

Université Batna 2.
Faculté des Mathématiques et de l'Informatique.
Département d'Informatique.
Année universitaire 2020-2021.

Cours de Méthodes Numériques

Mme OTSMANE Sarah

NB : Ce cours concerne les étudiants de la deuxième année licence LMD

Plan du cours

Chapitre 1 : Généralités sur l'analyse numérique et le calcul scientifique.

Chapitre 2 : Méthodes directes de résolution des systèmes linéaires.

Chapitre 3 : Méthodes itératives de résolution des systèmes linéaires.

Chapitre 4 : Calcul de valeurs et de vecteurs propres.

Connaissances préalables recommandées : Mathématiques de base.

Mode d'évaluation : Examen (60%), contrôle continu (40%).

Pour en savoir plus : (Livre)

- ✓ Mathématique sur ordinateur, François COTTET-EMARD et Pierre GOETGHELUCK.
- ✓ Algorithmique numérique, C. BREZINSKI.
- ✓ Initiation à l'analyse numérique, P. LASCAUX.
- ✓ Méthodes numériques appliquées, A. GOURDIN, M. BOUMAHART.
- ✓ Introduction à l'analyse numérique, Jacques BARANGER.

Introduction

L'analyse numérique est une branche des mathématiques. Beaucoup de problèmes ne sont pas résolubles par les méthodes analytiques connues, c'est à cause de cela que sont apparues les méthodes numériques.

En analyse numérique, et pour un problème posé (\mathbf{P}), on étudie toutes les méthodes de résolution de (\mathbf{P}), au moyen du calcul arithmétique. L'étude peut englober aussi bien les conditions d'existence et d'unicité de la solution du problème (\mathbf{P}), et aussi les performances et l'efficacité du procédé choisi (précision, convergence, stabilité, rapidité . . . ect).

Dans plusieurs cas l'approchement de la solution exacte dépend du nombre d'opérations à répéter, ce qui impose une difficulté contre l'application de ces méthodes numériques.

L'apparition de l'ordinateur et l'extension de l'informatique a rendu l'application des méthodes numériques très aisée du fait de l'élaboration d'algorithmes implémentés dans des machines à processeurs puissants.

Mais faire beaucoup d'opérations ne veut pas dire faire n'importe quoi : les méthodes ont un coût (nombre d'opérations arithmétiques élémentaires), lié d'une part au temps de calcul, et d'autre part à la capacité de mémoire nécessaire pour stocker les données et les résultats.

Aujourd'hui la technologie ne cesse d'avancer proposant du nouveau continuellement dans différents domaines, la recherche scientifique est allée plus loin, ayant pu comprendre et modéliser les mécanismes de phénomènes physiques qui étaient ambigus il y a quelques années. Si cela est rendu possible c'est grâce à l'analyse numérique.

Chapitre 01 : Généralités sur l'analyse numérique et le calcul scientifique.

1. Motivations	5
2. Arithmétique en virgule flottante et erreurs d'arrondis	5
2.1. Représentation des nombres en machine	6
2.1.1. Exigences pour une représentation des réels en machine	6
2.1.2. Représentation des réels par les nombres à virgule flottante	7
2.2. Arrondi et l'erreur d'arrondi	8

Chapitre 01 : Généralités sur l'analyse numérique et le calcul scientifique.

1. Motivations :

Un algorithme est un énoncé décrivant, à l'aide d'opérations élémentaires, toutes les étapes d'une démarche systématique permettant la résolution d'un problème spécifique. Un algorithme peut à son tour contenir des sous-algorithmes et doit pouvoir s'achever après un nombre fini d'opérations élémentaires afin de pouvoir être utilisé dans un programme informatique. La mise en œuvre d'un algorithme consiste en l'écriture de la série d'opérations élémentaires le composant dans un langage de programmation, ce que l'on appelle aussi fréquemment une implémentation.

La complexité d'un algorithme est une mesure de son temps d'exécution. Calculer la complexité d'un algorithme fait donc partie de l'analyse de l'efficacité et du coût d'une méthode numérique.

2. Arithmétique en virgule flottante et erreurs d'arrondis :

De nos jours, de nombreux calculs sont effectués à l'aide d'ordinateurs. Nombreux sont les gens qu'effectuent des calculs sur l'ordinateur sans attention particulière.

Pourtant, même le plus simple des programmes peut avoir un comportement surprenant, comme le montre l'exemple suivant, saisi dans l'interpréteur « **MATLAB** ».

```
>> x=0.1
x =
    0.1000
>> x+0.2==0.3
ans =
    0
```

Cette inégalité un peu inattendue est causée par les arrondis effectués par la machine, qui rendent le comportement difficilement prédictible. Ce comportement n'est d'ailleurs pas propre à **Matlab** ; un programme plus complexe, peu importe le langage, regorge de cas similaires !

Les erreurs d'arrondi sont inhérentes au calcul sur ordinateur, et donc inévitables. Ainsi, il est important de connaître leur origine pour les identifier et les maîtriser.

Ce chapitre est une introduction à l'arithmétique des nombres flottants. Vous y apprendrez comment les nombres à virgule flottante représentent les nombres réels en machine, en quoi ils en sont une approximation, et enfin quelles sont les conséquences en termes de calcul.

2.1. Représentation des nombres en machine :

Cette partie explique tout d'abord les principales exigences que doit remplir une représentation des réels. Elle introduit ensuite le type de représentation des réels le plus classique, dit à virgule flottante. Elle présente enfin le format le plus courant de nombres à virgule flottante, le format binary64.

2.1.1. Exigences pour une représentation des réels en machine :

Il existe des solutions pour représenter les réels de manière exacte en machine. Cependant, les contraintes de performance restreignent l'usage de ces méthodes au calcul formel. Les autres usages utilisent ainsi systématiquement des représentations approximatives des réels, qui sont les seules à remplir les exigences principales pour le calcul numérique, à savoir **l'amplitude**, **la précision**, et **la performance**.

Amplitude : L'amplitude d'une représentation correspond aux ordres de grandeur des nombres représentables. Il est nécessaire de pouvoir représenter à la fois des nombres très grands et très petits pour que les scientifiques s'y retrouvent. En mathématique, il est facile de générer des nombres encore plus extrêmes. Vous connaissez peut-être l'extrait de poème suivant :

Je fais souvent ce rêve étrange et pénétrant
D'une femme inconnue, et que j'aime, et qui m'aime,

« Mon rêve familial », Paul Verlaine

La probabilité de taper du premier coup ce texte de 105 caractères, espaces compris, en frappant aléatoirement les touches d'une machine à écrire, est de 45^{-105} soit environ 10^{-174} . Si l'on considère le nombre de textes de 105 caractères que ce procédé peut générer, on trouve l'ordre de grandeur inverse, c'est-à-dire 10^{-174} .

Précision : La précision d'une représentation est encore plus importante que son amplitude. La précision correspond au nombre de chiffres significatifs qui peuvent être stockés. En physique, les constantes connues avec le plus de précision ne dépassent pas la vingtaine de chiffres significatifs. Il est donc important que la représentation des réels utilisée garde la même précision quel que soit l'ordre de grandeur.

Performance : La performance correspond à la rapidité avec laquelle les calculs peuvent s'effectuer. Il est en effet hors de question d'attendre l'éternité pour obtenir un résultat. Il s'agit aussi d'être performant sur toutes les machines couramment utilisées.

Il existe différentes familles de représentation des réels qui remplissent les critères d'amplitude, précision et performance, mais la plus utilisée est sans conteste la représentation à virgule flottante.

2.1.2. Représentation des réels par les nombres à virgule flottante :

De nombreux domaines de l'informatique nécessitent une manipulation efficace des nombres réels. Or l'ensemble des nombres réels est à la fois infini et continu.

Une représentation complète de l'ensemble des nombres réels en machine est donc impossible. Pour approximer les nombres réels, plusieurs approches ont été utilisées, comme la représentation en virgule fixe, les nombres rationnels... . La représentation en virgule flottante reste la plus utilisée pour approximer les nombres réels.

Définition :

Les nombres à virgule flottante souvent appelés nombres flottants, ou encore flottants sont une représentation d'un sous-ensemble fini des nombres réels. Ils sont utilisés dans les programmes informatiques pour approcher des valeurs de type réel.

Ils sont représentés par le triplet [Goldberg 1991] : s : signe = (0 ou 1),

m : mantisse et e : exposant. La valeur d'un flottant est donnée par :

$$(-1)^s m \times \beta^e ,$$

Où β est la base de calcul (2 dans le cas le plus courant).

L'**exposant** est un entier signé délimité par $e_{min} \leq e \leq e_{max}$.

La **mantisse** est représentée par un nombre fixé de chiffres.

$$d_0, d_1, \dots, d_{p-1}. \text{ Avec } 0 \leq d_i \leq \beta.$$

Où p est le nombre de chiffres qui composent la mantisse. Notons qu'il existe une relation directe entre p et la précision du calcul.

Le bit de poids fort d_0 est un bit implicite dont la valeur se déduit de celle de l'exposant e .

2.2. Arrondi et l'erreur d'arrondi :

La plupart du temps, le résultat d'une opération sur flottants n'est pas un nombre flottant. Aussi afin de n'avoir que des nombres flottants à manipuler, ce résultat doit être arrondi vers l'un des deux flottants les plus proches de résultat réel, en fonction du mode choisi.

L'opération d'arrondi introduit donc une erreur d'arrondi. L'erreur introduite dans les calculs sur les nombres flottants peut être mesurée de deux façons différentes : erreur absolue et erreur relative.

On note par o la fonction d'arrondi : $\hat{x} = o(x)$.

Où x est un nombre de type réel et \hat{x} son approximation dans les nombres flottants.

L'erreur absolue : L'erreur absolue (Δ) est la distance absolue entre le résultat flottant et la valeur réelle. Cette mesure peut être utilisée pour estimer l'erreur d'arrondi d'une formule lorsqu'on connaît une majoration pour les valeurs des variables.

$$\Delta x = |x - o(x)|.$$

L'erreur relative : l'erreur relative (ϵ) est le rapport entre l'erreur absolue et la valeur réelle.

$$\epsilon(x) = \frac{\Delta(x)}{|x|} = \left| \frac{x - o(x)}{x} \right|.$$

L'erreur relative est utilisée pour estimer l'erreur indépendamment de l'échelle de grandeur de la variable.

Par exemple, si nous souhaitons une longueur en centimètres et que cette longueur est supérieure ou inférieure de moins 0.5 cm à une valeur $o(x)$ en centimètres, on a : $\Delta x < 0.5$

Ce qui se traduit par intervalle pour x (valeur réelle de la longueur) :

$$\Delta x = |x - o(x)| < 0.5$$

Soit : $o(x) - 0.5 < x < o(x) + 0.5$.

Ou encore : x est dans l'intervalle $[o(x) - 0.5, o(x) + 0.5]$

Exercice 01 : Pour mesurer l'épaisseur d'un cylindre creux, vous mesurez le diamètre D_1 et le diamètre extérieur D_2 et vous trouvez $D_1 = 19.5 \pm 0.1$ mm et $D_2 = 26.7 \pm 0.1$ mm . Donnez le résultat de mesure et sa précision.

Réponse : 3.6 ± 0.1 mm ; 3%.

Erreurs numériques :

L'homme fournit à l'ordinateur, en général, les nombres sous forme décimale.

A l'intérieur de l'ordinateur, ces nombres sont transformés et manipulés sous forme binaire (avec les erreurs que cela comporte). Une fois le calcul terminé, ces nombres sont reconvertis en base de 10 ce qui introduit une nouvelle erreur. L'affichage à l'écran ne fournit donc pas nécessairement la valeur exacte contenue en mémoire.

Comment expliquez-vous les résultats suivants ?

Opération	Résultat affiché
$\frac{1}{3} + \frac{1}{3} + \frac{1}{3}$	1
$1 - \frac{1}{3} - \frac{1}{3} - \frac{1}{3}$	$x \neq 0$

La première opération donne un résultat apparemment exact. Si ce résultat était rigoureusement exact on devrait avoir 0 comme résultat de la seconde opération. Le résultat de la première opération n'est pas 1 (elle vaut soit $1 + \varepsilon$, soit $1 - \varepsilon$ selon la technique d'arrondissement utilisé par l'ordinateur), mais après l'avoir converti en base 10, cette valeur est arrondi à 1. Ceci n'est pas le cas dans la seconde opération.