note (dans la limite de u infinitésimal) que

$$\frac{\|f(x + \delta x) - f(x)\|}{\|f(x)\|} \leq C_2 \kappa_\epsilon(x) u \approx C_2 \kappa(x) u.$$

Par conséquent, un algorithme possédant la stabilité directe satisfait aussi

$$\boxed{\frac{\|\hat{y} - y\|}{\|y\|} \leq C_1 C_2 \kappa_\epsilon(x) u \approx C_1 C_2 \kappa(x) u}$$

CONDITIONNEMENT (SUITE)



$$\kappa(x) := \lim_{\epsilon \rightarrow 0} \sup_{\|\delta x\| \leq \epsilon \|x\|} \frac{\|f(x + \delta x) - f(x)\| / \|f(x)\|}{\|\delta x\| / \|x\|} = \sup \frac{\text{erreur relative résultat}}{\text{erreur relative données}}$$

COMMENTAIRES :

- le conditionnement ne dépend pas d'un algorithme particulier (c.a.d \hat{y}) ; il ne dépend que du problème considéré (via $y = f(x)$).
- si $\kappa(x) \gg 1$ on parle d'un problème mal conditionné ; dans le cas contraire, il est bien conditionné.
- si $f(x)$ est différentiable (et $f'(x)$ est la matrice Jacobienne), on a

$$\kappa(x) = \frac{\|f'(x)\| \|x\|}{\|f(x)\|}$$

CONDITIONNEMENT : EXEMPLES

$$\kappa(x) = \frac{\|f'(x)\|\|x\|}{\|f(x)\|}$$

EXEMPLE 3 : Conditionnement de l'opération racine carrée \sqrt{x} (pour $x > 0$) .

$$f(x) = \sqrt{x}$$

et donc

$$\kappa(x) = \frac{|1/(2\sqrt{x})| |x|}{|\sqrt{x}|} = \frac{1}{2}.$$

CONCLUSION : C'est un problème bien conditionné ; un algorithme qui a la stabilité directe doit fournir une solution presque sans perte de précision.

EXEMPLE 4 : Conditionnement de la soustraction $x_1 - x_2$.

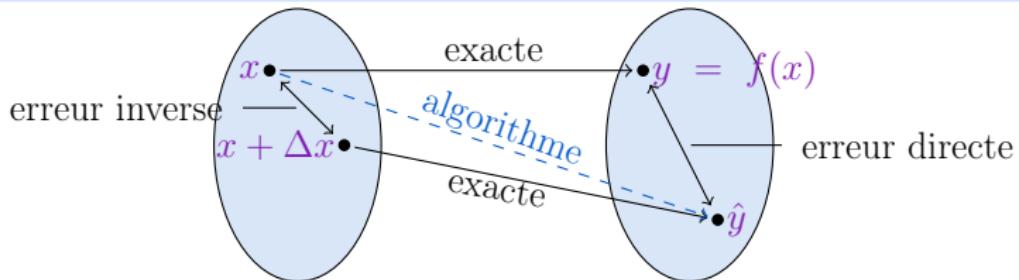
$$f(x_1, x_2) = x_1 - x_2, \quad f'(x_1, x_2) = (1, -1)$$

et donc (en utilisant la norme euclidienne pour $\| \|$)

$$\kappa(x) = \frac{\sqrt{2} \sqrt{x_1^2 + x_2^2}}{|x_1 - x_2|}.$$

CONCLUSION : Risque de perte de précision si $x_1 \approx x_2$. (on le savait déjà !)

ERREUR INVERSE, STABILITÉ INVERSE



- L'erreur inverse d'un algorithme \hat{y} est un Δx tel que

$$f(x + \Delta x) = \hat{y}; \quad (7)$$

(Un tel Δx n'existe pas nécessairement !)

- Un algorithme a la stabilité inverse si Δx satisfaisant (7) existe toujours et satisfait (pour un $C \geq 1$ petit)

$$\frac{\|\Delta x\|}{\|x\|} \leq Cu \quad (8)$$

- Comme la stabilité inverse implique

$$\|\hat{y} - y\| = \|f(x + \Delta x) - f(x)\|$$

pour un Δx de l'ordre de grandeur des erreurs d'arrondi (car (8)), elle implique la stabilité directe (avec $C_1 = 1$ et $C_2 = C$).