Introduction à l'Analyse Numérique

Quentin Louveaux

ULg - Institut Montefiore

2007

Cours d'analyse numérique

Qu'est-ce que c'est ?

- Savoir transposer la connaissance mathématique pure à un ordinateur aux performances finies
- Résoudre numériquement des problèmes dont la solution analytique est connue ou non
- Analyser le comportement des méthodes

Table des matières

Cours d'analyse numérique

Qu'est-ce que c'est ?

- Savoir transposer la connaissance mathématique pure à un ordinateur aux performances finies
- Résoudre numériquement des problèmes dont la solution analytique est connue ou non
- Analyser le comportement des méthodes

Table des matières

- Représentation des nombres et erreurs
- Interpolation et Régression
- Résolution d'équations non linéaires
- Résolution de systèmes linéaires
- Dérivation et Intégration numérique
- Résolution d'équations différentielles ordinaires

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`

    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B,...`
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3 ;4 5 6 ;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B,...`
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B,...`
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B,...`
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B`,...
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B`,...
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B,...`
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B`,...
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B`,...
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Matlab

- MATrix LABoratory
- Mode interactif
- Syntaxe très simple pour le calcul matriciel
  On peut utiliser des opérations aussi simples que
  - `A=[1 2 3;4 5 6;7 8 9]`
    $$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$
  - `A*b, rank(A), A(1, :), A.*B,...`
- De nombreuses fonctions de haut niveau sont pré-implémentées
  - Résolution de systèmes non linéaires
  - Opérations complexes sur des polynômes
  - Intégration numérique
  - Dessin de graphes en 2D ou 3D

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

  Si on utilise les 4 premiers termes, on a donc, comme approximation

$$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

  Si on utilise les 4 premiers termes, on a donc, comme approximation

$$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

Si on utilise les 4 premiers termes, on a donc, comme approximation

$$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

Si on utilise les 4 premiers termes, on a donc, comme approximation

$$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

  Si on utilise les 4 premiers termes, on a donc, comme approximation

$$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

  $$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

  Si on utilise les 4 premiers termes, on a donc, comme approximation

  $$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

Si on utilise les 4 premiers termes, on a donc, comme approximation

$$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Des exemples typiques de méthodes numériques

Des problèmes dont la solution analytique est connue

- Résolution de systèmes d'équations linéaires
- Evaluation de la constante e.
  A partir du théorème de Taylor,

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots$$

  Si on utilise les 4 premiers termes, on a donc, comme approximation

$$e \approx 1 + 1 + \frac{1}{2} + \frac{1}{6} + \frac{1}{120} \approx 2.71667$$

Des problèmes dont la solution analytique n'est pas connue

- Résolution d'équations non linéaires
- Approximation de l'intégrale définie d'une fonction
- Résolution de systèmes d'équations différentielles ordinaires
- Problèmes d'optimisation de fonctions avec contraintes

Résolution numérique d'une équation non linéaire

Cherchons une racine à l'équation

$$x^3 + x - 1 = 0.$$

Itér.	x	y	$z := \frac{x+y}{2}$	$f(z)$
0	0.000000	1.000000	0.500000	-0.375000
1	0.500000	1.000000	0.750000	0.171875
2	0.500000	0.750000	0.625000	-0.130859
3	0.625000	0.750000	0.687500	0.012451
4	0.625000	0.687500	0.656250	-0.061127
5	0.656250	0.687500	0.671875	-0.024830
6	0.671875	0.687500	0.679688	-0.006314
7	0.679688	0.687500	0.683594	0.003037
8	0.679688	0.683594	0.681641	-0.001646
9	0.681641	0.683594	0.682617	0.000694
10	0.681641	0.682617	0.682129	-0.000477

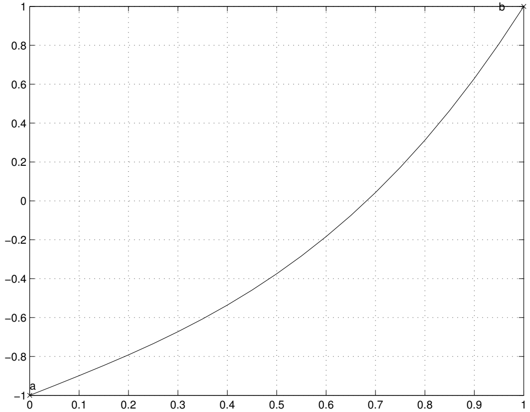
Résolution numérique d'une équation non linéaire

Cherchons une racine à l'équation

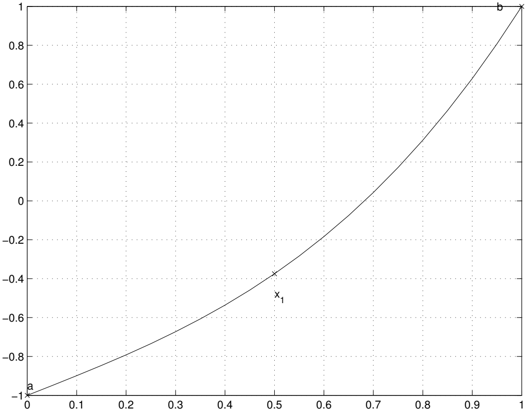
$$x^3 + x - 1 = 0.$$

Itér.	x	y	$z := \frac{x+y}{2}$	$f(z)$
0	0.000000	1.000000	0.500000	-0.375000
1	0.500000	1.000000	0.750000	0.171875
2	0.500000	0.750000	0.625000	-0.130859
3	0.625000	0.750000	0.687500	0.012451
4	0.625000	0.687500	0.656250	-0.061127
5	0.656250	0.687500	0.671875	-0.024830
6	0.671875	0.687500	0.679688	-0.006314
7	0.679688	0.687500	0.683594	0.003037
8	0.679688	0.683594	0.681641	-0.001646
9	0.681641	0.683594	0.682617	0.000694
10	0.681641	0.682617	0.682129	-0.000477

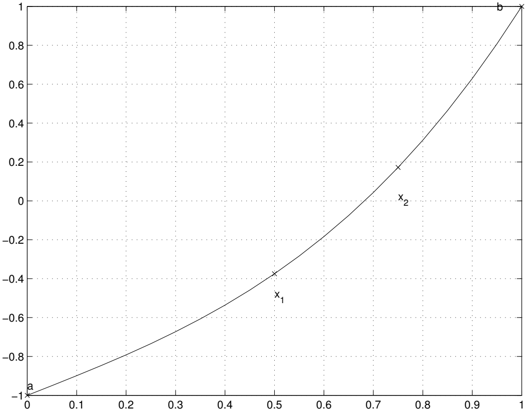
Graphe de la fonction $f(x) = x^3 + x - 1$



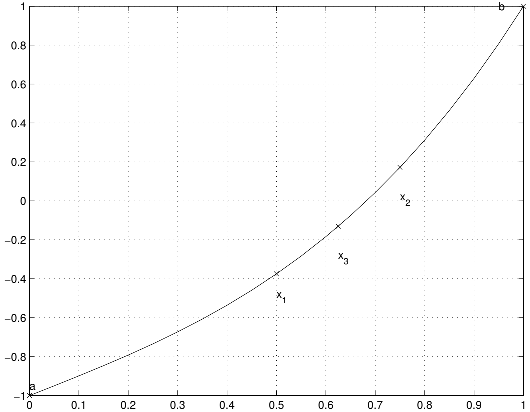
Graphe de la fonction $f(x) = x^3 + x - 1$



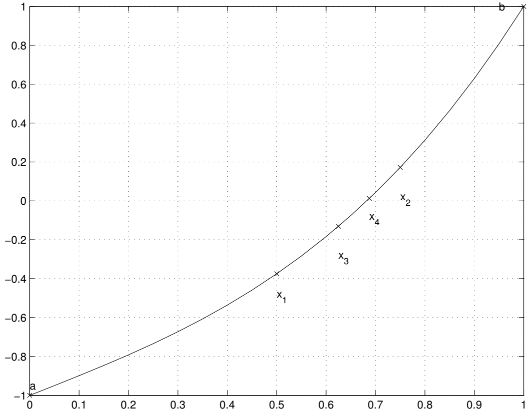
Graphe de la fonction $f(x) = x^3 + x - 1$



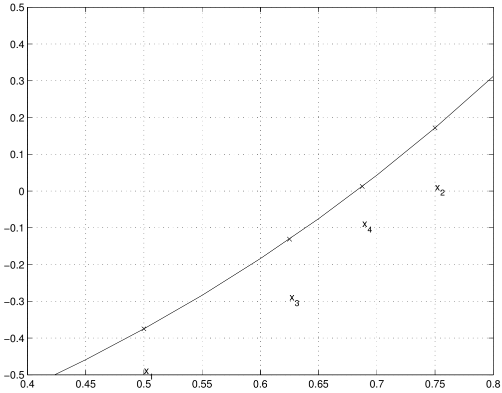
Graphe de la fonction $f(x) = x^3 + x - 1$



Graphe de la fonction $f(x) = x^3 + x - 1$



Graphe de la fonction $f(x) = x^3 + x - 1$



Résolution numérique d'une équation non linéaire

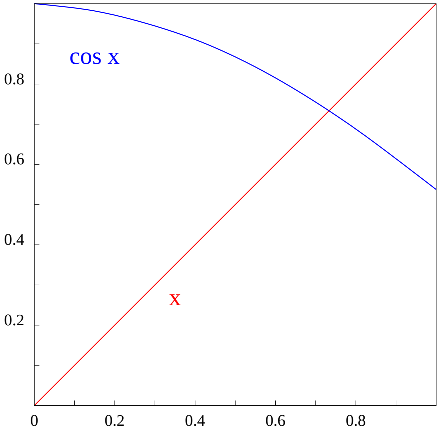
Cherchons une racine à l'équation

$$x - \cos x = 0.$$

Résolution numérique d'une équation non linéaire

Cherchons une racine à l'équation

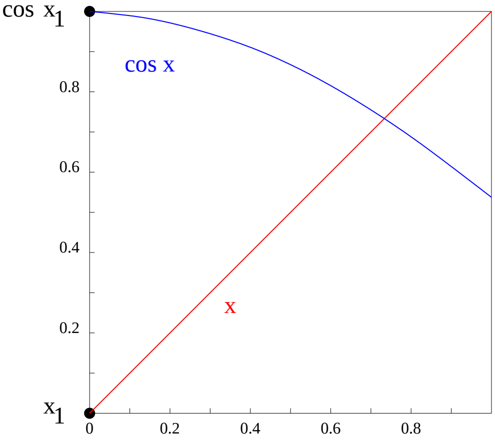
$$x - \cos x = 0.$$



Résolution numérique d'une équation non linéaire

Cherchons une racine à l'équation

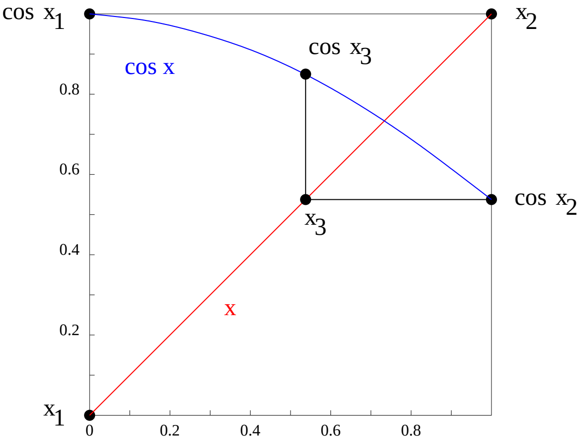
$$x - \cos x = 0.$$



Résolution numérique d'une équation non linéaire

Cherchons une racine à l'équation

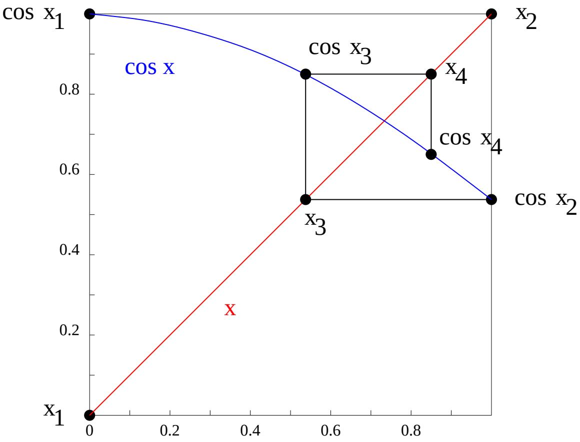
$$x - \cos x = 0.$$



Résolution numérique d'une équation non linéaire

Cherchons une racine à l'équation

$$x - \cos x = 0.$$



Résolution numérique d'une équation non linéaire

Nous résolvons l'équation

$$x = \cos x.$$

Itération	x	$\cos x$
1	0	1
2	1	0.5403
3	0.5403	0.8576
4	0.8576	0.6543
5	0.6543	0.7935
6	0.7935	0.7014
7	0.7014	0.7640
8	0.7640	0.7221

Remarque : converge lentement dans ce cas-ci.
Dans certains cas, la méthode ne fonctionne pas du tout.

Résolution numérique d'une équation non linéaire

Nous résolvons l'équation

$$x = \cos x.$$

Itération	x	$\cos x$
1	0	1
2	1	0.5403
3	0.5403	0.8576
4	0.8576	0.6543
5	0.6543	0.7935
6	0.7935	0.7014
7	0.7014	0.7640
8	0.7640	0.7221

Remarque : converge lentement dans ce cas-ci.
Dans certains cas, la méthode ne fonctionne pas du tout.

Résolution numérique d'une équation non linéaire

Nous résolvons l'équation

$$x = \cos x.$$

Itération	x	$\cos x$
1	0	1
2	1	0.5403
3	0.5403	0.8576
4	0.8576	0.6543
5	0.6543	0.7935
6	0.7935	0.7014
7	0.7014	0.7640
8	0.7640	0.7221

Remarque : converge lentement dans ce cas-ci.
Dans certains cas, la méthode ne fonctionne pas du tout.

Analyse du comportement des méthodes

Complexité d'un algorithme

Calcul du nombre d'opérations à effectuer pour arriver au résultat
Exemple : multiplication de deux polynômes de même degré

$$p(x) = p_n x^n + p_{n-1}x^{n-1} + \cdots + p_1 x + p_0$$
$$q(x) = q_n x^n + q_{n-1}x^{n-1} + \cdots + q_1 x + q_0.$$

On a

$$p(x)q(x) = p_n q_n x^{2n} + \cdots + (p_n q_i + p_{n-1} q_{i+1} + \cdots + p_i q_n)x^{n+i} + \cdots + p_0 q_0$$

On aura

Degré	$*$	$+$	Total	Degré	$*$	$+$	Total
$2n$	1	0	1	0	1	0	1
$2n-1$	2	1	3	1	2	1	3
	$\vdots$				$\vdots$		
$n+1$	n	$n-1$	$2n-1$	$n-1$	n	$n-1$	$2n-1$
n	$n+1$	n	$2n+1$				

Nombre d'opérations : $2\sum_{i=1}^{n}(2i-1) + (2n+1) = 2n^2 + 2n + 1$.
On dit $\mathcal{O}(n^2)$.

Analyse du comportement des méthodes

Complexité d'un algorithme

Calcul du nombre d'opérations à effectuer pour arriver au résultat
Exemple : multiplication de deux polynômes de même degré

$$p(x) = p_n x^n + p_{n-1}x^{n-1} + \cdots + p_1 x + p_0$$
$$q(x) = q_n x^n + q_{n-1}x^{n-1} + \cdots + q_1 x + q_0.$$

On a

$$p(x)q(x) = p_n q_n x^{2n} + \cdots + (p_n q_i + p_{n-1}q_{i+1} + \cdots + p_i q_n)x^{n+i} + \cdots + p_0 q_0$$

On aura

Degré	$*$	$+$	Total	Degré	$*$	$+$	Total
$2n$	1	0	1	0	1	0	1
$2n-1$	2	1	3	1	2	1	3
	$\vdots$				$\vdots$		
$n+1$	n	$n-1$	$2n-1$	$n-1$	n	$n-1$	$2n-1$
n	$n+1$	n	$2n+1$				

Nombre d'opérations : $2\sum_{i=1}^{n}(2i-1) + (2n+1) = 2n^2 + 2n + 1$.
On dit $\mathcal{O}(n^2)$.

Complexité d'un algorithme

Calcul du nombre d'opérations à effectuer pour arriver au résultat
Exemple : multiplication de deux polynômes de même degré

$$p(x) = p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0$$
$$q(x) = q_n x^n + q_{n-1} x^{n-1} + \cdots + q_1 x + q_0.$$

On a

$$p(x)q(x) = p_n q_n x^{2n} + \cdots + (p_n q_i + p_{n-1} q_{i+1} + \cdots + p_i q_n) x^{n+i} + \cdots + p_0 q_0$$

On aura

Degré	$*$	$+$	Total	Degré	$*$	$+$	Total
$2n$	1	0	1	0	1	0	1
$2n-1$	2	1	3	1	2	1	3
		$\vdots$				$\vdots$	
$n+1$	n	$n-1$	$2n-1$	$n-1$	n	$n-1$	$2n-1$
n	$n+1$	n	$2n+1$				

Nombre d'opérations : $2\sum_{i=1}^{n}(2i-1) + (2n+1) = 2n^2 + 2n + 1$.
On dit $\mathcal{O}(n^2)$.

Analyse du comportement des méthodes

Complexité d'un algorithme

Calcul du nombre d'opérations à effectuer pour arriver au résultat
Exemple : multiplication de deux polynômes de même degré

$$p(x) = p_n x^n + p_{n-1}x^{n-1} + \cdots + p_1 x + p_0$$
$$q(x) = q_n x^n + q_{n-1}x^{n-1} + \cdots + q_1 x + q_0.$$

On a

$$p(x)q(x) = p_n q_n x^{2n} + \cdots + (p_n q_i + p_{n-1} q_{i+1} + \cdots + p_i q_n)x^{n+i} + \cdots + p_0 q_0$$

On aura

Degré	$*$	$+$	Total	Degré	$*$	$+$	Total
$2n$	1	0	1	0	1	0	1
$2n-1$	2	1	3	1	2	1	3
		$\vdots$			$\vdots$		
$n+1$	n	$n-1$	$2n-1$	$n-1$	n	$n-1$	$2n-1$
n	$n+1$	n	$2n+1$				

Nombre d'opérations : $2\sum_{i=1}^{n}(2i-1) + (2n+1) = 2n^2 + 2n + 1$.
On dit $\mathcal{O}(n^2)$.

Analyse du comportement des méthodes

Complexité d'un algorithme

Calcul du nombre d'opérations à effectuer pour arriver au résultat
Exemple : multiplication de deux polynômes de même degré

$$
\begin{aligned}
p(x) &= p_n x^n + p_{n-1} x^{n-1} + \cdots + p_1 x + p_0 \\
q(x) &= q_n x^n + q_{n-1} x^{n-1} + \cdots + q_1 x + q_0.
\end{aligned}
$$

On a

$$p(x)q(x) = p_n q_n x^{2n} + \cdots + (p_n q_i + p_{n-1} q_{i+1} + \cdots + p_i q_n) x^{n+i} + \cdots + p_0 q_0$$

On aura

Degré	$*$	$+$	Total	Degré	$*$	$+$	Total
$2n$	1	0	1	0	1	0	1
$2n-1$	2	1	3	1	2	1	3
		$\vdots$				$\vdots$	
$n+1$	n	$n-1$	$2n-1$	$n-1$	n	$n-1$	$2n-1$
n	$n+1$	n	$2n+1$				

Nombre d'opérations : $2\sum_{i=1}^{n}(2i-1) + (2n+1) = 2n^2 + 2n + 1$.
On dit $\mathcal{O}(n^2)$.

Analyse du comportement des méthodes

Ordre de convergence

Nombre d'itérations nécessaires pour arriver à une précision désirée.
Vitesse à laquelle l'itéré se rapproche de la solution désirée.
Exemple : Cas de la dichotomie où $x_n \to x$

- On part d'un intervalle $[a, b]$
- A l'itération i, la taille de l'intervalle est $\frac{b-a}{2^i}$
- Si on veut s'assurer que $|x_n - x| \leq \epsilon$, on a

$$|x_n - x| \leq \frac{1}{2}\frac{b-a}{2^n} \leq \epsilon$$

c'est-à-dire $2^n \geq \frac{b-a}{2\epsilon}$ ou

$$n \geq \frac{\ln\left(\frac{b-a}{2\epsilon}\right)}{\ln 2}.$$

- Lorsque $y_n \to y$ avec $|y_n - y| \leq c_0 c^{pn}$, pour $c_0, p > 0$ et $0 < p < 1$, on parle de convergence d'ordre p.
- La dichotomie a un ordre de convergence d'ordre 1 ou linéaire.

Analyse du comportement des méthodes

Ordre de convergence

Nombre d'itérations nécessaires pour arriver à une précision désirée.
Vitesse à laquelle l'itéré se rapproche de la solution désirée.
Exemple : Cas de la dichotomie où $x_n \to x$

- On part d'un intervalle $[a, b]$
- A l'itération i, la taille de l'intervalle est $\frac{b-a}{2^i}$
- Si on veut s'assurer que $|x_n - x| \leq \epsilon$, on a

$$|x_n - x| \leq \frac{1}{2}\frac{b-a}{2^n} \leq \epsilon$$

  c'est-à-dire $2^n \geq \frac{b-a}{2\epsilon}$ ou

$$n \geq \frac{\ln\left(\frac{b-a}{2\epsilon}\right)}{\ln 2}.$$

- Lorsque $y_n \to y$ avec $|y_n - y| \leq c_0 c^{pn}$,
  pour $c_0, p > 0$ et $0 < p < 1$, on parle de convergence d'ordre p.
- La dichotomie a un ordre de convergence d'ordre 1 ou linéaire.

Analyse du comportement des méthodes

Ordre de convergence

Nombre d'itérations nécessaires pour arriver à une précision désirée.
Vitesse à laquelle l'itéré se rapproche de la solution désirée.
Exemple : Cas de la dichotomie où $x_n \to x$

- On part d'un intervalle $[a, b]$
- A l'itération i, la taille de l'intervalle est $\frac{b-a}{2^i}$
- Si on veut s'assurer que $|x_n - x| \leq \epsilon$, on a

$$|x_n - x| \leq \frac{1}{2}\frac{b-a}{2^n} \leq \epsilon$$

c'est-à-dire $2^n \geq \frac{b-a}{2\epsilon}$ ou

$$n \geq \frac{\ln\left(\frac{b-a}{2\epsilon}\right)}{\ln 2}.$$

- Lorsque $y_n \to y$ avec $|y_n - y| \leq c_0 c^{pn}$,
  pour $c_0, p > 0$ et $0 < p < 1$, on parle de convergence d'ordre p.
- La dichotomie a un ordre de convergence d'ordre 1 ou linéaire.

Analyse du comportement des méthodes

Ordre de convergence

Nombre d'itérations nécessaires pour arriver à une précision désirée.
Vitesse à laquelle l'itéré se rapproche de la solution désirée.
Exemple : Cas de la dichotomie où $x_n \to x$

- On part d'un intervalle $[a, b]$
- A l'itération i, la taille de l'intervalle est $\frac{b-a}{2^i}$
- Si on veut s'assurer que $|x_n - x| \leq \epsilon$, on a

$$|x_n - x| \leq \frac{1}{2}\frac{b-a}{2^n} \leq \epsilon$$

  c'est-à-dire $2^n \geq \frac{b-a}{2\epsilon}$ ou

$$n \geq \frac{\ln\left(\frac{b-a}{2\epsilon}\right)}{\ln 2}.$$

- Lorsque $y_n \to y$ avec $|y_n - y| \leq c_0 c^{pn}$,
  pour $c_0, p > 0$ et $0 < p < 1$, on parle de convergence d'ordre p.
- La dichotomie a un ordre de convergence d'ordre 1 ou linéaire.

Analyse du comportement des méthodes

Ordre de convergence

Nombre d'itérations nécessaires pour arriver à une précision désirée.
Vitesse à laquelle l'itéré se rapproche de la solution désirée.
Exemple : Cas de la dichotomie où $x_n \to x$

- On part d'un intervalle $[a, b]$
- A l'itération i, la taille de l'intervalle est $\frac{b-a}{2^i}$
- Si on veut s'assurer que $|x_n - x| \leq \epsilon$, on a

$$|x_n - x| \leq \frac{1}{2}\frac{b-a}{2^n} \leq \epsilon$$

  c'est-à-dire $2^n \geq \frac{b-a}{2\epsilon}$ ou

$$n \geq \frac{\ln\left(\frac{b-a}{2\epsilon}\right)}{\ln 2}.$$

- Lorsque $y_n \to y$ avec $|y_n - y| \leq c_0 c^{pn}$,
  pour $c_0, p > 0$ et $0 < p < 1$, on parle de convergence d'ordre p.
- La dichotomie a un ordre de convergence d'ordre 1 ou linéaire.

Analyse du comportement des méthodes

Ordre de convergence

Nombre d'itérations nécessaires pour arriver à une précision désirée.
Vitesse à laquelle l'itéré se rapproche de la solution désirée.
Exemple : Cas de la dichotomie où $x_n \rightarrow x$

- On part d'un intervalle $[a, b]$
- A l'itération i, la taille de l'intervalle est $\frac{b-a}{2^i}$
- Si on veut s'assurer que $|x_n - x| \leq \epsilon$, on a
$$|x_n - x| \leq \frac{1}{2}\frac{b-a}{2^n} \leq \epsilon$$
c'est-à-dire $2^n \geq \frac{b-a}{2\epsilon}$ ou
$$n \geq \frac{\ln\left(\frac{b-a}{2\epsilon}\right)}{\ln 2}.$$
- Lorsque $y_n \rightarrow y$ avec $|y_n - y| \leq c_0 c^{pn}$,
pour $c_0, p > 0$ et $0 < p < 1$, on parle de convergence d'ordre p.
- La dichotomie a un ordre de convergence d'ordre 1 ou linéaire.

Analyse du comportement des méthodes

Sensibilité aux erreurs des données

Certains problèmes sont mal conditionnés.

Une petite perturbation des données entraîne une grosse variation de la solution.

Exemple :

Considérons le système

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Solution : $x = (2 \quad -2)^T$.

Petite erreur dans la matrice :

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.144 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Nouvelle solution $x = (0.6663 \; 0.0002)^T$

Analyse du comportement des méthodes

Sensibilité aux erreurs des données

Certains problèmes sont mal conditionnés.
Une petite perturbation des données entraîne une grosse variation de la solution.
Exemple :
Considérons le système

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Solution : $x = (2 \quad -2)^T$.
Petite erreur dans la matrice :

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.144 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Nouvelle solution $x = (0.6663 \; 0.0002)^T$

Analyse du comportement des méthodes

Sensibilité aux erreurs des données

Certains problèmes sont mal conditionnés.
Une petite perturbation des données entraîne une grosse variation de la solution.
Exemple :
Considérons le système

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Solution : $x = (2 \quad -2)^T$.

Petite erreur dans la matrice :

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.144 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Nouvelle solution $x = (0.6663 \; 0.0002)^T$

Analyse du comportement des méthodes

Sensibilité aux erreurs des données

Certains problèmes sont mal conditionnés.
Une petite perturbation des données entraîne une grosse variation de la solution.
Exemple :
Considérons le système

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Solution :$x = (2 \;\; -2)^T$.
Petite erreur dans la matrice :

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.144 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Nouvelle solution $x = (0.6663 \; 0.0002)^T$

Analyse du comportement des méthodes

Sensibilité aux erreurs des données

Certains problèmes sont mal conditionnés.
Une petite perturbation des données entraîne une grosse variation de la solution.
Exemple :
Considérons le système

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.1441 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Solution :$x = (2 \;\; -2)^T$.
Petite erreur dans la matrice :

$$\begin{pmatrix} 1.2969 & 0.8648 \\ 0.2161 & 0.144 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.8642 \\ 0.1440 \end{pmatrix}.$$

Nouvelle solution $x = (0.6663 \; 0.0002)^T$

Analyse du comportement des méthodes

Les erreurs d'arrondi

L'ordinateur calcule en précision finie.
Cela implique de devoir arrondir les nombres pour les stocker dans la mémoire.
Parfois anodin mais peut mener à des comportements aberrants.
Exemple :
On souhaite approximer la dérivée de

$$f(x) = x^4$$

en $x = 1$. Si les mathématiques sont correctes, on devrait avoir

$$f'(x) = 4x^3, \qquad f'(1) = 4.$$

On calcule

$$f'(1) \approx \frac{f(1+h) - f(1)}{h}$$

pour h suffisamment petit.

Analyse du comportement des méthodes

Les erreurs d'arrondi

L'ordinateur calcule en précision finie.
Cela implique de devoir arrondir les nombres pour les stocker dans la mémoire.
Parfois anodin mais peut mener à des comportements aberrants.
Exemple :
On souhaite approximer la dérivée de

$$f(x) = x^4$$

en $x = 1$. Si les mathématiques sont correctes, on devrait avoir

$$f'(x) = 4x^3, \qquad f'(1) = 4.$$

On calcule

$$f'(1) \approx \frac{f(1+h) - f(1)}{h}$$

pour h suffisamment petit.

Analyse du comportement des méthodes

Les erreurs d'arrondi

L'ordinateur calcule en précision finie.
Cela implique de devoir arrondir les nombres pour les stocker dans la mémoire.
Parfois anodin mais peut mener à des comportements aberrants.
Exemple :
On souhaite approximer la dérivée de

$$f(x) = x^4$$

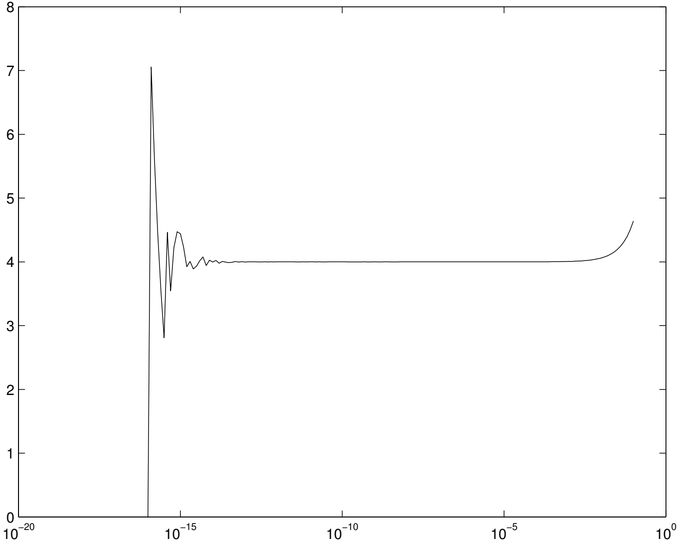
en $x = 1$. Si les mathématiques sont correctes, on devrait avoir

$$f'(x) = 4x^3, \qquad f'(1) = 4.$$

On calcule

$$f'(1) \approx \frac{f(1+h) - f(1)}{h}$$

pour h suffisamment petit.



$\frac{f(1+h)-f(1)}{h}$ en fonction de h

Chapitre 2

Représentation des nombres et erreurs

Séries de Taylor

Théorème

Soit f, une fonction possédant ses $(n+1)$ premières dérivées continues sur un intervalle fermé $[a,b]$, alors pour chaque $c, x \in [a,b]$, f peut s'écrire comme

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(c)}{k!}(x-c)^k + E_{n+1}$$

où le terme d'erreur E_{n+1} peut s'écrire sous la forme

$$E_{n+1} = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-c)^{n+1},$$

et ξ est un point situé entre c et x.

Notation $\mathcal{O}$

On dit que le terme d'erreur est *d'ordre* $n+1$.
$E_{n+1} = \mathcal{O}(h^{n+1})$, où h représente $x - c$.

Définition

Soit $f : \mathbb{R} \mapsto \mathbb{R}$. On dit que $f = \mathcal{O}(x^p)$ au voisinage de 0 si il existe $C \in \mathbb{R}_+$ et $x_0 \in \mathbb{R}_+$ tels que

$$|f(x)| \leq C|x^p| \quad \text{pour tout } -x_0 \leq x \leq x_0.$$

La notation $\mathcal{O}(x^p)$ permet de décrire le fait que la fonction tend au moins aussi vite vers 0 que x^p lorsque x tend vers 0.

Erreur absolue et relative

Définition

a - valeur réelle d'une variable

$\tilde{a}$ - valeur approchée

On définit deux types d'erreur

(i) l'*erreur absolue* : la quantité $\tilde{a} - a$,

(ii) l'*erreur relative* : la quantité $\frac{\tilde{a}-a}{a}$.

Chiffres significatifs et décimales correctes

$a = 9182.1234$ et son approximation $\tilde{a} = 9182.1111$

$\tilde{a}$ a

- 1 décimale correcte
- 5 chiffres significatifs

Erreur absolue et relative

Définition

a - valeur réelle d'une variable
$\tilde{a}$ - valeur approchée
On définit deux types d'erreur

(i) l'*erreur absolue* : la quantité $\tilde{a} - a$,
(ii) l'*erreur relative* : la quantité $\frac{\tilde{a}-a}{a}$.

Chiffres significatifs et décimales correctes

$a = 9182.1234$ et son approximation $\tilde{a} = 9182.1111$
$\tilde{a}$ a
- 1 décimale correcte
- 5 chiffres significatifs

Erreur absolue et relative

Définition

a - valeur réelle d'une variable

$\tilde{a}$ - valeur approchée

On définit deux types d'erreur

(i) l'*erreur absolue* : la quantité $\tilde{a} - a$,

(ii) l'*erreur relative* : la quantité $\frac{\tilde{a}-a}{a}$.

Chiffres significatifs et décimales correctes

$a = 9182.1234$ et son approximation $\tilde{a} = 9182.1111$

ã a

- 1 décimale correcte
- 5 chiffres significatifs

Erreur absolue et relative

Définition

a - valeur réelle d'une variable

$\tilde{a}$ - valeur approchée

On définit deux types d'erreur

(i) l'*erreur absolue* : la quantité $\tilde{a} - a$,

(ii) l'*erreur relative* : la quantité $\frac{\tilde{a}-a}{a}$.

Chiffres significatifs et décimales correctes

$a = 9182.1234$ et son approximation $\tilde{a} = 9182.1111$

$\tilde{a}$ a

- 1 décimale correcte
- 5 chiffres significatifs

Les entiers

Un entier (pas trop grand ou petit) est représenté de manière exacte.
C'est un nombre binaire (de nos jours 32 bits dont 1 pour le signe).
$\longrightarrow$ tous les entiers de -2147483647 à 2147483648.
Sur cet intervalle, les opérations de $\mathbb{Z} \times \mathbb{Z} \mapsto \mathbb{Z}$ sont exactes.

Les nombres réels

Représentés en virgule flottante

$$a = m\ 10^q, \qquad \text{avec } \frac{1}{10} \leq |m| < 1 \text{ et } q \in \mathbb{Z}.$$

m = la mantisse
q = l'exposant

En réalité, on travaille en binaire : $a = m\ 2^q$.
Double précision : 52 bits pour la mantisse, 1 bit pour le signe et 11 bits pour l'exposant.

Représentation des nombres dans un ordinateur

Les entiers

Un entier (pas trop grand ou petit) est représenté de manière exacte.
C'est un nombre binaire (de nos jours 32 bits dont 1 pour le signe).
$\longrightarrow$ tous les entiers de -2147483647 à 2147483648.
Sur cet intervalle, les opérations de $\mathbb{Z} \times \mathbb{Z} \mapsto \mathbb{Z}$ sont exactes.

Les nombres réels

Représentés en virgule flottante

$$a = m\ 10^q, \qquad \text{avec } \frac{1}{10} \leq |m| < 1 \text{ et } q \in \mathbb{Z}.$$

$m =$ la mantisse
$q =$ l'exposant

En réalité, on travaille en binaire : $a = m\ 2^q$.
Double précision : 52 bits pour la mantisse, 1 bit pour le signe et 11 bits pour l'exposant.

Les entiers

Un entier (pas trop grand ou petit) est représenté de manière exacte.
C'est un nombre binaire (de nos jours 32 bits dont 1 pour le signe).
$\longrightarrow$ tous les entiers de -2147483647 à 2147483648.
Sur cet intervalle, les opérations de $\mathbb{Z} \times \mathbb{Z} \mapsto \mathbb{Z}$ sont exactes.

Les nombres réels

Représentés en virgule flottante

$$a = m\ 10^q, \qquad \text{avec } \frac{1}{10} \leq |m| < 1 \text{ et } q \in \mathbb{Z}.$$

m = la mantisse
q = l'exposant

En réalité, on travaille en binaire : $a = m\ 2^q$.
Double précision : 52 bits pour la mantisse, 1 bit pour le signe et 11 bits pour l'exposant.

Représentation des nombres dans un ordinateur

Signe	Mantisse	Exposant
0	0 1 1 1 0 1 ... 1 0 1 0	1 0 0 1 0 0 0

Il existe la notion de nombres consécutifs en représentation machine.
La différence de mantisse entre les deux s'appelle epsilon machine.

Définition alternative : le plus petit $\epsilon > 0$ tel que $1 + \epsilon \neq 1$.

Représentation des nombres dans un ordinateur

Signe	Mantisse	Exposant
0	0 1 1 1 0 1 ... 1 0 1 0	1 0 0 1 0 0 0

Il existe la notion de nombres consécutifs en représentation machine.
La différence de mantisse entre les deux s'appelle epsilon machine.

Définition alternative : le plus petit $\epsilon > 0$ tel que $1 + \epsilon \neq 1$.

Représentation des nombres dans un ordinateur

Signe	Mantisse	Exposant
0	0 1 1 1 0 1 ... 1 0 1 0	1 0 0 1 0 0 0

Il existe la notion de nombres consécutifs en représentation machine.
La différence de mantisse entre les deux s'appelle epsilon machine.

Définition alternative : le plus petit $\epsilon > 0$ tel que $1 + \epsilon \neq 1$.

Calcul en virgule flottante

Supposons que l'on travaille en base 10 avec 4 chiffres après la virgule pour la mantisse.

$$0.4378 + 10.2345$$

$$0.4378\ 10^0 + 0.1023\ 10^2$$

0.0	0	4	3	7	8	10^2
0.1	0	2	3	0	0	10^2
0.1	0	6	6	7	8	10^2

Résultat :$0.1067\ 10^2$

2 expériences relatives à la représentation des nombres et à l'epsilon machine

A cause de l'arrondi, on considère

Entre x et sa représentation machine $x(1+\delta)$ où $|\delta| \leq \epsilon$

Entre $x+y$ et sa représentation machine $(x+y)(1+\delta)$ où $|\delta| \leq \epsilon$

Calcul en virgule flottante

Supposons que l'on travaille en base 10 avec 4 chiffres après la virgule pour la mantisse.

$$0.4378 + 10.2345$$

$$0.4378\ 10^0 + 0.1023\ 10^2$$

0.0	0	4	3	7	8	10^2
0.1	0	2	3	0	0	10^2
0.1	0	6	6	7	8	10^2

Résultat : $0.1067\ 10^2$

2 expériences relatives à la représentation des nombres et à l'epsilon machine

A cause de l'arrondi, on considère

Entre x et sa représentation machine $x(1+\delta)$ où $|\delta| \leq \epsilon$

Entre $x+y$ et sa représentation machine $(x+y)(1+\delta)$ où $|\delta| \leq \epsilon$

Calcul en virgule flottante

Supposons que l'on travaille en base 10 avec 4 chiffres après la virgule pour la mantisse.

$$0.4378 + 10.2345$$

$$0.4378\ 10^0 + 0.1023\ 10^2$$

0.0	0	4	3	7	8	10^2
0.1	0	2	3	0	0	10^2
0.1	0	6	6	7	8	10^2

Résultat : $0.1067\ 10^2$

2 expériences relatives à la représentation des nombres et à l'epsilon machine

A cause de l'arrondi, on considère

Entre x et sa représentation machine $x(1+\delta)$ où $|\delta| \leq \epsilon$

Entre $x+y$ et sa représentation machine $(x+y)(1+\delta)$ où $|\delta| \leq \epsilon$

Calcul en virgule flottante

Supposons que l'on travaille en base 10 avec 4 chiffres après la virgule pour la mantisse.

$$0.4378 + 10.2345$$

$$0.4378\ 10^0 + 0.1023\ 10^2$$

0.0	0	4	3	7	8	10^2
0.1	0	2	3	0	0	10^2
0.1	0	6	6	7	8	10^2
Résultat :0.1067 10^2						

2 expériences relatives à la représentation des nombres et à l'epsilon machine

A cause de l'arrondi, on considère

Entre x et sa représentation machine $x(1+\delta)$ où $|\delta| \leq \epsilon$

Entre $x+y$ et sa représentation machine $(x+y)(1+\delta)$ où $|\delta| \leq \epsilon$