Méthodes et Analyse Numériques

Eric Goncalvès da Silva

INSTITUT POLYTECHNIQUE DE GRENOBLE

METHODES, ANALYSE ET CALCULS NUMERIQUES

Eric Goncalvès - septembre 2005

Table des matières

Chapitre I

MODELISATION, DISCRETISATION ET SIMULATION NUMERIQUE

I.1 Qu'est-ce qu'un modèle ?

Le principe d'un modèle est de remplacer un système complexe en un objet ou opérateur simple reproduisant les aspects ou comportements principaux de l'original (ex : modèle réduit, maquette, modèle mathématique ou numérique, modèle de pensée ou raisonnement).

I.2 Pourquoi faut-il modéliser ?

Dans la nature, les systèmes et phénomènes physiques les plus intéressants sont aussi les plus complexes à étudier. Ils sont souvent régis par un grand nombre de paramètres non-linéaires interagissant entre eux (la météorologie, la turbulence des fluides...).

I.3 Quels sont les différents modèles ?

L'une des solutions est de recourir à une série d'expériences pour analyser les paramètres et grandeurs du système. Mais les essais peuvent s'avérer très coûteux (essais en vol, essais avec matériaux rares, instrumentations très chères...) et ils peuvent être très dangereux (essais nucléaires, environnement spatial...). Enfin il peut être difficile de mesurer tous les paramètres : échelles du problème trop petites (chimie du vivant, couche limite en fluide...) ou trop grandes (astrophysique, météorologie, géophysique...).

On peut aussi construire un modèle mathématique permettant la représentation du phénomène physique. Ces modèles utilisent très souvent des systèmes d'équations aux dérivées partielles (EDP) non-linéaires dont on ne connait pas de solutions analytiques en général. Il faut alors résoudre le problème numériquement en transformant les équations continues de la physique en un problème discret sur un certain domaine de calcul (le maillage). Dans certains cas il s'agit de la seule alternative (nucléaire, astrophysique, spatial...). Dans d'autres cas, les simulations numériques sont ménées en parallèle avec des expérimentations.

I.4 De la modélisation à la simulation numérique

Les différentes étapes pour modéliser un système complexe :

- Recherche d'un modèle mathématique réprésentant la physique. Mise en équation.
- Elaboration d'un maillage. Discrétisation des équations de la physique.
- Résolution des équations discrètes (souvent systèmes linéaires à résoudre).
- Transcription informatique et programmation des relations discrètes.
- Simulation numérique et exploitation des résultats.

L'ingénieur peut être amené à intervenir sur l'une ou plusieurs de ces différentes étapes.

I.5 Aspect fini des ordinateurs

La solution exacte d'un problème d'EDO ou d'EDP est une fonction continue. Les ordinateurs ne connaissent que le fini et le discret. En effectuant un calcul numérique, un ordinateur ne peut retenir qu'un nombre fini de chiffres pour représenter les opérandes et les résultats des calculs intermédiaires. Les solutions approchées seront calculées comme des ensembles de valeurs discrètes sous la forme de composantes d'un vecteur solution d'un problème matriciel. La représentation des nombres dans un ordinateur introduit la notion **d'erreur d'arrondi** ou de **troncature**. Ces erreurs peuvent se cumuler sur un calcul et la solution numérique finale pourra s'avérer très éloignée de la solution exacte.

Exemple d'erreur d'arrondi : considérons un ordinateur utilisant 4 chiffres pour représenter un nombre. Calculons la somme 1.348+9.999. Le résultat exact est 11.347 et comporte 5 chiffres. Le calculateur va le preprésenter de manière approchée : 11.35. Il commet une erreur d'arrondi ègale à (11.35-11.347)=0.003.

I.5.1 Représentation des entiers

Les entiers sont représentés par une suite de bits organisés en octets. Par exemple un entier codé sur 2 octets occupera 16 bits ($2^{16} = 65536$) et pourra réprensenter un entier compris entre -32768 et 32767. On parle d'entier simple précision.
Le type entier codé sur 4 octets ($2^{32} = 4294967296$) permet la représentation des entiers compris entre -2 147 483 648 et 2 147 483 647. On parle d'entier double précision.

Les opérations sur les entiers, dans la mesure où le résultat est un entier représentable par la machine, s'effectuent exactement.

I.5.2 Représentation des réels ou nombres flottants

Un nombre flottant s'écrit sous la forme $X = a.b^n$ où a est la mantisse, b la base et n l'exposant. Par exemple, la représentation de π avec 10 caractères est : $+0.314159\,10^{+1}$. Les 10 caractères sont répartis selon : 1 pour le signe, 6 pour la mantisse, 3 pour l'exposant dont 1 pour son signe.

La représentation standard des réels choisie par les principaux constructeurs d'ordinateur est sous forme de nombre flottants où $b = 2$ et a, n sont deux nombres binaires.

Un réel en simple précision occupe 4 octets (32 bits). Son exposant est stocké sur un octet (il prend toutes les valeurs entières entre -128 et +127), son signe sur un bit et sa mantisse occupe les 23 bits restants représentée par $t = 23$ caractères binaires $d_1, d_2, ..., d_t$ avec $d_1 = 1$. Un réel X correspond au nombre suivant :

$$X = \frac{d_1}{2} + \frac{d_2}{2^2} + \frac{d_3}{2^3} + ... + \frac{d_t}{2^{23}}$$

Le plus nombre en valeur absolue ou zéro machine est : $2^{-129} \simeq 1.47\,10^{-39}$
Le plus grand nombre en valeur absolue ou infini machine est : $(1 - 2^{-23})\,2^{127} \simeq 1.7\,10^{38}$

La meilleure précision possible pour des calculs sur des nombres de l'ordre de l'unité sera : $2^{-23} \simeq 1.19\,10^{-7}$
Pour des nombres de l'ordre de 1000, la meilleure précision tombe à : $2^{-23}2^{10} \simeq 1.22\,10^{-4}$

Un réel en double précision occupe 8 octets soit 64 bits : 1 bit de signe, 11 bits pour l'exposant et 52 bits pour la mantisse.

I.6 Notion de stabilité

On distingue trois types de stabilité

- La stabilité d'un problème physique.
- La stabilité d'un problème mathématique.
- La stabilité numérique d'une méthode de calcul.

I.6.1 Stabilité d'un problème physique : système chaotique

Un problème est dit *chaotique* si une petite variation des données initiales entraîne une variation totalement imprévisible des résultats. Cette notion de chaos, liée à la physique d'un problème, est indépendante du modèle mathématique utilisé et encore plus de la méthode numérique utilisée pour résoudre ce problème mathématique. De nombreux problèmes sont chaotiques, par exemple la turbulence des fluides.

I.6.2 Stabilité d'un problème mathématique : sensibilité

Un problème est dit *très sensible* ou *mal conditionné* si une petite variation des données ou des paramètres entraîne une grande variation des résultats. Cette notion de conditionnement, liée au problème mathématique, est indépendante de la méthode numérique utilisée pour le résoudre. Pour modéliser un problème physique qui n'est pas chaotique, on construira un modèle mathématique qui sera le mieux conditionné possible.

I.6.3 Stabilité d'une méthode numérique

Une méthode est dite instable si elle est sujette à une propagation importante des erreurs numériques de discrétisation et d'arrondi.
Un problème peut être bien conditionné alors que la méthode numérique choisie pour le résoudre est instable. Dans ce cas, il est impératif de changer de méthode numérique. Par contre, si le problème de départ est mal conditionné, aucune méthode numérique ne pourra y remédier. Il faudra alors essayer de trouver une formulation mathématique différente du même problème, si on sait que le problème physique sous-jacent est stable.

I.7 Un peu d'histoire...

I.7.1 Avant les ordinateurs : les calculateurs

Le mot calcul vient du latin *calculus*, qui signifie "petite pierre". Les romains, comme beaucoup de peuples antiques, utilisaient couramment de petites pierres pour éviter de mémoriser les termes d'une addition. Cette pratique se perfectionna et donna naissance à la machine à calculer la plus ancienne connue : **le boulier**, ou abaque, qui fut d'une utilisation presque universelle jusqu'à tout récemment.

Des machines mécaniques furent mises au point au XVIIeme siècle. La plus connue est la **pascaline**, construite par Blaise Pascal à l'âge de 19 ans pour soulager son père, collecteur d'impôts, du fardeau des calculs répétitifs. La machine, à base de roues dentées, ne pouvait qu'additionner et soustraire. Leibniz transforma la pascaline en une machine capable de multiplier. Il a fallu attendre le milieu du XIXeme siècle avant qu'une machine, construite par le Français C. Thomas de Colmar, fonctionne véritablement et connaisse un succès commercial.

Le concept de machine programmable fut conçue sur le papier par l'Anglais Charles Babbage, basée sur la lecture sur des cartes perforées des instructions de calcul et des données à traiter. Signalons que les cartes perforées furent popularisées dans le contexte des métiers à tisser par Joseph-Marie Jacquard. La machine analytique de Babbage inspira les constructeurs de machines à calculer du début du XXeme siècle.

Vers 1890, l'Américain Herman Hollerith construira une machine à cartes perforées destinée à compiler les résultats du recensement des Etats-Unis. En 1896, Hollerith fonde sa compagnie, la *Tabulating Machines Corporation* qui deviendra en 1924 l'*International Business Machines* (IBM).

La nécessité d'effectuer des calculs scientifiques motivera la conception et la construction de machines dédiées à ces activités. L'Américain Vannevar Bush construira, dans les années 1930, un calculateur mécanique **analogique**. Ce calculateur simulait par un dispositif mécanique l'intégration d'une équation différentielle. Ce type de machine sera utilisé pendant la deuxième guerre mondiale pour les besoins de la ballistique.

Les besoins des militaires lors de la deuxième guerre mondiale stimulera la conception et la construction de calculateurs encore plus puissants. Aux Etats-Unis, l'armée par l'intermédiaire de l'Université de Pennsylvanie va mettre au point le plus puissant calculateur jamais construit : l'ENIAC (Electronic Numerator, Integrator, Analyser and Computer). Il ne fut terminé que trois mois après la fin de la guerre. Il comptait une multitude de lampes électroniques qui devaient être remplacées souvent. Les lampes étaient susceptibles d'être rendues inopérantes quand un moustique (bug) s'y écrasait, ce qui est à l'origine de l'expression courante pour désigner les erreurs de programmation. L'ENIAC n'est pas un ordinateur mais une calculatrice géante, cadencée à 200kHz.

I.7.2 Les ordinateurs

Le célèbre mathématicien John von Neumann est à l'origine de l'architecture logique des machines pour automatiser les calculateurs. En juin 1945, il écrit un rapport dans lequel il décrit l'architecture d'une future machine qui inspirera les premiers ordinateurs. L'essentiel de l'architecture proposée par von Neumann consiste à confier la gestion du calcul à une **unité de contrôle** (Central Processing Unit ou CPU). L'unité de contrôle gère les instructions d'un programme et coordonne les autres unités de l'appareil : mémoire, entrée/sortie et unité de calcul. Les instructions sont exécutées de manière séquentielle. L'architecture de base des ordinateurs est toujours la même que celle imaginée par von Neumann.

Von Neumann fut inspiré dans ses travaux par ceux d'un jeune mathématicien anglais, Alan Turing. En 1936, Turing précisa la notion d'**algorithme** et imagina une machine automatique, la machine de Turing, qui pouvait résoudre n'importe quel problème : c'était une machine universelle. Elle fonctionnait à partir d'opérations logiques élémentaires et écrivait, copiait ou lisait de l'information dans un registre.

Turing fut impliqué dans la construction du tout premier ordinateur, construit à Manchester de 1946 à 1948 et surnommé Manchester MARK1. Cet ordinateur fut un exemplaire unique. Il était réservé à des applications militaires (armements nucléaires). Le premier ordinateur civil fur le UNIVAC1 (UNIVersal Automatic Computer), créé et commercialisé en 1951 par Eckert et Mauchly. IBM livrera son modèle 701 aux militaires en 1951 et commercialisera son modèle 650 en 1953. A Los Alamos, la machine MANIAC (Mathematical And Numerical Integrator And Computer) sera opérationnelle en 1952.

On distingue généralement cing générations d'ordinateurs qui diffèrent essentiellement (sauf la cinquième) par les moyens techniques utilisés :

- La première génération (1948-1955) est caractérisée par l'utilisation de lampes électroniques et de tambours magnétiques pour la mémoire. Le langage machine utilisé pour leur programmation n'est pas universel et est conçu sur mesure pour une application précise.

- La deuxième génération (1956-1963) est caractérisée par le remplacement des lampes par des transistors ; la mémoire y est souvent constituée de noyaux magnétiques. Le langage machine a fait place à l'assembleur.

- La troisième génération (1964-1971) remplace un assemblage de transistors individuels par des circuits intégrés (dispositifs à semiconducteurs dans lesquels sont intégrés des éléments de type résistances, transistors, condensateurs...). Cette génération est aussi caractérisée par l'utilisation d'un système d'opération, un programme central qui coordonne l'exécution de plusieurs autres programmes.

- La quatrième génération est caractérisée par l'emploi des microprocesseurs (unités de contrôle, de traitement et de mémoire rassemblées sur une même puce de silicium). Le premier microprocesseur fut commercialisé par Intel en 1971. L'utilisation de microprocesseurs fabriqués à une échelle industrielle permettra la commercialisation de petits ordinateurs et même d'ordinateurs personnels à la fin des années 1970.

- La cinquième génération est difficile à définir ! Ce terme est désigné à tort et à travers pour diverses innovations réalisées.

La révolution informatique fut rendue possible par les progrès inouïs de l'éctronique. Le point de départ de cette révolution technologique est l'invention du **transistor**.

La compréhension du comportement des semiconducteurs (substance cristalline comme le germanium ou silicium, dont les propriétés de conduction électrique sont intermédiaires entre un métal et un isolant) date des annés 1930. Les laboratoires Bell mettront au point le premier transistor en décembre 1947 ce qui vaudra à ses auteurs Shockley, Bardeen, Brattain le prix Nobel de physique en 1956. En 1959, le jeune ingénieur Jack Kilby de la firme Texas Instrument construit le premier **circuit intégré**. De nombreuses compagnies s'établiront à Palo alto en Californie et constitueront la Silicon Valley. L'une de ses entreprises, fondée en 1965 par Gordon Moore et Bob Noyce, choisit le nom d'Intel (pour INTegrated ELectronics) et produit en 1971 le premier "ordinateur sur une puce" ou microprocesseur, le Intel 4004 avec 2300 transistors. En 1978, elle lance le Intel 8086 qui compte 29000 transistors et est cadencé à 4,77 MHz. Toute une série de processeurs suivent : 286 (en 1982), 386 (1985), 486 (1989), Pentium (1993), Pentium II (1997), Pentium III (1999), Pentium IV (2001)...

La progression constante de la puissance des ordinateurs associée à l'abaissement considérable des coûts a ouvert la possibilité de réaliser des simulations numériques sur des ordinateurs personnels. Même si les super-ordinateurs restent nécessaires pour des simulations très importantes, il devient possible de faire exécuter des simulations numériques sur des PC bon marché. L'unité de mesure pour évaluer les performances d'un ordinateur est le GFlops (Giga FLoating OPeration per Second ou milliard d'opérations en virgule flottante par seconde). Un PC actuel de type Pentium IV cadencé à 2.4 Ghz peut délivrer une puissance d'environ 2Gflops.

I.7.3 Petite chronologie

Voici une chronologie sommaire du développement de l'informatique :

1936 Publication par Alan Turing des principes généraux des machines automatiques suivant un algorihme.

1945 Proposition de von Neumann pour l'architecture des calculateurs automatiques.
Inauguration de l'ENIAC le dernier grand calculateur avant l'ordinateur.

1947 Invention du transistor par Bardeen, Brattain et Shockley aux laboratoires Bell.

1948 Le Manchester MARK1, premier ordinateur construit sur le plan de von Neumann.
Le mathématicien américain Claude Shannon publie sa théorie mathématique des communications et introduit l'acronyme bit (BInary digiT) pour désigner le plus petit élément d'information.

1950 Invention de l'assembleur, langage de programmation de bas niveau.

1951 Le UNIVAC 1, premier ordinateur commercial.

1952 Premier ordinateur produit par IBM : le modèle 701.

1954 Lancement du modèle 704 d'IBM, doté d'une mémoire de 144ko.

1955 Premier réseau informatique : SABRE, créé pour American Airlines.
W. Shockley quitte Bell pour fonder sa propre compagnie à Palo Alto, en Californie, la première de ce qui deviendra la Silicon Valley.

1956 Le premier ordinateur à transistors : le TRADIC de Bell, marque le début de la deuxième génération.

1957 Création par John Backus du premier langage de programmation supérieur : le FORTRAN.

1958 Premier circuit intégré, réalisé par Jack Kilby, de Texas Instrument.

1959 Le premier ordinateur interactif : le PDP 1 ; de la Digital Equipment Corporation.

1962 Production des premiers transistors à effet de champ commerciaux.

1965 Fondation de la compagnie Intel, dans la Silicon Valley.

1968 Lancement du réseau ARPANET, l'ancêtre d'INTERNET.
Invention de l'environnement fenêtres-souris.

1969 Début de la création du système d'opération UNIX.

1970 Première mémoire vive RAM à base de semiconducteurs : le Intel 1103 avec 1K de mémoire.

1971 Création du premier microprocesseur : le Intel 4004, qui compte 2300 transistors.

1973 Création du langage de programmation C, étroitement lié au système d'opération UNIX.

1976 Lancement du supercalculateur CRAY 1 (puissance de crête 100 MFlops).

1977 Premier ordinateur Apple.

1981 IBM se lance dans la commercialisation des ordinateurs personnels.

1983 Création du langage de programmation C++.

1984 Lancement du Macintosh de Apple, premier succès commercial d'un ordinateur à environnement fenêtre-souris.

1986 Lancement du système d'opération Windows 1.1 ; par Microsoft.

1989 Création du World Wide Web et du langage HTML, au Centre Européen de Recherche Nucléaire (CERN).

1994 Intel lance le Pentium, microprocesseur contenant plus de cinq millions de transistors. Les nouveautés : bus de données élargit à 64 bits pour l'accès à la mémoire, capacité du processeur à pouvoir traiter deux instructions par cycle d'horloge et deux niveaux de mémoire cache afin d'accélérer le traitement des instructions au niveau du processeur.

1998 Lancement du Pentium II d'Intel et du K6-2 d'AMD.

2001 Début du Pentium III d'Intel. Ce processeur monte la fréquence des PC à 866 MHz.

2003 Début du Pentium IV d'Intel. Lancé à 1 Ghz, il atteindra jusqu'à 3,8 Ghz.

2005 Le nombre de transistors sur une puce de PC atteint les 1,7 milliards.
Un microprocesseur de PC peut délivrer jusqu'à 6,4 GFlops.
Le supercalculateur le plus puissant est le DOE BlueGene d'IBM intallé au Lawrence Livermore National Laboratory (USA) avec une puissance maximale de 280,6 TFlops.
Le supercalculateur le plus puissant de France (62ème rang mondial) se trouve au CEA pour des applications militaires : le NovaScale Quadrics de Bull SA (5,8TFlops).

Loi de Moore

Devant l'évolution extrèmement rapide des technologies liées aux microprocesseurs, plusieurs personnes ont cherché à formuler des hypothèses sur le progrès de leurs performances. Ainsi Gordon Moore, cofondateur de la société Intel a affirmé en 1965 pour une conférence de presse, que "le nombre de transistors par cicuit de même taille va doubler tous les 18 mois". Cette affirmation a marqué les esprits, puisqu'elle est devenue un défi à tenir pour les fabriquants de microprocesseurs.

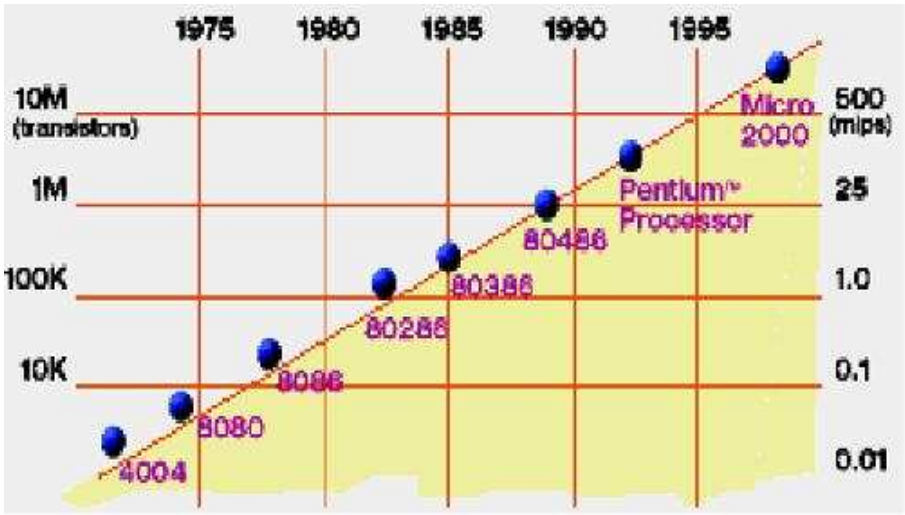

FIG. I.1 – Loi de Moore

Chapitre II

DISCRETISATION DES EDP

II.1 LES TROIS GRANDES FAMILLES DE METHODES

Pour passer d'une problème exact continu régit par une EDP au problème approché discret, il existe trois grandes familles de méthodes :

- **Les différences finies**.
  La méthode consiste à remplacer les dérivées partielles par des différences divisées ou combinaisons de valeurs ponctuelles de la fonction en un nombre fini de points discrets ou noeuds du maillage.
  Avantages : grande simplicité d'écriture et faible coût de calcul.
  Inconvénients : limitation à des géométries simples, difficultés de prise en compte des conditions aux limites de type Neumann.

- **Les volumes finis**.
  La méthode intègre, sur des volumes élémentaires de forme simple, les équations écrites sous forme de loi de conservation. Elle fournit ainsi de manière naturelle des approximations discrètes conservatives et est particulièrement bien adaptée aux équations de la mécanique des fluides. Sa mise en oeuvre est simple avec des volumes élémentaires rectangles.
  Avantages : permet de traiter des géométries complexes avec des volumes de forme quelconque, détermination plus naturelle des conditions aux limites de type Neumann.
  Inconvénient : peu de résultats théoriques de convergence.

- **Les éléments finis**.
  La méthode consiste à approcher, dans un sous-espace de dimension finie, un problème écrit sous forma variationnelle (comme minimisation de l'énergie en général) dans un espace de dimension infinie. La solution approchée est dans ce cas une fonction déterminée par un nombre fini de paramètres comme, par exemple, ses valeurs en certains points ou noeuds du maillage.
  Avantages : traitement possible de géométries complexes, nombreux résultats théoriques sur la convergence.
  Inconvénient : complexité de mise en oeuvre et grand coût en temps de calcul et mémoire.

II.2 LES DIFFERENCES FINIES

II.2.1 Principe - ordre de précision

La méthode des différences finies consiste à approximer les dérivées des équations de la physique au moyen des développements de Taylor et se déduit directement de la définition de la dérivée. Elle est due aux travaux de plusieurs mathématiciens du 18éme siècle (Euler, Taylor, Leibniz...).

Soit $u(x, y, z, t)$ une fonction de l'espace et du temps. Par définition de la dérivée, on a :

$$\frac{\partial u}{\partial x} = \lim_{\Delta x \to 0} \frac{u(x + \Delta x, y, z, t) - u(x, y, z, t)}{\Delta x}$$

Si Δx est petit, un développement de Taylor de $u(x + \Delta x, y, z, t)$ au voisinage de x donne :

$$u(x + \Delta x, y, z, t) = u(x, y, z, t) + \Delta x \frac{\partial u}{\partial x}(x, y, z, t) + \frac{\Delta x^2}{2} \frac{\partial^2 u}{\partial x^2}(x, y, z, t) + \frac{\Delta x^3}{6} \frac{\partial^3 u}{\partial x^3}(x, y, z, t) + ...$$

En **tronquant** la série au premier ordre en Δx, on obtient :

$$\frac{u(x + \Delta x, y, z, t) - u(x, y, z, t)}{\Delta x} = \frac{\partial u}{\partial x}(x, y, z, t) + \mathcal{O}(\Delta x)$$

L'approximation de la dérivée $\frac{\partial u}{\partial x}(x)$ est alors <u>d'ordre 1</u> indiquant que l'erreur de troncature $\mathcal{O}(\Delta x)$ tend vers zéro comme la puissance première de Δx.

<u>Définition</u> : la puissance de Δx avec laquelle l'erreur de troncature tend vers zéro est appelée **l'ordre de la méthode**.

II.2.2 Notation indicielle - cas 1D

Considérons un cas monodimensionnel où l'on souhaite déterminer une grandeur $u(x)$ sur l'intervalle [0,1]. La recherche d'une solution discrète de la grandeur u amène à constituer un maillage de l'intervalle de définition. On considère un maillage (ou grille de calcul) composé de $N + 1$ points x_i pour $i = 0, ..., N$ régulièrement espacés avec un pas Δx. Les points $x_i = i\Delta x$ sont appelés les noeuds du maillage.
Le problème continu de départ de détermination d'une grandeur sur un ensemble de dimension infinie se ramène ainsi à la recherche de N valeurs discrètes de cette grandeur aux différents noeuds du maillage.

<u>Notation</u> : on note u_i la valeur discrète de $u(x)$ au point x_i, soit $u_i = u(x_i)$. De même pour la dérivée de $u(x)$ au noeud x_i, on note $\left(\frac{\partial u}{\partial x}\right)_{x=x_i} = \left(\frac{\partial u}{\partial x}\right)_i = u'_i$. Cette notation s'utilise de façon équivalente pour toutes les dérivées d'ordre successif de la grandeur u.

Le schéma aux différences finies d'ordre 1 présenté au-dessus s'écrit, en notation indicielle :

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{u_{i+1} - u_i}{\Delta x} + \mathcal{O}(\Delta x)$$

Ce schéma est dit "avant" ou "décentré avant" ou upwind.

Il est possible de construire un autre schéma d'ordre 1, appelé "arrière" :

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{u_i - u_{i-1}}{\Delta x} + \mathcal{O}(\Delta x)$$

II.2.3 Schéma d'ordre supérieur

Des schémas aux différences finies d'ordre supérieur peuvent être construits en manipulant des développement de Taylor au voisinage de x_i. On écrit :

$$\begin{aligned} u_{i+1} &= u(x_i + \Delta x) = u_i + \Delta x \left(\frac{\partial u}{\partial x}\right)_i + \frac{\Delta x^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i + \mathcal{O}(\Delta x^3) \\ u_{i-1} &= u(x_i - \Delta x) = u_i - \Delta x \left(\frac{\partial u}{\partial x}\right)_i + \frac{\Delta x^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i + \mathcal{O}(\Delta x^3) \end{aligned}$$

La soustraction de ces deux relations donne : $u_{i+1} - u_{i-1} = 2\Delta x \left(\frac{\partial u}{\partial x}\right)_i + \mathcal{O}(\Delta x^3)$
Ce qui permet d'obtenir le schéma d'ordre deux dit "centré" pour approximer la dérivée première de u :

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{u_{i+1} - u_{i-1}}{2\Delta x} + \mathcal{O}(\Delta x^2)$$

Pour obtenir des ordres supérieurs, il faut utiliser plusieurs noeuds voisins de x_i . Le nombre de points nécessaire à l'écriture du schéma s'appelle le stencil. Par exemple, un schéma aux différences finies d'ordre 3 pour la dérivée première s'écrit :

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{-u_{i+2} + 6u_{i+1} - 3u_i - 2u_{i-1}}{6\Delta x} + \mathcal{O}(\Delta x^3)$$

II.2.4 Dérivée d'ordre supérieur

Le principe est identique et repose sur les développements de Taylor au voisinage de x_i. Par exemple pour construire un schéma d'approximation de la dérivée seconde de u, on écrit :

$$\begin{aligned} u_{i+1} &= u_i + \Delta x \left(\frac{\partial u}{\partial x}\right)_i + \frac{\Delta x^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i + \frac{\Delta x^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \mathcal{O}(\Delta x^4) \\ u_{i-1} &= u_i - \Delta x \left(\frac{\partial u}{\partial x}\right)_i + \frac{\Delta x^2}{2}\left(\frac{\partial^2 u}{\partial x^2}\right)_i - \frac{\Delta x^3}{6}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \mathcal{O}(\Delta x^4) \end{aligned}$$

En faisant la somme de ces deux égalités, on aboutit à : $u_{i+1} + u_{i-1} - 2u_i = \Delta x^2 \left(\frac{\partial^2 u}{\partial x^2}\right)_i + \mathcal{O}(\Delta x^4)$
Ce qui permet d'obtenir le schéma d'ordre deux dit "centré" pour approximer la dérivée seconde de u :

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} + \mathcal{O}(\Delta x^2)$$

Il existe aussi une formulation "avant" et "arrière" pour la dérivée seconde, toute deux d'ordre 1 :

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_i = \frac{u_{i+2} - 2u_{i+1} + u_{i-1}}{\Delta x^2} + \mathcal{O}(\Delta x) \qquad \left(\frac{\partial^2 u}{\partial x^2}\right)_i = \frac{u_i - 2u_{i-1} + u_{i-2}}{\Delta x^2} + \mathcal{O}(\Delta x)$$

Il est également possible de construire, par le même procédé, des schémas aux différences finies d'ordre supérieur pour les dérivées deuxième, troisième, etc...

II.2.5 Généralisation de la notation indicielle

Dans le cas 1D instationnaire, considérons l'évolution d'une grandeur $u(x,t)$ en fonction de l'espace et du temps. Le domaine de définition de u est décomposé en N noeuds x_i répartis régulièrement avec un pas d'espace Δx. De même, le temps est décomposé en intervalle élémentaire de pas constant Δt. On notera u_i^n la valeur discrète de la grandeur $u(x,t)$ au noeud x_i et au temps $n\Delta t$.

Dans le cas 2D, considérons une grandeur $u(x,y)$ définie sur un certain domaine. Ce dernier est décomposé en $N \times P$ noeuds (x_i, y_j) répartis régulièrement avec un pas d'espace Δx dans la direction x et Δy dans l'autre direction. On notera u_{ij} la valeur discrète de la grandeur $u(x,y)$ au noeud (x_i, y_j).

De façon similaire, dans le cas 2D instationnaire, on notera u_{ij}^n la valeur discrète de la grandeur $u(x,y,t)$ au noeud x_i, y_j et au temps $n\Delta t$. Et dans le cas 3D instationnaire, on notera u_{ijk}^n la valeur discrète de la grandeur $u(x,y,z,t)$ au noeud (x_i, y_j, z_k) et au temps $n\Delta t$.

II.2.6 Quelques schémas en 1D

Différences finies avant, ordre 1

	u_i	u_{i+1}	u_{i+2}	u_{i+3}	u_{i+4}
$\Delta x u_i'$	-1	1			
$\Delta x^2 u_i''$	1	-2	1		
$\Delta x^3 u_i'''$	-1	3	-3	1	
$\Delta x^4 u_i^{(4)}$	1	-4	6	-4	1

Différences finies arrière, ordre 1

	u_{i-4}	u_{i-3}	u_{i-2}	u_{i-1}	u_i
$\Delta x u_i'$				-1	1
$\Delta x^2 u_i''$			1	-2	1
$\Delta x^3 u_i'''$		-1	3	-3	1
$\Delta x^4 u_i^{(4)}$	1	-4	6	-4	1

Différences finies centré, ordre 2

	u_{i-2}	u_{i-1}	u_i	u_{i+1}	u_{i+2}
$2\Delta x u_i'$		-1		1	
$\Delta x^2 u_i''$		1	-2	1	
$2\Delta x^3 u_i'''$	-1	2	0	-2	1
$\Delta x^4 u_i^{(4)}$	1	-4	6	-4	1

Différences finies centré, ordre 4

	u_{i-3}	u_{i-2}	u_{i-1}	u_i	u_{i+1}	u_{i+2}	u_{i+3}
$12\Delta x u_i'$		1	-8	0	8	-1	
$12\Delta x^2 u_i''$		-1	16	-30	16	-1	
$8\Delta x^3 u_i'''$	-1	-8	13	0	-13	8	-1
$6\Delta x^4 u_i^{(4)}$	-1	12	-39	56	-39	12	-1

II.2.7 Dérivées croisées

Déterminons une approximation de la dérivée croisée $\dfrac{\partial^2 f}{\partial x \partial y}$ de la fonction de 2 variables $f(x,y)$. La discrétisation du domaine de calcul est bidimensionnelle et fait intervenir deux pas d'espace supposés constants Δx et Δy dans les directions x et y.

La principe est toujours basé sur les développements de Taylor :

$$\begin{aligned} f(x_{i+l}, y_{j+m}) &= f(x_i, y_j) + l\Delta x \left(\frac{\partial f}{\partial x}\right)_i + m\Delta y \left(\frac{\partial f}{\partial y}\right)_j + \frac{(l\Delta x)^2}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_i + \frac{(m\Delta y)^2}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_j \\ &+ \frac{2ml\Delta x\Delta y}{2}\left(\frac{\partial^2 f}{\partial x \partial y}\right)_{i,j} + ... \end{aligned}$$

Au voisinage du point (i, j) :

$$\begin{aligned} f_{i+1,j+1} &= f_{i,j} + \Delta x \left(\frac{\partial f}{\partial x}\right)_i + \Delta y \left(\frac{\partial f}{\partial y}\right)_j + \Delta x\Delta y \left(\frac{\partial^2 f}{\partial x \partial y}\right)_{i,j} + \frac{\Delta x^2}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_i + \frac{\Delta y^2}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_i \\ f_{i-1,j-1} &= f_{i,j} - \Delta x \left(\frac{\partial f}{\partial x}\right)_i - \Delta y \left(\frac{\partial f}{\partial y}\right)_j + \Delta x\Delta y \left(\frac{\partial^2 f}{\partial x \partial y}\right)_{i,j} + \frac{\Delta x^2}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_i + \frac{\Delta y^2}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_i \\ f_{i+1,j-1} &= f_{i,j} + \Delta x \left(\frac{\partial f}{\partial x}\right)_i - \Delta y \left(\frac{\partial f}{\partial y}\right)_j - \Delta x\Delta y \left(\frac{\partial^2 f}{\partial x \partial y}\right)_{i,j} + \frac{\Delta x^2}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_i + \frac{\Delta y^2}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_i \\ f_{i-1,j+1} &= f_{i,j} - \Delta x \left(\frac{\partial f}{\partial x}\right)_i + \Delta y \left(\frac{\partial f}{\partial y}\right)_j - \Delta x\Delta y \left(\frac{\partial^2 f}{\partial x \partial y}\right)_{i,j} + \frac{\Delta x^2}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_i + \frac{\Delta y^2}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_i \end{aligned}$$

En effectuant une combinaison linéaire des quatre équations précédentes ((1)+(2)-(3)-(4)), nous obtenons une approximation de la dérivée croisée à l'ordre 1 :

$$\left(\frac{\partial^2 f}{\partial x \partial y}\right)_{i,j} = \frac{f_{i+1,j+1} - f_{i+1,j-1} - f_{i-1,j+1} + f_{i-1,j-1}}{4\Delta x\Delta y}$$

II.2.8 Exemple simple 1D avec conditions de Dirichlet

Considérons l'équation différentielle suivante :

$$\begin{cases} -u''(x) = f(x) & , \quad x \in]0, 1[\\ u(0) = \alpha \quad \text{et} \quad u(1) = \beta & \end{cases}$$

où f est une fonction continue.
Le maillage est construit en introduisant $N + 1$ noeuds x_i avec $i = 0, 1, .., N$, régulièrement espacés avec un pas Δx. La quantité u_i désignera la valeur de la fonction $u(x)$ au noeud x_i.
L'équation à résoudre s'écrit, sous forme discrète en chaque noeud x_i :

$$-\left(\frac{d^2 u}{dx^2}\right)_i = f(x_i) = f_i$$

Approximons la dérivée seconde de u au moyen d'un schéma centré à l'ordre 2 :

$$\left(\frac{d^2 u}{dx^2}\right)_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}$$

L'équation discrétisée est ainsi :

$$\frac{2u_i - u_{i+1} - u_{i-1}}{\Delta x^2} = f_i \qquad \text{; pour } i \text{ variant de 1 à N-1}$$

Il est très pratique d'utiliser une formulation matricielle en faisant apparaître le vecteur des inconnues discrètes :

$$\frac{1}{\Delta x^2}\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} f_1 + \alpha/\Delta x^2 \\ f_2 \\ \vdots \\ f_{N-2} \\ f_{N-1} + \beta/\Delta x^2 \end{bmatrix}$$

II.2.9 Exemple simple 1D avec conditions mixtes Dirichlet-Neumann

Considérons l'équation différentielle suivante :

$$\begin{cases} -u''(x) = f(x) & , \qquad x \in]0,1[\\ u(0) = \alpha \quad \text{et} \quad u'(1) = \beta \end{cases}$$

où l'on a cette fois une condition de Neumann en $x = 1$.

Les modifications du problème discrétisé par rapport au cas précédent sont les suivantes. Tout d'abord, le nombre d'inconnues a changé. Il y a une inconnue au bord en $x = 1$. Le problème discret a donc maintenant, sur la base du même maillage que précédemment, N inconnues u_i pour i variant de 1 à N.
D'autre part, il faut discrétisée la condition de Neumann $u'(1) = \beta$. Plusieurs choix sont possibles pour approximer cette dérivée première. C'est un des inconvénients de la méthode des différences finies : elle ne donne pas de façon naturelle une bonne approximation des conditions de Neumann. Dans notre cas, utilisons une approximation d'ordre 1 : $u'(1) = \dfrac{u_N - u_{N-1}}{\Delta x}$
Sous forme matricielle, on obtient :

$$\frac{1}{\Delta x^2}\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} f_1 + \alpha/\Delta x^2 \\ f_2 \\ \vdots \\ f_{N-2} \\ f_{N-1} \\ \beta/\Delta x \end{bmatrix}$$

II.2.10 Discrétisation de l'équation de la chaleur 1D

Considérons le problème monodimensionnel de la conduction de la chaleur dans une barre de 1m de longueur. Le champ de température $T(x,t)$ vérifie l'équation de la chaleur :

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}$$

où α est la diffusivité thermique.
A cette EDP s'ajoute deux conditions aux limites aux extrémités de la barre $T(0,t) = T_g$ et $T(1,t) = T_d$ ainsi qu'une condition initale $T(x,0) = T_0$.

L'intervalle [0,1] est discrétisé en $N+1$ noeuds de coordonnées x_i (i variant de 0 à N) régulièrement espacés. Notons Δx le pas d'espace. Le temps est discrétisé en intervalles de pas constant Δt. Notons T_i^n la température au noeud $x_i = i\Delta x$ et à l'instant $t = n\Delta t$.

On peut utiliser deux approches pour discrétiser cette équation de la chaleur. La première dite **explicite** utilise une discrétisation au noeud x_i et à l'itération courante n :

$$\left(\frac{\partial T}{\partial t}\right)_i^n = \alpha \left(\frac{\partial^2 T}{\partial x^2}\right)_i^n$$

Et la seconde dite **implicite** utilise une discrétisation au noeud x_i et à l'itération $n+1$:

$$\left(\frac{\partial T}{\partial t}\right)_i^{n+1} = \alpha \left(\frac{\partial^2 T}{\partial x^2}\right)_i^{n+1}$$

II.2.10.1 Schéma explicite

Nous utilisons un schéma avant d'ordre 1 pour évaluer la dérivée temporelle et un schéma centré d'ordre 2 pour la dérivée seconde en espace :

$$\begin{aligned}
\left(\frac{\partial T}{\partial t}\right)_i^n &= \frac{T_i^{n+1} - T_i^n}{\Delta t} \\
\left(\frac{\partial^2 T}{\partial x^2}\right)_i^n &= \frac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{\Delta x^2}
\end{aligned}$$

En posant $\lambda = \alpha \dfrac{\Delta t}{\Delta x^2}$, la température à l'itération $n+1$ est donnée par :

$$T_i^{n+1} = \lambda T_{i-1}^n + (1-2\lambda) T_i^n + \lambda T_{i+1}^n \qquad i \text{ variant de 1 à N-1}$$

Soit sous forme matricielle :

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N-2} \\ T_{N-1} \end{bmatrix}^{n+1} = \begin{bmatrix} 1-2\lambda & \lambda & 0 & \cdots & 0 \\ \lambda & 1-2\lambda & \lambda & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \lambda & 1-2\lambda & \lambda \\ 0 & 0 & 0 & \lambda & 1-2\lambda \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N-2} \\ T_{N-1} \end{bmatrix}^n + \lambda \begin{bmatrix} T_g \\ 0 \\ \vdots \\ 0 \\ T_d \end{bmatrix}$$

II.2.10.2 Schéma implicite

Nous utilisons un schéma arrière d'ordre 1 pour évaluer la dérivée temporelle et un schéma centré d'ordre 2 pour la dérivée seconde en espace :

$$\left(\frac{\partial T}{\partial t}\right)_i^{n+1} = \frac{T_i^{n+1} - T_i^n}{\Delta t}$$

$$\left(\frac{\partial^2 T}{\partial x^2}\right)_i^{n+1} = \frac{T_{i+1}^{n+1} - 2T_i^{n+1} + T_{i-1}^{n+1}}{\Delta x^2}$$

En posant $\lambda = \alpha \dfrac{\Delta t}{\Delta x^2}$, la température à l'itération $n+1$ est donnée par :

$$(1+2\lambda)T_i^{n+1} - \lambda(T_{i+1}^{n+1} + T_{i-1}^{n+1}) = T_i^n \qquad i \text{ variant de 1 à N-1}$$

On constate que les inconnues à l'itération $n+1$ sont reliées entre elles par une relation implicite (d'où le nom de la méthode).

Sous forme matricielle :

$$\begin{bmatrix} 1+2\lambda & -\lambda & 0 & \cdots & 0 \\ -\lambda & 1+2\lambda & -\lambda & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & -\lambda & 1+2\lambda & -\lambda \\ 0 & 0 & 0 & -\lambda & 1+2\lambda \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N-2} \\ T_{N-1} \end{bmatrix}^{n+1} = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N-2} \\ T_{N-1} \end{bmatrix}^{n} + \lambda \begin{bmatrix} T_g \\ 0 \\ \vdots \\ 0 \\ T_d \end{bmatrix}$$

A chaque itération, le vecteur des inconnues discrètes se détermine par résolution d'un système linéaire. La matrice du système étant tridiagonale, un algorithme de Thomas (basé sur la méthode du pivot de Gauss) est très souvent utilisé.

Algorithme de Thomas

Cet algorithme est utilisé pour la résolution d'un système avec une matrice tridiagonale de dimension N faisant intervenir un vecteur d'inconnues discrètes X_i, de la forme :

$$\begin{array}{ccccccccl} & & b_1X_1 & + & c_1X_2 & = & d_1 & & i=1 \\ a_iX_{i-1} & + & b_iX_i & + & c_iX_{i+1} & = & d_i & & i \text{ variant de 2 à N-1} \\ a_NX_{N-1} & + & b_NX_N & & & = & d_N & & i=N \end{array}$$

Le calcul s'effectue en deux étapes (qui correspondent aux deux étapes du pivot de Gauss). La triangularisation fait apparaître les coefficients α_i et β_i évalués par récurrence :

$$\alpha_i = \frac{-a_i}{b_i + c_i\alpha_{i+1}} \qquad \text{et} \qquad \beta_i = \frac{d_i - c_i\beta_{i+1}}{b_i + c_i\alpha_{i+1}} \qquad \text{pour i variant de N à 1}$$

La deuxième étape détermine les inconnues selon la récurrence : $X_1 = \beta_1$ puis $X_i = \alpha_i X_{i-1} + \beta_i$ pour i variant de 2 à N.

II.2.11 Discrétisation de l'équation de la chaleur 2D stationnaire

Considérons le problème bidimensionnel stationnaire de la conduction de la chaleur dans un domaine rectangulaire $[0,L_x]\times[0,L_y]$. Le champ de température $T(x,y)$ vérifie l'équation de Laplace :

$$\begin{cases} \Delta T = \dfrac{\partial^2 T}{\partial x^2} + \dfrac{\partial^2 T}{\partial y^2} = 0 \quad , & (x,y) \in [0,L_x]\times[0,L_y] \\ T(0,y) = T_g \quad \text{et} \quad T(L_x,y) = T_d & 0 < y < L_y \\ T(x,0) = T_b \quad \text{et} \quad T(x,L_y) = T_h & 0 < x < L_x \end{cases}$$

Le domaine de calcul est discrétisé en (N+1)×(P+1) noeuds (x_i, y_j) (i variant de 0 à N et j variant de 0 à P). On supposera que les pas d'espace dans chaque direction Δx et Δy sont constants. La température discrète au noeud (x_i, y_j) sera notée $T_{ij} = T(x_i, y_j)$.

Nous utilisons un schéma centré d'ordre 2 pour approximer les dérivées secondes en espace :

$$\left(\frac{\partial^2 T}{\partial x^2}\right)_{ij} = \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2}$$

$$\left(\frac{\partial^2 T}{\partial y^2}\right)_{ij} = \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2}$$

La formulation discrétisée est alors, pour i variant de 1 à $N-1$ et j variant de 1 à $P-1$:

$$\Delta y^2\,(T_{i+1,j} + T_{i-1,j}) + \Delta x^2\,(T_{i,j+1} + T_{i,j-1}) - 2(\Delta x^2 + \Delta y^2)T_{i,j} = 0$$

Soit sous forme matricielle, pour N=P=4, en posant $A = \Delta x^2 + \Delta y^2$:

$$\begin{bmatrix} -2A & \Delta y^2 & 0 & \Delta x^2 & 0 & 0 & 0 & 0 & 0 \\ \Delta y^2 & -2A & \Delta y^2 & 0 & \Delta x^2 & 0 & 0 & 0 & 0 \\ 0 & \Delta y^2 & -2A & 0 & 0 & \Delta x^2 & 0 & 0 & 0 \\ \Delta x^2 & 0 & 0 & -2A & \Delta y^2 & 0 & \Delta x^2 & 0 & 0 \\ 0 & \Delta x^2 & 0 & \Delta y^2 & -2A & \Delta y^2 & 0 & \Delta x^2 & 0 \\ 0 & 0 & \Delta x^2 & 0 & \Delta y^2 & -2A & 0 & 0 & \Delta x^2 \\ 0 & 0 & 0 & \Delta x^2 & 0 & 0 & -2A & \Delta y^2 & 0 \\ 0 & 0 & 0 & 0 & \Delta x^2 & 0 & \Delta y^2 & -2A & \Delta y^2 \\ 0 & 0 & 0 & 0 & 0 & \Delta x^2 & 0 & \Delta y^2 & -2A \end{bmatrix} \begin{bmatrix} T_{11} \\ T_{21} \\ T_{31} \\ T_{12} \\ T_{22} \\ T_{32} \\ T_{13} \\ T_{23} \\ T_{33} \end{bmatrix} = - \begin{bmatrix} \Delta x^2 T_b + \Delta y^2 T_g \\ \Delta x^2 T_b \\ \Delta x^2 T_b + \Delta y^2 T_d \\ \Delta y^2 T_g \\ 0 \\ \Delta y^2 T_d \\ \Delta x^2 T_h + \Delta y^2 T_g \\ \Delta x^2 T_h \\ \Delta x^2 T_h + \Delta y^2 T_d \end{bmatrix}$$

Dans le cas où les pas d'espace sont identiques $\Delta x = \Delta y$, la formulation devient, pour i variant de 1 à $N-1$ et j variant de 1 à $P-1$:

$$T_{i+1,j} + T_{i,j-1} + T_{i-1,j} + T_{i,j+1} - 4T_{i,j} = 0$$

Soit sous forme matricielle, pour N=P=4 :

$$\begin{bmatrix} -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 \end{bmatrix} \begin{bmatrix} T_{11} \\ T_{21} \\ T_{31} \\ T_{12} \\ T_{22} \\ T_{32} \\ T_{13} \\ T_{23} \\ T_{33} \end{bmatrix} = - \begin{bmatrix} T_b + T_g \\ T_b \\ T_b + T_d \\ T_g \\ 0 \\ T_d \\ T_h + T_g \\ T_h \\ T_h + T_d \end{bmatrix}$$

Notons I la matrice identité d'ordre 3 et D la matrice de dimension 3 définie par :

$$D = \begin{bmatrix} -4 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & -4 \end{bmatrix}$$

Notons T_1, T_2 et T_3 les vecteurs à 3 composantes définis par :

$$T_1 = \begin{bmatrix} T_{11} \\ T_{21} \\ T_{31} \end{bmatrix} \qquad T_2 = \begin{bmatrix} T_{12} \\ T_{22} \\ T_{32} \end{bmatrix} \qquad T_3 = \begin{bmatrix} T_{13} \\ T_{23} \\ T_{33} \end{bmatrix}$$

Le système peut s'écrire sous la forme matricielle bloc suivante :

$$\begin{bmatrix} D & I & 0 \\ I & D & I \\ 0 & I & D \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} = - \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}$$

La matrice obtenue est tridiagonale et chacun de ses blocs est tridiagonal. La résolution du système peut s'effectuer par une méthode de Thomas matriciel où une méthode itérative matricielle (méthode de Gauss-Seidel).

Algorithme de Thomas matriciel

Cet algorithme est utilisé pour la résolution d'un système avec une matrice, de dimension N, tridiagonale par bloc, faisant intervenir un vecteur d'inconnues discrètes X_i, de la forme :

$$\begin{array}{ccccccccl} & & B_1X_1 & + & C_1X_2 & = & D_1 & & i = 1 \\ A_iX_{i-1} & + & B_iX_i & + & C_iX_{i+1} & = & D_i & & i \text{ variant de 2 à N-1} \\ A_NX_{N-1} & + & B_NX_N & & & = & D_N & & i = N \end{array}$$

øù A_i, B_i, C_i sont des matrices et D_i un vecteur.

On introduit la matrice α_i et le vecteur β_i évalués par les relations de récurrence suivantes :

$$\alpha_i \;=\; -\left(B_i + C_i\alpha_{i+1}\right)^{-1}\times A_i \qquad \text{et} \qquad \beta_i \;=\; \left(B_i + C_i\alpha_{i+1}\right)^{-1}\times\left(D_i - C_i\beta_{i+1}\right) \qquad \text{i variant de N à 1}$$

La deuxième étape détermine les inconnues selon la récurrence : $X_1 = \beta_1$ puis $X_i = \alpha_i X_{i-1} + \beta_i$ pour i variant de 2 à N.

<u>Remarque</u> : Avec une condition de Neumann sur l'un des bords du domaine, par exemple en $y = 0$ un flux de chaleur égale à ϕ_b, il faudrait ajouter à la formulation précédente la discrétisation de cette condition au bord.
Ceci a pour conséquence l'ajout de $N - 1$ inconnues supplémentaires à savoir les valeurs de la température au bord ($j = 0$ et i variant de 1 à $N - 1$).

Par exemple utilisons un schéma d'ordre 1 pour évaluer le flux de chaleur :

$$-\lambda\left(\frac{\partial T}{\partial y}\right)_{i,0} \;=\; \frac{T_{i,1} - T_{i,0}}{\Delta y} \;=\; \phi_b$$

Soit sous forme matricielle, dans le cas où $\Delta x = \Delta y$ et pour N=P=4 :

$$\begin{bmatrix}
-1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4
\end{bmatrix}
\begin{bmatrix}
T_{10} \\ T_{20} \\ T_{30} \\ T_{11} \\ T_{21} \\ T_{31} \\ T_{12} \\ T_{22} \\ T_{32} \\ T_{13} \\ T_{23} \\ T_{33}
\end{bmatrix}
= -
\begin{bmatrix}
\phi_b\,\Delta_x/\lambda \\ \phi_b\,\Delta_x/\lambda \\ \phi_b\,\Delta_x/\lambda \\ T_g \\ 0 \\ T_d \\ T_g \\ 0 \\ T_d \\ T_h + T_g \\ T_h \\ T_h + T_d
\end{bmatrix}$$

II.3 LES VOLUMES FINIS

II.3.1 Introduction

La méthode des Volumes Finis consiste à intégrer, sur des volumes élémentaires, les équations écrites sous forme intégrale. C'est une méthode particulièrement bien adaptée à la discrétisation spatiale **des lois de conservation**, contrairement aux Eléments Finis, et est ainsi très utilisée en mécanique des fluides.

Sa mise en oeuvre est simple si les volumes élémentaires ou "volumes de contrôle" sont des rectangles en 2D ou des parallélépipèdes en 3D. Cependant, la méthode des Volumes Finis permet d'utiliser des volumes de forme quelconque et donc de traiter des géométries complexes, contrairement aux Différences Finies.

De nombreux codes de simulation numérique en mécanique des fluides reposent sur cette méthode : Fluent, StarCD, CFX, FineTurbo, elsA...

II.3.2 Volumes Finis pour une loi de conservation

Considérons une loi de conservation d'une grandeur physique w dans une maille de volume Ω, faisant intervenir un flux $F(w)$ et un terme source $S(w)$. Son expression sous forme intégrale est :

$$\frac{\partial}{\partial t}\int_\Omega w\,d\Omega + \int_\Omega \operatorname{div} F(w)\,d\Omega = \int_\Omega S(w)\,d\Omega$$

Appelons Σ la surface de la maille, de normale extérieure n. Le théorème d'Ostrogradski conduit à :

$$\frac{\partial}{\partial t}\int_\Omega w\,d\Omega + \oint_\Sigma F.n\,d\Sigma = \int_\Omega S\,d\Omega$$

L'intégrale $\oint_\Sigma F.n\,d\Sigma$ représente la somme des flux à travers chaque face de la maille. Le flux est supposé constant sur chaque face, l'intégrale se ramène à une somme discrète sur chaque face de la maille. Il vient :

$$\oint_\Sigma F.n\,d\Sigma = \sum_{faces\ de\ la\ maille} F_{face}.n_{face}\,\Sigma_{face}$$

La quantité $F_{face} = F(w_{face})$ est une approximation du flux F sur une face de la maille, c'est **le flux numérique** sur la face considérée.

La discrétisation spatiale revient à calculer le bilan des flux sur une maille élémentaire. Ce bilan comprend la somme des contributions évaluées sur chaque face de la maille. La manière dont on approche les flux numériques en fonction de l'inconnue discrète détermine **le schéma numérique**. L'écriture du schéma numérique peut également utiliser des inconnues auxiliaires, par exemple le gradient de l'inconnue par maille.

Explicitons maintenant le terme de dérivée temporelle. Un élément fondamental de la discrétisation en Volumes Finis est de supposer que **la grandeur w est constante dans chaque maille** et égale à une valeur approchée de sa moyenne sur la maille ou bien à sa valeur au centre de la maille.
D'autre part, le terme de dérivation en temps est évalué au moyen d'une méthode numérique d'intégration d'équation différentielle (Runge-Kutta, Euler explicite ou implicite...) et fait intervenir un pas de temps d'intégration Δt. Ce dernier peut être constant ou variable. Pour fixer les idées, on écrira la formulation avec une méthode d'Euler explicite. Notons Δw l'incrément de la grandeur w entre deux itérations temporelles successives. On peut ainsi écrire :

$$\frac{\partial}{\partial t}\int_{\Omega} w\, d\Omega \;=\; \Omega\left(\frac{dw}{dt}\right)_{maille} \;=\; \Omega\,\frac{\Delta w}{\Delta t}$$

Finalement la loi de sconservation discrétisée avec la méthode des Volumes Finis peut s'écrire :

$$\boxed{\Omega\,\frac{\Delta w}{\Delta t} \;+\; \sum_{faces} F_{face}.n_{face}\,\Sigma_{face} \;=\; \Omega\, S}$$

La méthodes des Volumes Finis consiste donc à :

- Décomposer la géométrie en mailles élémentaires (élaborer un maillage).
- Initialiser la grandeur w sur le domaine de calcul.
- Lancer le processus d'intégration temporelle jusqu'à convergence avec :
  - ⋆ Calcul du bilan de flux par maille par un schéma numérique.
  - ⋆ Calcul du terme source.
  - ⋆ Calcul de l'incrément temporel par une méthode numérique d'intégration.
  - ⋆ Application des conditions aux limites.

II.3.2.1 Cas monodimensionnel

Considèrons une loi de conservation 1D :

$$\frac{\partial}{\partial t}\int u\, dx \;+\; \int \frac{\partial f(u)}{\partial x}\, dx \;=\; 0$$

Où u est une grandeur physique fonction de la variable d'espace x et du temps t
et $f(u)$ est une fonction de u.

Le domaine de calcul est divisé en N mailles de centre x_i. Chaque maille a une taille $h_i = x_{i+1/2} - x_{i-1/2}$. Les indices demi-entier désignent les interfaces de la maille avec les mailles voisines (voir figure II.1).

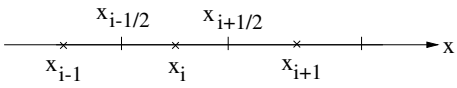

FIG. II.1 – Maillage 1D

Le temps est discrétisé en intervalles de pas constant Δt. La fonction u est supposée constante dans chaque maille et égale à une valeur approchée de la moyenne. Notons u_i^n cette valeur moyenne dans la i-ème maille de centre x_i, à l'instant $t = n\Delta t$. Ainsi :

$$\forall x \in [x_{i-1/2}, x_{i+1/2}] \text{ et } t = n\Delta t, \quad u(x,t) = u_i^n$$

Souvent, cette valeur approchée de la moyenne est la valeur de la fonction u au centre x_i de la maille, on parle alors de Volumes Finis Cell-Centered (et dans ce cas, $u_i^n = u(x_i, t)$).

Le discrétisation spatiale par les Volumes Finis consiste à intégrer maille par maille la loi de conservation :

$$\frac{\partial}{\partial t}\int_{maille} u\,dx \;+\; \int_{maille} \frac{\partial f(u)}{\partial x}\,dx \;=\; 0$$

Soit pour la i-ème maille de centre x_i, au temps $t = n\Delta t$:

$$\frac{\partial}{\partial t}\int_{x_{i-1/2}}^{x_{i+1/2}} u\,dx \;+\; \int_{x_{i-1/2}}^{x_{i+1/2}} \frac{\partial f}{\partial x}\,dx \;=\; 0$$

Ce qui s'intégre comme suit :

$$h_i \frac{\partial u_i^n}{\partial t} \;+\; \hat{f}_{i+1/2}^n - \hat{f}_{i-1/2}^n \;=\; 0$$

La quantité $\hat{f}_{i+1/2}^n$ désigne une approximation du flux $f(u)$ à l'interface $x_{i+1/2}$ et au temps $n\Delta t$. C'est le flux numérique au point $x_{i+1/2}$. Ce flux numérique s'évalue en fonction des valeurs moyennes de u dans les mailles voisines, ce qui détermine le schéma numérique.

Une méthode d'Euler explicite est utilisée pour évaluer la dérivée en temps (d'autres schémas peuvent être utilisés, par exemple le schéma de Runge-Kutta). La formulation discrètisée en Volumes Finis de la loi de conservation est ainsi :

$$\boxed{h_i \frac{u_i^{n+1} - u_i^n}{\Delta t} \;+\; \hat{f}_{i+1/2}^n - \hat{f}_{i-1/2}^n \;=\; 0}$$

II.3.2.2 Cas bidimensionnel

Considérons une loi de conservation d'une grandeur physique $u(x, y, t)$ où x et y sont les deux directions d'espace. Le domaine géométrique est divisé en mailles élémentaires, par exemple en mailles rectangulaires comme représenté sur la figure II.2. La grandeur u est supposée constante dans chaque maille et égale à une valeur approchée de la moyenne sur la maille (ou encore à la valeur au centre de la maille).

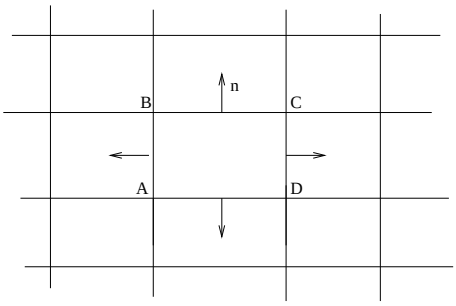

FIG. II.2 – Maillage 2D

Dans le cas bidimensionnel, le terme intégral $\oint_\Sigma F.n\, d\Sigma$ représente la circulation sur un contour d'une maille élémentaire. Plaçons-nous sur la maille de contour $ABCD$ comme indiqué sur la figure. Le flux F est supposé constant sur chaque arête de la maille AB, BC, CD et AD. L'intégrale se ramène à une somme discrète sur chaque arête :

$$\oint_\Sigma F.n\, d\Sigma = \oint_{ABCD} F.n\, dl = \sum_{AB,BC,CD,AD} F_{arete}.n_{arete}\, Longueur_{arete}$$

Ceci revient à évaluer le bilan des flux à travers chaque facette de la maille.

II.3.3 Exemple simple 1D avec conditions de Dirichlet

Considérons l'équation différentielle suivante :

$$\begin{cases} -u''(x) = f(x) & , \quad x \in]0,1[\\ u(0) = \alpha \quad \text{et} \quad u(1) = \beta \end{cases}$$

où f est une fonction continue.
L'intervalle]0,1[est discrétisé en N mailles de centre x_i et de taille $h_i = x_{i+1/2} - x_{i-1/2}$. La fonction $u(x)$ est supposé constante dans chaque maille et égale à une valeur approchée de la moyenne sur la maille considérée. On notera u_i cette valeur dans la i-ème maille de centre x_i. Ainsi, on a dans la i-ème maille : $\forall x \in [x_{i-1/2}, x_{i+1/2}]$, $u(x) = u_i$.

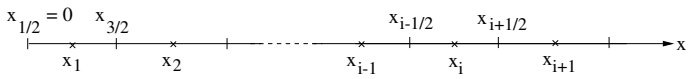

FIG. II.3 – Maillage 1D

La discrétisation spatiale par les Volumes Finis consiste à intégrer maille par maille l'équation différentielle du problème, soit pour la i-ème maille :

$$\int_{x_{i-1/2}}^{x_{i+1/2}} -u''(x)\,dx = \int_{x_{i-1/2}}^{x_{i+1/2}} f(x)\,dx$$

Ce qui donne après intégration :

$$u'(x_{i-1/2}) - u'(x_{i+1/2}) = h_i \tilde{f}_i \qquad \text{pour } i \text{ variant de 1 à N}$$

où $\tilde{f}_i$ désigne la valeur moyenne de f sur la i-ème maille : $\tilde{f}_i = \dfrac{1}{h_i} \displaystyle\int_{x_{i-1/2}}^{x_{i+1/2}} f(x)\,dx$

Il reste maintenant à exprimer $u'(x_{i-1/2})$ en fonction des inconnues u_i. L'approximation la plus naturelle est de prendre la valeur moyenne de $u'(x)$ sur le segment $[x_{i-1}, x_i]$, soit :

$$u'(x_{i-1/2}) = \frac{1}{\frac{h_{i-1}+h_i}{2}} \int_{x_{i-1}}^{x_i} u'(x)\,dx = \frac{u(x_i) - u(x_{i-1})}{h_{i-1/2}} = \frac{u_i - u_{i-1}}{h_{i-1/2}}$$

avec $h_{i-1/2} = \dfrac{h_{i-1} + h_i}{2}$

Cette dernière expression n'est pas valable au bord gauche, pour $i = 1$, en $x_{1/2} = 0$, car elle fait intervenir le point x_0 qui n'est pas défini. Il se pose alors le problème du traitement des bords qui exige une formulation particulière. Une possibilité est de définir une maille fictive à gauche de l'intervalle [0,1], et d'affecter une valeur moyenne de la fonction u dans cette maille. Une autre possibilité est de considérer la valeur moyenne de $u'(x_{1/2})$ non plus sur le segment $[x_0, x_1]$ qui

n'est pas défini mais sur le segment $[x_{1/2}, x_1]$, c'est ce que nous choisissons dans cet exemple. Ainsi on écrit :

$$u'(x_{1/2}) = \frac{2}{h_1}\int_{x_{1/2}}^{x_1} u'(x)\,dx = \frac{2(u_1 - u(0))}{h_1} = \frac{2(u_1-\alpha)}{h_1}$$

Et de même pour le terme $u'(x_{i+1/2})$, on écrit que : $u'(x_{i+1/2}) = \dfrac{u_{i+1}-u_i}{h_{i+1/2}}$. Le même problème survient au bord droit, pour $i = N$, en $x_{N+1/2} = 1$. On considère la valeur moyenne de $u'(x_{N+1/2})$ non plus sur le segment $[x_N, x_{N+1}]$ qui n'est pas défini mais sur le segment $[x_N, x_{N+1/2}]$, soit :

$$u'(x_{N+1/2}) = \frac{2}{h_N}\int_{x_N}^{x_{N+1/2}} u'(x)\,dx = \frac{2(u(1)-u_N)}{h_N} = \frac{2(\beta-u_N)}{h_N}$$

La discrétisation en Volumes Finis est donc finalement :

$$\boxed{\begin{array}{rcll}
\dfrac{u_i-u_{i-1}}{h_{i-1/2}} - \dfrac{u_{i+1}-u_i}{h_{i+1/2}} & = & h_i\tilde{f}_i & \text{pour } i \text{ variant de 2 à N-1} \\
\dfrac{2(u_1-\alpha)}{h_1} - \dfrac{u_2-u_1}{h_{3/2}} & = & h_1\tilde{f}_1 & \\
\dfrac{u_N-u_{N-1}}{h_{N-1/2}} - \dfrac{2(\beta-u_N)}{h_N} & = & h_N\tilde{f}_N &
\end{array}}$$

Dans le cas particulier d'un maillage régulier de pas h. La discrétisation en Volumes Finis devient :

$$\begin{array}{rcll}
\dfrac{2u_i-u_{i-1}-u_{i+1}}{h^2} & = & \tilde{f}_i & \text{pour } i \text{ variant de 2 à N-1} \\
\dfrac{3u_1-u_2}{h^2} & = & \tilde{f}_1 + \dfrac{2\alpha}{h^2} & \\
\dfrac{3u_N-u_{N-1}}{h^2} & = & \tilde{f}_N + \dfrac{2\beta}{h^2} &
\end{array}$$

Sous forme matricielle, ceci s'exprime :

$$\frac{1}{h^2}\begin{bmatrix} 3 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 3 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} \tilde{f}_1 + 2\alpha/h^2 \\ \tilde{f}_2 \\ \vdots \\ \tilde{f}_{N-1} \\ \tilde{f}_N + 2\beta/h^2 \end{bmatrix}$$

Comparaison avec un schéma aux Différences Finies

Nous introduisons un schéma aux Différences Finies afin de comparer les deux méthodes de discrétisation.

On se place sur le même maillage construit pour les Volumes Finis. Les points x_i seront considérés comme les noeuds du maillage pour les Différences Finies. Et la quantité u_i désignera alors la valeur de la fonction u au noeud x_i.
ATTENTION aux notations propres aux deux méthodes, dans un cas u_i désigne une valeur moyenne de u sur la i-ème maille, et dans l'autre cas, u_i désigne la valeur de u en x_i.

L'équation à résoudre s'écrit, sous forme discrète en chaque noeud x_i :

$$-\left(\frac{d^2u}{dx^2}\right)_i = f(x_i) = f_i$$

Approximons la dérivée seconde de u au moyen d'un schéma à l'ordre 2. On écrit les développements de Taylor de u_{i+1} et u_{i-1} au voisinage de x_i :

$$\begin{aligned}
u_{i+1} &= u(x_i + h_{i+1/2}) = u_i + h_{i+1/2}\left(\frac{du}{dx}\right)_i + \frac{h_{i+1/2}^2}{2}\left(\frac{d^2u}{dx^2}\right)_i + \mathcal{O}(h_{i+1/2}^3) \\
u_{i-1} &= u(x_i - h_{i-1/2}) = u_i - h_{i-1/2}\left(\frac{du}{dx}\right)_i + \frac{h_{i-1/2}^2}{2}\left(\frac{d^2u}{dx^2}\right)_i + \mathcal{O}(h_{i-1/2}^3)
\end{aligned}$$

Ce qui peut s'exprimer sous la forme :

$$\begin{aligned}
\frac{u_{i+1} - u_i}{h_{i+1/2}} &= \left(\frac{du}{dx}\right)_i + \frac{h_{i+1/2}}{2}\left(\frac{d^2u}{dx^2}\right)_i + \mathcal{O}(h_{i+1/2}^2) \\
\frac{u_{i-1} - u_i}{h_{i-1/2}} &= -\left(\frac{du}{dx}\right)_i + \frac{h_{i-1/2}}{2}\left(\frac{d^2u}{dx^2}\right)_i + \mathcal{O}(h_{i-1/2}^2)
\end{aligned}$$

Par somme des deux égalités, on obtient une approximation à l'ordre 2 de la dérivée seconde de u. Au final, la discrétisation par des Différences Finies est la suivante :

$$\boxed{\frac{2}{h_{i+1/2} + h_{i-1/2}}\left(\frac{u_i - u_{i-1}}{h_{i-1/2}} - \frac{u_{i+1} - u_i}{h_{i+1/2}}\right) = f_i \qquad \text{pour } i \text{ variant de 1 à N}}$$

Dans la cas particulier d'un maillage régulier de pas h, l'équation discrétisée s'écrit :

$$\frac{2u_i - u_{i+1} - u_{i-1}}{h^2} = f_i \qquad \text{pour } i \text{ variant de 1 à N}$$

II.3.4 Exemple simple 1D avec conditions mixtes Dirichlet-Neumann

Considérons l'équation différentielle suivante :

$$\begin{cases} -u''(x) = f(x) & , \quad x \in]0,1[\\ u(0) = \alpha \quad \text{et} \quad u'(1) = \beta \end{cases}$$

où l'on a cette fois une condition de Neumann en $x = 1$.
On se place sur le même maillage que précédemment et on adopte la même démarche.

L'équation intégrée sur une maille élémentaire est :

$$u'(x_{i-1/2}) - u'(x_{i+1/2}) \;=\; h_i \tilde{f}_i \qquad \text{pour } i \text{ variant de 1 à N}$$

Le calcul des termes de dérivée aux interfaces s'effectue de la même manière que précédemment. Au bord droit, à l'interface $x_{N+1/2} = 1$, l'application de la condition $u'(1) = \beta$ s'applique très naturellement et l'on a : $u'(x_{N+1/2}) = \beta$.

La discrétisation en Volumes Finis est donc finalement :

$$\boxed{\begin{aligned} \frac{u_i - u_{i-1}}{h_{i-1/2}} - \frac{u_{i+1} - u_i}{h_{i+1/2}} &= h_i \tilde{f}_i \qquad \text{pour } i \text{ variant de 2 à N-1} \\ \frac{2(u_1 - \alpha)}{h_1} - \frac{u_2 - u_1}{h_{3/2}} &= h_1 \tilde{f}_1 \\ \frac{u_N - u_{N-1}}{h_{N-1/2}} - \beta &= h_N \tilde{f}_N \end{aligned}}$$

Dans le cas particulier d'un maillage régulier de pas h. La discrétisation en Volumes Finis devient :

$$\begin{aligned} \frac{2u_i - u_{i-1} - u_{i+1}}{h^2} &= \tilde{f}_i \qquad \text{pour } i \text{ variant de 2 à N-1} \\ \frac{3u_1 - u_2}{h^2} &= \tilde{f}_1 + \frac{2\alpha}{h^2} \\ \frac{u_N - u_{N-1}}{h^2} &= \tilde{f}_N + \frac{\beta}{h} \end{aligned}$$

Sous forme matricielle, ceci s'exprime :

$$\frac{1}{h^2} \begin{bmatrix} 3 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} \tilde{f}_1 + 2\alpha/h^2 \\ \tilde{f}_2 \\ \vdots \\ \tilde{f}_{N-1} \\ \tilde{f}_N + \beta/h \end{bmatrix}$$

II.3.5 Discrétisation de l'équation de la chaleur 1D

Considérons le problème monodimensionnel de la conduction de la chaleur dans une barre de 1m de longueur. Le champ de température $T(x,t)$ vérifie l'équation de la chaleur :

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}$$

où α est la diffusivité thermique que l'on supposera égale à 1.
A cette EDP s'ajoute deux conditions aux limites aux extrémités de la barre $T(0,t) = T_g$ et $T(1,t) = T_d$ ainsi qu'une condition initale $T(x,0) = T_0$.

L'intervalle [0,1] est discrétisé en N mailles de centre x_i (i variant de 1 à N), de taille $\Delta x = x_{i+1/2} - x_{i-1/2}$ constante. Le temps est discrétisé en intervalles de pas constant Δt. A chaque instant, la température $T(x,t)$ est supposée constante dans chaque maille et égale à une valeur approchée de la moyenne sur la maille considérée. On notera T_i^n cette valeur dans la i-ème maille de centre x_i à l'instant $t = n\Delta t$.

La discrétisation spatiale par les Volumes Finis consiste à intégrer maille par maille l'EDP du problème, soit pour la i-ème maille :

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \frac{\partial T}{\partial t}\, dx = \int_{x_{i-1/2}}^{x_{i+1/2}} \frac{\partial^2 T}{\partial x^2}\, dx$$

Nous utilisons un schéma d'Euler explicite pour évaluer la dérivée temporelle, il vient :

$$\Delta x \frac{T_i^{n+1} - T_i^n}{\Delta t} = \left[\left(\frac{\partial T}{\partial x}\right)^n_{x_{i+1/2}} - \left(\frac{\partial T}{\partial x}\right)^n_{x_{i-1/2}} \right]$$

Les termes de dérivée première aux interfaces $x_{i+1/2}$ sont évalués en considérant la valeur moyenne de $\dfrac{\partial T}{\partial x}$ sur le segment $[x_i, x_{i+1}]$, soit :

$$\left(\frac{\partial T}{\partial x}\right)^n_{x_{i+1/2}} = \frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} \frac{\partial T}{\partial x}\, dx = \frac{T_{i+1}^n - T_i^n}{\Delta x}$$

Cette formulation n'est pas valable dans la maille N à l'extrémité droite de la barre. Dans cette maille, on considère la valeur moyenne calculée sur l'intervalle $[x_N, 1]$. D'où :

$$\left(\frac{\partial T}{\partial x}\right)^n_{x_{N+1/2}} = \frac{2}{\Delta x} \int_{x_N}^{1} \frac{\partial T}{\partial x}\, dx = 2\frac{T_d^n - T_N^n}{\Delta x}$$

De même, les termes de dérivée première aux interfaces $x_{i-1/2}$ sont évalués en considérant la valeur moyenne de $\dfrac{\partial T}{\partial x}$ sur le segment $[x_{i-1}, x_i]$, soit :

$$\left(\frac{\partial T}{\partial x}\right)^n_{x_{i-1/2}} = \frac{1}{\Delta x} \int_{x_{i-1}}^{x_i} \frac{\partial T}{\partial x}\, dx = \frac{T_i^n - T_{i-1}^n}{\Delta x}$$

Avec un problème dans la première maille à l'extrémité gauche de la barre. Dans cette maille, on considère la valeur moyenne calculée sur l'intervalle $[0, x_1]$. D'où :

$$\left(\frac{\partial T}{\partial x}\right)^n_{x_{1/2}} = \frac{2}{\Delta x}\int_0^{x_1}\frac{\partial T}{\partial x}\,dx = 2\frac{T_1^n - T_g^n}{\Delta x}$$

En posant $\lambda = \dfrac{\Delta t}{\Delta x^2}$, la température à l'itération $n+1$ est donnée par :

$$\begin{aligned}
T_i^{n+1} &= \lambda T_{i-1}^n + (1-2\lambda)T_i^n + \lambda T_{i+1}^n \qquad i \text{ variant de 2 à N-1}\\
T_1^{n+1} &= 2\lambda T_g + (1-3\lambda)T_1^n + \lambda T_2^n\\
T_N^{n+1} &= \lambda T_{N-1}^n + (1-3\lambda)T_N^n + 2\lambda T_d^n
\end{aligned}$$

Soit sous forme matricielle :

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N-1} \\ T_N \end{bmatrix}^{n+1} = \begin{bmatrix} 1-3\lambda & \lambda & 0 & \cdots & 0 \\ \lambda & 1-2\lambda & \lambda & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \lambda & 1-2\lambda & \lambda \\ 0 & 0 & 0 & \lambda & 1-3\lambda \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N-1} \\ T_N \end{bmatrix}^n + 2\lambda \begin{bmatrix} T_g \\ 0 \\ \vdots \\ 0 \\ T_d \end{bmatrix}$$

II.3.6 Discrétisation de l'équation de la chaleur 2D stationnaire

Considérons le problème bidimensionnel stationnaire de la conduction de la chaleur dans un domaine rectangulaire $[0,L_x]\times[0,L_y]$. Le champ de température $T(x,y)$ vérifie l'équation de Laplace :

$$\begin{cases} \Delta T = \dfrac{\partial^2 T}{\partial x^2} + \dfrac{\partial^2 T}{\partial y^2} = 0 \quad , \quad (x,y) \in [0,L_x]\times[0,L_y] \\ T(0,y) = T_g \quad \text{et} \quad T(L_x,y) = T_d \qquad 0 < y < L_y \\ T(x,0) = T_b \quad \text{et} \quad T(x,L_y) = T_h \qquad 0 < x < L_x \end{cases}$$

Le domaine de calcul est discrétisé en N×P mailles de centre (x_i, y_j) (i variant de 1 à N et j variant de 1 à P). On supposera que les pas d'espace dans chaque direction $\Delta x = x_{i+1/2} - x_{i-1/2}$ et $\Delta y = y_{j+1/2} - y_{j-1/2}$ sont constants.
La température $T(x,y)$ est supposée constante dans chaque maille et égale à une valeur approchée de la moyenne sur la maille considérée. On notera T_{ij} cette valeur dans la maille (i,j).

La discrétisation spatiale par les Volumes Finis consiste à intégrer maille par maille l'EDP du problème, soit pour la maille (i,j) de centre (x_i, y_j) :

$$\int_{x_{i-1/2}}^{x_{i+1/2}}\int_{y_{j-1/2}}^{y_{j+1/2}}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)dxdy = 0$$

Il vient :

$$\int_{y_{j-1/2}}^{y_{j+1/2}} \left[\left(\frac{\partial T}{\partial x}\right)_{x_{i+1/2}} - \left(\frac{\partial T}{\partial x}\right)_{x_{i-1/2}} \right] dy + \int_{x_{i-1/2}}^{x_{i+1/2}} \left[\left(\frac{\partial T}{\partial y}\right)_{y_{j+1/2}} - \left(\frac{\partial T}{\partial y}\right)_{y_{j-1/2}} \right] dx = 0$$

Le terme de dérivée première $\left(\frac{\partial T}{\partial x}\right)_{x_{i+1/2}}$ à l'interface $x_{i+1/2}$ est évalué en calculant une valeur moyenne sur l'intervalle $[x_i,x_{i+1}]$:

$$\left(\frac{\partial T}{\partial x}\right)_{x_{i+1/2}} = \frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} \frac{\partial T}{\partial x} dx = \frac{T_{i+1,j} - T_{ij}}{\Delta x}$$

De même, le terme $\left(\frac{\partial T}{\partial x}\right)_{x_{i-1/2}}$ à l'interface $x_{i-1/2}$ est évalué en calculant une valeur moyenne sur l'intervalle $[x_{i-1},x_i]$. Ce qui permet d'écrire :

$$\begin{aligned} \int_{y_{j-1/2}}^{y_{j+1/2}} \left[\left(\frac{\partial T}{\partial x}\right)_{x_{i+1/2}} - \left(\frac{\partial T}{\partial x}\right)_{x_{i-1/2}} \right] dy &= \int_{y_{j-1/2}}^{y_{j+1/2}} \left[\frac{T_{i+1,j} - T_{ij}}{\Delta x} - \frac{T_{i,j} - T_{i-1,j}}{\Delta x} \right] dy \\ &= \Delta y \frac{T_{i+1,j} + T_{i-1,j} - 2T_{ij}}{\Delta x} \end{aligned}$$

En opérant identiquement pour les termes $\frac{\partial T}{\partial y}$ aux interfaces $y_{j+1/2}$ et $y_{j-1/2}$, on aboutit à l'expression suivante valable pour i variant de 2 à $N-1$ et j variant de 2 à $P-1$:

$$\Delta y^2(T_{i+1,j} + T_{i-1,j}) + \Delta x^2(T_{i,j+1} + T_{i,j-1}) - 2(\Delta x^2 + \Delta y^2)T_{ij} = 0$$

Cette relation n'est pas valable aux bords du domaine pour lesquels les termes de dérivées premières sont évalués en considérant une valeur moyenne sur une demie-maille.
Par exemple, pour la dérivée $\left(\frac{\partial T}{\partial x}\right)_{x_{1/2}}$, la valeur moyenne sera calculée sur l'intervalle $[0,x_1]$ et fera intervenir les conditions aux limites (la température T_g au bord gauche) :

$$\left(\frac{\partial T}{\partial x}\right)_{x_{1/2}} = \frac{2}{\Delta x} \int_0^{x_1} \frac{\partial T}{\partial x} dx = 2\frac{T_{1j} - T_g}{\Delta x}$$

Ainsi pour les cellules adjacentes au bord gauche ($i = 1$, j variant de 1 à P), la formulation est :

$$\begin{aligned} \Delta y^2(T_{2,j} + 2T_g) + \Delta x^2(T_{1,j+1} + T_{1,j-1}) - (2\Delta x^2 + 3\Delta y^2)T_{1j} &= 0 \quad \text{; j=2 à P-1} \\ \Delta y^2(T_{21} + 2T_g) + \Delta x^2(T_{12} + 2T_b) - 3(\Delta x^2 + \Delta y^2)T_{11} &= 0 \quad \text{; j=1} \\ \Delta y^2(T_{2P} + 2T_g) + \Delta x^2(2T_h + T_{1,P-1}) - 3(\Delta x^2 + \Delta y^2)T_{1P} &= 0 \quad \text{; j=P} \end{aligned}$$

On aura une formulation équivalente pour les cellules adjacentes aux 3 autres bords du domaine. Soit sous forme matricielle, pour N=P=3, en posant $A = \Delta x^2 + \Delta y^2$, $B = 3\Delta x^2 + 2\Delta y^2$ et $C = 2\Delta x^2 + 3\Delta y^2$:

$$\begin{bmatrix} -3A & \Delta y^2 & 0 & \Delta x^2 & 0 & 0 & 0 & 0 & 0 \\ \Delta y^2 & -B & \Delta y^2 & 0 & \Delta x^2 & 0 & 0 & 0 & 0 \\ 0 & \Delta y^2 & -3A & 0 & 0 & \Delta x^2 & 0 & 0 & 0 \\ \Delta x^2 & 0 & 0 & -C & \Delta y^2 & 0 & \Delta x^2 & 0 & 0 \\ 0 & \Delta x^2 & 0 & \Delta y^2 & -2A & \Delta y^2 & 0 & \Delta x^2 & 0 \\ 0 & 0 & \Delta x^2 & 0 & \Delta y^2 & -C & 0 & 0 & \Delta x^2 \\ 0 & 0 & 0 & \Delta x^2 & 0 & 0 & -3A & \Delta y^2 & 0 \\ 0 & 0 & 0 & 0 & \Delta x^2 & 0 & \Delta y^2 & -B & \Delta y^2 \\ 0 & 0 & 0 & 0 & 0 & \Delta x^2 & 0 & \Delta y^2 & -3A \end{bmatrix} \begin{bmatrix} T_{11} \\ T_{21} \\ T_{31} \\ T_{12} \\ T_{22} \\ T_{32} \\ T_{13} \\ T_{23} \\ T_{33} \end{bmatrix} = -2 \begin{bmatrix} \Delta x^2 T_b + \Delta y^2 T_g \\ \Delta x^2 T_b \\ \Delta x^2 T_b + \Delta y^2 T_d \\ \Delta y^2 T_g \\ 0 \\ \Delta y^2 T_d \\ \Delta x^2 T_h + \Delta y^2 T_g \\ \Delta x^2 T_h \\ \Delta x^2 T_h + \Delta y^2 T_d \end{bmatrix}$$

Dans le cas où les pas d'espace sont identiques $\Delta x = \Delta y$, la formulation matricielle, pour N=P=3 devient :

$$\begin{bmatrix} -6 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -5 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -6 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -5 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -5 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -6 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -5 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -6 \end{bmatrix} \begin{bmatrix} T_{11} \\ T_{21} \\ T_{31} \\ T_{12} \\ T_{22} \\ T_{32} \\ T_{13} \\ T_{23} \\ T_{33} \end{bmatrix} = -2 \begin{bmatrix} T_b + T_g \\ T_b \\ T_b + T_d \\ T_g \\ 0 \\ T_d \\ T_h + T_g \\ T_h \\ T_h + T_d \end{bmatrix}$$

<u>Remarque</u> : dans le cas de conditions aux limites mixtes Dirichlet-Neumann, la condition de flux de chaleur est prise en compte très simplement, directement dans les termes de dérivées aux interfaces du bord concerné.

II.4 LES ELEMENTS FINIS EN 1D

II.4.1 Introduction

La méthode des Eléments Finis consiste à approcher, dans un sous-espace de dimension finie, un problème écrit sous forme variationnelle dans un espace de dimension infinie. Cette forme variationnelle est équivalente à une forme de minimisation de l'énergie en général (principe des travaux virtuels). La solution approchée est dans ce cas une fonction déterminée par un nombre fini de paramètres, par exemple, ses valeurs en certains points (les noeuds du maillage).

Cette méthode est particulièrement bien adaptée aux problèmes d'équilibre. Elle permet de traiter des géométries complexes contrairement aux Différences Finies mais elle demande un grand coût de temps de calcul et de mémoire.

De nombreux codes de calculs de structure reposent sur les Eléments Finis : ANSYS, CADDS, CATIA...

II.4.2 Exemple simple 1D

Reprenons le cas précédent de l'équation différentielle :

$$\begin{cases} -u''(x) = f(x) \quad , \qquad x \in]0,1[\\ u(0) = u(1) = 0 \end{cases}$$

La présentation très succinte faite sur cet exemple simple a pour but de donner les idées de base et ne constitue qu'une première introduction à la méthodes des Eléments Finis. L'approche repose sur la **méthode de Galerkin** qui permet d'écrire le système différentiel sous forme variationnelle dans un espace de dimension finie.

Soit une fonction $v(x) \in \mathcal{C}^1([0,1])$, nulle en 0 et 1. On peut écrire :

$$-\int_0^1 u''(x)\, v(x)\, dx \;=\; \int_0^1 f(x)\, v(x)\, dx$$

En intégrant par parties, il vient :

$$\int_0^1 u'(x)\, v'(x)\, dx \;=\; \int_0^1 f(x)\, v(x)\, dx \qquad \forall v \in V \tag{II.1}$$

avec $V = \big\{ v \in \mathcal{C}^0([0,1]);\, v(0) = v(1) = 0\,,\text{v' continue par morceaux} \big\}$ un sous-espace vectoriel de $\mathcal{C}^1([0,1])$.

Une solution de la forme variationnelle (II.1) s'appelle **solution faible** du problème différentiel de départ.

On cherche alors à écrire un problème approché dans un sous-espace vectoriel de dimension finie. Soit $\tilde{V}$ un sous-espace vectoriel de V de dimension N finie. Soient $\phi_1, \phi_2, ..., \phi_N$ N fonctions linéairement indépendantes de V. Ces fonctions constituent une base du sous-espace $\tilde{V}$. Ainsi, toute fonction $\tilde{u}$ de $\tilde{V}$ peut se décomposer selon :

$$\tilde{u}(x) = \sum_{j=1}^{N} u_j \phi_j(x)$$

Résoudre le problème différentiel de départ revient alors à chercher une solution $\tilde{u} \in \tilde{V}$ telle que :

$$\int_0^1 \tilde{u}'(x)\, \tilde{v}'(x)\, dx \;=\; \int_0^1 f(x)\, \tilde{v}(x)\, dx \qquad \forall \tilde{v} \in \tilde{V}$$

C'est-à-dire chercher N réels $u_1, u_2, ..., u_N$ vérifiant :

$$\sum_{j=1}^{N} u_j \int_0^1 \phi_j'(x)\, \tilde{v}'(x)\, dx \;=\; \int_0^1 f(x)\, \tilde{v}(x) dx \qquad \forall \tilde{v} \in \tilde{V}$$

Ou encore :

$$\sum_{j=1}^{N} u_j \int_0^1 \phi_j'(x)\, \phi_i'(x)\, dx \;=\; \int_0^1 f(x)\, \phi_i(x)\, dx \qquad \forall \phi_i \in \tilde{V}$$

Soient A la matrice $N \times N$ d'élément courant a_{ij} et B le vecteur à N composantes b_i définies par :

$$a_{ij} = \int_0^1 \phi_j'(x)\, \phi_i'(x)\, dx \qquad \text{et} \qquad b_i = \int_0^1 f(x)\, \phi_i(x)\, dx$$

Par définition, la matrice A est symétrique. Notons U le vecteur des N inconnues $u_1, u_2, ..., u_N$. Le problème différentiel se ramène finalement à la résolution du système linéaire :

$$A.U \;=\; B$$

Il reste maintenant à choisir les N fonctions ϕ_i de façon à ce que le système soit simple à résoudre numériquement.

II.4.2.1 Choix des fonctions ϕ_i : les éléments finis

L'intervalle]0,1[est discrétisé en N points de coordonnées x_i. Les fonctions $\phi_i(x)$ sont choisies comme fonctions polynomiales de degré 1 définies par :

$$\phi_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & \text{si} \quad x_{i-1} \leq x \leq x_i \\ \dfrac{x - x_{i+1}}{x_i - x_{i+1}} & \text{si} \quad x_i \leq x \leq x_{i+1} \\ 0 & \text{sinon} \end{cases}$$

Ces fonctions sont appelées les éléments finis de degré 1. Avec ces éléments finis, la matrice A est tridiagonale. Il est aussi possible de choisir pour éléments finis des fonctions de degré 2 ou plus.

Le calcul de la matrice A fait intervenir les dérivées $\phi_i'(x)$ simples à calculer :

$$\phi_i'(x) = \begin{cases} \dfrac{1}{x_i - x_{i-1}} & \text{si} \quad x_{i-1} \leq x \leq x_i \\ \dfrac{1}{x_i - x_{i+1}} & \text{si} \quad x_i \leq x \leq x_{i+1} \\ 0 & \text{sinon} \end{cases}$$

Calculons maintenant les éléments de la matrice A, tridiagonale et symétrique. Les trois termes des diagonales sont :

$$\begin{aligned} a_{ii} &= \int_0^1 \phi_i'(x)\,\phi_i'(x)\,dx = \frac{1}{x_i - x_{i-1}} + \frac{1}{x_{i+1} - x_i} \\ a_{i,i+1} &= \int_0^1 \phi_{i+1}'(x)\,\phi_i'(x)\,dx = \frac{-1}{x_{i+1} - x_i} \\ a_{i-1,i} &= \int_0^1 \phi_i'(x)\,\phi_{i-1}'(x)\,dx = \frac{-1}{x_i - x_{i-1}} \end{aligned}$$

Et calculons les composantes du vecteur B par une méthode des trapèzes (chaque intégrale sur un segment élémentaire sera évaluée comme l'aire du trapèze correspondant), soit :

$$b_i = \int_0^1 f(x)\,\phi_i(x)\,dx = f_i \left(\frac{x_{i+1} - x_{i-1}}{2} \right)$$

Le système linéaire à résoudre s'écrit donc, sous forme indicielle :

$$\boxed{\frac{u_i - u_{i-1}}{x_i - x_{i-1}} - \frac{u_{i+1} - u_i}{x_{i+1} - x_i} = \frac{x_{i+1} - x_{i-1}}{2} f_i \qquad \text{pour } i \text{ variant de 1 à N}}$$

On rappelle la discrétisation avec un schéma aux Différences Finies d'ordre 2 :

$$\frac{u_i - u_{i-1}}{x_i - x_{i-1}} - \frac{u_{i+1} - u_i}{x_{i+1} - x_i} = \frac{x_{i+1} - x_{i-1}}{2} f_i \qquad \text{pour } i \text{ variant de 1 à N}$$

Ainsi, on constate que les deux méthodes sont rigoureusement identiques. Ceci n'est plus vérifié quand les composantes du vecteur B ne sont plus évaluées avec une méthode des trapèzes.

Dans le cas où les N points de l'intervalle]0,1[sont régulièrement espacés avec un pas h. La discrétisation en Eléments Finis devient :

$$\frac{2u_i - u_{i+1} - u_{i-1}}{h^2} = f_i \qquad \text{pour } i \text{ variant de 1 à N}$$

Soit sous forme matricielle :

$$\frac{1}{h^2}\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N-1} \\ f_N \end{bmatrix}$$

II.4.2.2 Bilan

La méthodes des Eléments Finis 1D consiste donc à :

- Choisir N points entre 0 et 1 et choisir les fonctions ϕ_i
- Construire la matrice A
- Déterminer le vecteur B (avec une méthode d'intégration)
- Résoudre le système linéaire $A.U = B$ où U désigne le vecteur des inconnues

II.5 APPLICATION NUMERIQUE

Comparons les trois méthodes de discrétisation sur le cas simple précédemment exposé. On choisit comme fonction $f(x) = \sin(\pi x)$. L'équation différentielle à résoudre est donc :

$$\begin{cases} -u''(x) = \sin(\pi x) \quad , \qquad x \in]0,1[\\ u(0) = u(1) = 0 \end{cases}$$

La solution analytique au problème est $u(x) = \dfrac{\sin(\pi x)}{\pi^2}$. Notons par un indice 'a' la solution analytique.

Divisons l'intervalle]0,1[en dix segments réguliers de pas $h = 0.1$. Pour les discrétisations avec les Différences Finies et les Elements Finis, il y a $N = 9$ noeuds de calculs. Et pour la méthode des Volumes Finis, il y a $N = 10$ mailles de calculs.

La solution discrète obtenue avec les Différences Finies (ou les Eléments Finis) est reportée dans le tableau V.1 :

x_i	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
u_i	0.0316	0.06	0.0826	0.09716	0.10216	0.09716	0.0826	0.06	0.0316
$(u_i)_a$	0.0313	0.0595	0.082	0.09636	0.10113	0.09636	0.082	0.0595	0.0313
|erreur|	$9.6\,10^{-3}$	$8.4\,10^{-3}$	$7.3\,10^{-3}$	$8.3\,10^{-3}$	$1\,10^{-2}$	$8.3\,10^{-3}$	$7.3\,10^{-3}$	$8.4\,10^{-3}$	$9.6\,10^{-3}$

TAB. II.1 – Méthode des Différences Finies et des Eléments Finis

La valeur moyenne par maille obtenue avec les Volumes Finis est reportée dans le tableau II.2. Le calcul de la valeur moyenne de $f(x)$ dans la i-ème maille est : $\tilde{f}_i = f_i \left(\dfrac{\sin \frac{\pi}{2} h}{\frac{\pi}{2} h} \right)$. Notons $(\tilde{u}_i)_a$ la valeur moyenne de la solution analytique calculée sur la i-ème maille soit :
$(\tilde{u}_i)_a = \dfrac{1}{h} \displaystyle\int_{x_{i-1/2}}^{x_{i+1/2}} u(x)\, dx = u_i \left(\dfrac{\sin \frac{\pi}{2} h}{\frac{\pi}{2} h} \right)$.

x_i	0.05	0.15	0.25	0.35	0.45	0.55	0.65	0.75	0.85	0.95
u_i	0.01589	0.04612	0.07184	0.0905	0.1003	0.1003	0.0905	0.07184	0.04612	0.01589
$(\tilde{u}_i)_a$	0.01585	0.046	0.07164	0.09028	0.1001	0.1001	0.09028	0.07164	0.046	0.01585
|erreur|	$2.5\,10^{-3}$	$2.6\,10^{-3}$	$2.8\,10^{-3}$	$2.4\,10^{-3}$	$2\,10^{-3}$	$2\,10^{-3}$	$2.4\,10^{-3}$	$2.8\,10^{-3}$	$2.6\,10^{-3}$	$2.5\,10^{-3}$

TAB. II.2 – Méthode des Volumes Finis

Les trois méthodes permettent d'obtenir des résultats avec une bonne précision. L'erreur la plus faible est obtenue avec la méthode des Volumes Finis.

II.6 CONSISTANCE, CONVERGENCE ET STABILITE

Un certain nombre de notion est nécessaire lors de la résolution d'équations aux dérivées partielles (EDP) au moyen de leurs équivalents discrétisés. Les trois principales sont **la convergence**, **la stabilité** et **la consistance**. Ces trois propriétés permettent de relier la solution exacte des équations continues à la solution exacte des équations discrétisées et à la solution numérique obtenue. Ces différents liens, résumés sur la figure II.4, sont :

- la stabilité, c'est la propriété qui assure que la différence entre la solution numérique obtenue et la solution exacte des équations discrétisées est bornée.

- la consistance, c'est la propriété qui assure que la solution exacte des équations discrétisées tende vers la solution exacte des équations continues lorsque le pas de discrétisation (Δt et Δx) tendent vers zéro.

- la convergence, c'est la propriété qui assure que la solution numérique tende vers la (ou une) solution exacte des équations continues. C'est évidemment la propriété la plus recherchée !

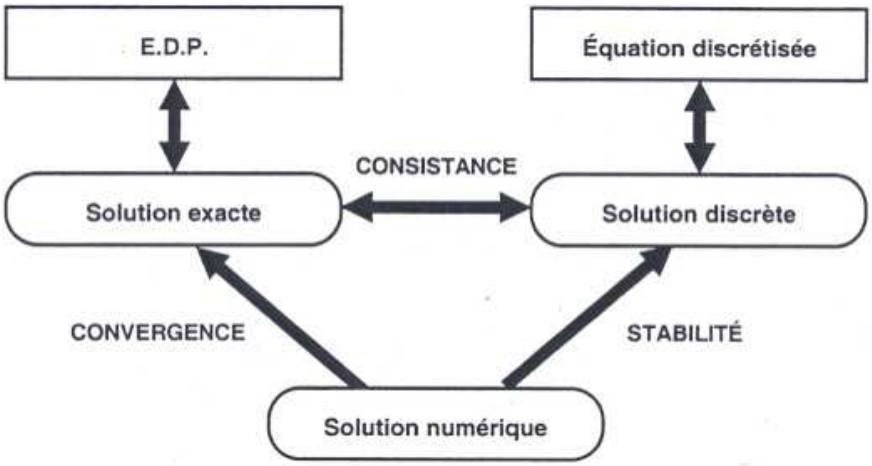

FIG. II.4 – Solutions exacte, numérique et discrète

Ces propriétés sont liées les unes aux autres par des théorèmes :

Le théorème de Lax

Dans un problème bien posé, et avec un schéma numérique consistant, la stabilité est une condition nécessaire et suffisante pour la convergence.

Le théorème de Lax-Wendroff

Si un schéma numérique consistant converge lorsqu'on raffine les pas de temps et d'espace, c'est-à-dire lorsque $\Delta t \to 0$ et $\Delta x \to 0$, alors il converge vers une solution faible des équations.

Condition de stabilité CFL

Pour des problèmes d'évolution temporelle, certains schémas sont stables à condition que le pas de temps soit inférieur à une certaine valeur critique fonction du pas d'espace. Cette inégalité constitue la condition de Courant-Friedrichs-Lewy (1928) ou **condition CFL**. Elle est nécessaire et suffisante pour assurer la stabilité.

La condition CFL varie d'une équation à une autre.

Par exemple pour l'équation de la chaleur 1D, les schémas explicites sont stables sous la condition CFL suivante :

$$\alpha \frac{\Delta t}{\Delta x^2} < 0.5$$

alors que les schémas implicites sont toujours stables.

Chapitre III

CLASSIFICATION DES EDP D'ORDRE 2

III.1 CLASSIFICATION DES EDP LINEAIRES D'ORDRE 2

Considérons une équation aux dérivées partielles (EDP) du second ordre ayant la forme suivante :

$$\boxed{a\frac{\partial^2 u}{\partial x^2} + b\frac{\partial^2 u}{\partial x \partial y} + c\frac{\partial^2 u}{\partial y^2} + d\frac{\partial u}{\partial x} + e\frac{\partial u}{\partial y} + fu = g} \tag{III.1}$$

dans laquelle u est une fonction de deux variables x et y.

On dit que l'équation (III.1) est :

- hyperbolique ssi $b^2 - 4ac > 0$
- parabolique ssi $b^2 - 4ac = 0$
- elliptique ssi $b^2 - 4ac > 0$

Remarque 1 : la terminologie utilisée dans cette définition est basée sur la classification des coniques du plan. On rappelle que la conique d'équation :

$$ax^2 + bxy + cy^2 + dx + ey + f = 0$$

est une hyperbole (resp. une parabole, une ellipse) ssi $b^2 - 4ac$ est positif (resp. nul, négatif).

Remarque 2 : Si les coefficients a, b, ..., g dépendent des variables x et y, le type de l'équation (III.1) est local. L'équation est hyperbolique au point (x_0, y_0) ssi $b(x_0, y_0)^2 - 4a(x_0, y_0)c(x_0, y_0) > 0$, etc.

Remarque 3 : le type de l'EDP est invariant par changement de base.

III.2 EQUATIONS ELLIPTIQUES

Les équations elliptiques régissent les problèmes stationnaires, d'équilibre, généralement définis sur un domaine spatial borné Ω de frontière Γ sur laquelle l'inconnue est soumise à des conditions aux limites, le plus souvent de type Dirichlet ou Neumann.

Le problème elliptique type est celui fourni par l'équation de Laplace (ou de Poisson) soumise à des conditions aux limites, par exemple de Dirichlet :

$$\begin{cases} -\Delta u = f & \text{dans} \quad \Omega \\ u = u_0 & \text{sur} \quad \Gamma \end{cases}$$

Exemples : Equation de la chaleur en stationnaire (température d'équilibre).
Déplacement vertical d'une membrane dont le bord est fixé.
En mécanique des fluide, dans le cas d'un écoulement plan, permanent d'un fluide parfait incompressible, le potentiel des vitesses vérifient une équation de Laplace.

III.3 EQUATIONS PARABOLIQUES

Les équations paraboliques régissent les problèmes d'évolution ou instationnaires dans lesquels intervient le mécanisme de diffusion ou de dissipation. Ces problèmes sont généralement définis sur un domaine spatial borné Ω de frontière Γ sur laquelle l'inconnue est soumise à des conditions aux limites du même type qu'en elliptique (quelquefois elles-mêmes instationnaires), ainsi qu'à des conditions initiales.

Le problème elliptique type est celui fourni par l'équation de la chaleur soumise à des conditions aux limites, par exemple de Dirichlet, ainsi qu'à des conditions initiales :

$$\begin{cases} \dfrac{\partial T}{\partial t} = \alpha \dfrac{\partial^2 T}{\partial x^2} & \text{dans} \quad \Omega \\ T = T_0 & \text{sur} \quad \Gamma \\ T(x,0) = f(x) & \text{dans} \quad \Omega \end{cases}$$

III.4 EQUATIONS HYPERBOLIQUES

III.4.1 Origine physique

Les équations hyperboliques modélisent la propagation d'ondes sans dissipation.
En linéaire, c'est par exemple la propagation du son dans un milieu homogène. En électromagnétisme, les équations de Maxwell sont hyperboliques et linéaires.

En non linéaire, les équations hyperboliques sont l'expression de lois de conservation. Par exemple, les équations d'Euler expriment la conservation de la masse, de la quantité de mouvement et de l'énergie totale dans un fluide parfait compressible.

III.4.2 Equations types

Pour une grandeur $w(x,t)$ on distingue deux grands types d'équation hyperbolique :
• Le premier type d'équation du cas hyperbolique est l'équation des ondes homogènes :

$$\boxed{\frac{\partial^2 w}{\partial t^2} - c^2 \frac{\partial^2 w}{\partial x^2} = 0}$$

• Le deuxième type du cas hyperbolique conduit à la définition suivante :
Soit A une matrice $n \times n$. Le système (linéaire ou non) du premier ordre

$$\boxed{\frac{\partial w}{\partial t} + A(w) \frac{\partial w}{\partial x} = 0}$$

est hyperbolique ssi la matrice A est diagonalisable à valeurs propres réelles, pour tout w.
Dans le cas scalaire ($n = 1$), le système se ramène à l'équation de convection pure :

$$\frac{\partial w}{\partial t} + c \frac{\partial w}{\partial x} = 0$$

III.4.3 Caractéristiques

III.4.3.1 Caractéristiques pour les équations du premier type

Considérons l'équation hyperbolique suivante :

$$a\frac{\partial^2 u}{\partial x^2} + b\frac{\partial^2 u}{\partial x \partial y} + c\frac{\partial^2 u}{\partial y^2} = 0 \quad \text{avec}\, b^2 - 4ac > 0$$

Cette équation peut s'écrire dans un autre jeu de coordonnées (X, Y) :

$$A\frac{\partial^2 u}{\partial X^2} + B\frac{\partial^2 u}{\partial X \partial Y} + C\frac{\partial^2 u}{\partial Y^2} + D\frac{\partial u}{\partial X} + E\frac{\partial u}{\partial Y} = 0$$

où A, B, C, D et E sont fonctions de a, b, c et des dérivées de X,Y par rapport à x, y.

Dans le but de déterminer une forme canonique de l'EDP, nous pouvons chercher s'il existe des coordonnées (X, Y) telle que $A = 0$ ou $C = 0$, ce qui revient à résoudre l'équation suivante :

$$a\left(\frac{dy}{dx}\right)^2 - b\left(\frac{dy}{dx}\right) + c = 0$$

Nous obtenons ainsi deux équations différentielles ordinaires (EDO) :

$$\boxed{\frac{dy}{dx} = \frac{b + \sqrt{b^2 - 4ac}}{2a} \qquad \text{et} \qquad \frac{dy}{dx} = \frac{b - \sqrt{b^2 - 4ac}}{2a}}$$

Celles-ci sont appelées **les équations caractéristiques**.

En résolvant ces deux équations nous obtenons les courbes caractéristiques. Et les nouvelles coordonnées $(X(x,y), Y(x,y))$ sont les coordonnées caractéristiques. Après ce changement de coordonnées, nous aurons une EDP de la forme :

$$\frac{\partial^2 u}{\partial X \partial Y} + K\frac{\partial u}{\partial X} + K'\frac{\partial u}{\partial Y} = 0$$

Remarque : Dans le cas où les coefficients a, b et c sont constants, les caractéristiques sont des droites.

Exemple : Considérons l'équation $y^2\dfrac{\partial^2 u}{\partial x^2} - x^2\dfrac{\partial^2 u}{\partial y^2} = 0$. Les équations caractéristiques sont :

$$\frac{dy}{dx} = \frac{\sqrt{4x^2y^2}}{2y^2} = \frac{x}{y} \qquad \text{et} \qquad \frac{dy}{dx} = \frac{-\sqrt{4x^2y^2}}{2y^2} = -\frac{x}{y}$$

Nous obtenons ainsi l'équations des deux courbes caractéristiques :

$$\frac{y^2 - x^2}{2} = \text{cte} \qquad \text{et} \qquad \frac{y^2 + x^2}{2} = \text{cte}$$

Les coordonnées caractéristiques sont : $X(x,y) = \dfrac{y^2 - x^2}{2}$ et $Y(x,y) = \dfrac{y^2 + x^2}{2}$
Et la nouvelle expression de l'EDP :

$$\frac{\partial^2 u}{\partial X \partial Y} = \frac{Y}{2(X^2 - Y^2)}\frac{\partial u}{\partial X} - \frac{X}{2(X^2 - Y^2)}\frac{\partial u}{\partial Y}$$

III.4.3.2 Caractéristiques pour l'équation de convection

Considérons l'équation de convection d'une grandeur $w(x,t)$:

$$\frac{\partial w}{\partial t} + c\frac{\partial w}{\partial x} = 0$$

où la vitesse c peut être constante ou non.

Dérivée le long d'une courbe

On introduit la notion de dérivée de la grandeur w par rapport au temps le long d'une courbe $\mathcal{C}$ du plan (x,t). On se limite au cas des courbes décrites par une équation du type $x = X(t)$. Cette notion de dérivée est facile à saisir intuitivement : on regarde la variation de w en suivant la courbe $\mathcal{C}$ et on dérive par rapport au temps.

On définit la dérivée de la grandeur $w(x,t)$ par rapport au temps le long de la courbe $\mathcal{C}$ d'équation $x = X(t)$ par :

$$\left(\frac{dw}{dt}\right)_{\mathcal{C}} = \frac{dw}{dt}(X(t),t) = \frac{\partial w}{\partial t} + \frac{\partial w}{\partial x}\left(\frac{dx}{dt}\right)_{\mathcal{C}}$$

Construction des solutions avec une famille de courbes

A partir de la définition ci-dessus, on constate que si le long de la courbe $\mathcal{C}$, $\left(\frac{dx}{dt}\right)_{\mathcal{C}} = c$, alors la dérivée de w le long de la courbe est nulle : $\left(\frac{dw}{dt}\right)_{\mathcal{C}} = \frac{\partial w}{\partial t} + c\frac{\partial w}{\partial x} = 0$.

On montre ainsi que la résolution de l'EDP de départ se ramène à la résolution d'un système d'EDO. Ce système définit une famille de courbes que l'on nomme courbes caractéristiques. Cette méthode peut aussi s'interpréter comme un changement de variable. Sa validité cesse lorsque les courbes caractéristiques se coupent. Ce cas correspond à l'apparition de singularités dans la solution de l'équation de convection (ondes de choc par exemple).

La méthode des caractéristiques consiste donc à remplacer la résolution de l'EDP par la recherche d'une famille de courbes $\mathcal{C}$ d'équation $x = X(t)$ et d'une famille de fonctions $w_{\mathcal{C}}$ solutions du système d'équations différentielles ordinaires couplées :

$$\left\{ \begin{array}{lcl} \left(\frac{dx}{dt}\right)_{\mathcal{C}} & = & c \\ \left(\frac{dw}{dt}\right)_{\mathcal{C}} & = & 0 \end{array} \right.$$

<u>Exemple</u> : Résolution de l'équation de convection $\frac{\partial w}{\partial t} + c_0\frac{\partial w}{\partial x} = 0$ avec la condition initiale $w(x,0) = w_0(x)$ et une vitesse c_0 constante. Le système d'EDO à résoudre est :

$$\left\{ \begin{array}{lcl} \dot{x} & = & c_0 \\ \dot{w} & = & 0 \end{array} \right.$$

avec les conditions initales $[x(0), w(0)] = [a, w_0(a)]$. On en déduit alors :

$$x = X(t) = a + c_0 t \qquad \text{et} \qquad w_{\mathcal{C}} = w_0(a) = w_0(x - c_0 t)$$

Les courbes caractéristiques sont donc des droites parallèles de pente $1/c_0$ dans le plan (x,t) et la grandeur w est invariante le long de ces droites. On en déduit que $w(x,t) = w_0(x - c_0 t)$. La grandeur w est dans ce cas appelée un invariant de Riemann.

III.4.3.3 Caractéristiques pour un système de lois de conservation

Considérons un système de lois de conservation d'ordre n sous la forme non-conservative :

$$\frac{\partial w}{\partial t} + A(w)\frac{\partial w}{\partial x} = 0$$

où la matrice A est diagonalisable à valeurs propres réelles $\lambda_k(w)$.

Ce système d'EDP peut se transformer en un système d'EDO faisant intervenir la dérivation des solutions le long de courbes caractéristiques dont le tracé dépend lui-même des solutions. Cette transformation permet une interprétation géométrique des solutions dans le plan (x,t) et conduit souvent à des résolutions analytiques ou numériques.

Cette méthode des caractéristiques n'est plus valide lorsque les courbes caractéristiques se coupent. C'est le cas lorsque des discontinuités existent (ondes de choc en Euler). Dans ce cas, il n'y a pas unicité de la solution.

On introduit un vecteur propre à gauche L_k de la matrice A (ie $L_k A = \lambda_k L_k$). On multiplie le système d'EDP par ce vecteur propre. Il vient :

$$L_k \left(\frac{\partial w}{\partial t} + A(w) \frac{\partial w}{\partial x} \right) = L_k \left(\frac{\partial w}{\partial t} + \lambda_k(w) \frac{\partial w}{\partial x} \right) = 0 \tag{III.2}$$

Pour k fixé, on introduit une famille de courbes $\mathcal{C}^k$ définies par l'EDO ;

$$\frac{dx}{dt} = \lambda_k(w)$$

L'équation (III.2) peut alors se réécrire :

$$L_k \left(\frac{\partial w}{\partial t} + \frac{\partial w}{\partial x} \left(\frac{dx}{dt} \right)_{\mathcal{C}^k} \right) = L_k \left(\frac{dw}{dt} \right)_{\mathcal{C}^k} = 0$$

Ou plus simplement :

$$\boxed{L_k \frac{dw}{dt} = 0 \qquad \text{sur la courbe } \mathcal{C}^k} \tag{III.3}$$

Les courbes $\mathcal{C}^k$ constituent **une famille de courbes caractéristiques**. Si w est continue alors il passe une courbe caractéristique et une seule en chaque point du plan (x, t).

En considérant n vecteurs propres linéairement indépendants, on peut former n équations scalaires de la forme (III.3), linéairement indépendantes, et remplacer le système de n EDP par un système de n EDO. Cette simplification du problème s'opère toutefois au prix d'un couplage entre les équations définissant les courbes caractéristiques et les équations (III.3) valables sur ces courbes, puisque les valeurs propres $\lambda_k(w)$ dépendent de l'inconnue w.

Remarque : Une méthode importante permettant de traiter le problème de l'absence d'unicité des solutions dans le cas où apparaissent des discontinuités consiste à introduire un terme de viscosité. Le système à considérer s'écrit :

$$\frac{\partial w}{\partial t} + A(w) \frac{\partial w}{\partial x} = \varepsilon \Delta w$$

où ε est un petit paramètre de viscosité que l'on appelle artificiel (car non physique).
On constate alors que les solutions de ce système sont régulières, les discontinuités ont disparu.

III.4.4 Domaines de dépendance et d'influence

Une propriété fondamentale des problèmes hyperboliques est l'existence de domaines de dépendance et d'influence.

Dans le plan (x, t), pour un point M fixé, il existe un domaine D_M qui influence la solution au point M. Ce domaine, appelé domaine de dépendance du point M, est délimité par les courbes caractéristiques qui passent par M.

Dans le plan (x, t), pour un point P fixé sur l'axe des abscisses, il existe un domaine D_P influencé par la solution au point P. Ce domaine, appelé domaine d'influence du point P, est délimité par les courbes caractéristiques qui partent de P.

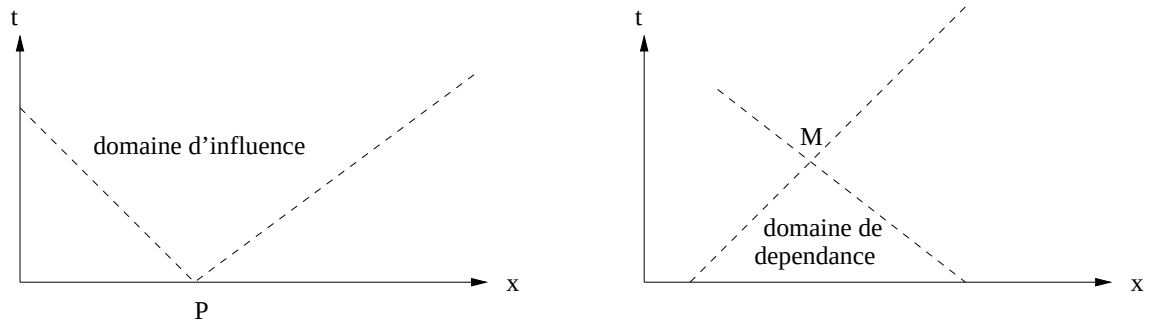

FIG. III.1 – Domaine de dépendance et d'influence

III.4.5 Forme conservative et non-conservative

Considérons un système de lois de conservation d'ordre n écrit pour un vecteur $w(x, t)$:

$$\boxed{\frac{\partial w}{\partial t} + \frac{\partial f(w)}{\partial x} = 0 \qquad \text{avec } w(x, 0) = w_0(x)} \tag{III.4}$$

où f est une fonction non linéaire dite **fonction de flux**. Le système (III.4) est dit **sous forme conservative** ou forme divergence.

Exemple : L'équation de Burgers monodimensionnelle :

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{u^2}{2}\right) = 0 \qquad \text{avec } u(x, 0) = u_0(x)$$

Si $w(, x, t)$ est une solution régulière vérifiant le système (III.4), on peut évaluer le terme de dérivée de la fonction flux en introduisant la matrice jacobienne de f par rapport à w. On obtient alos le système **sous forme non-conservative** :

$$\boxed{\frac{\partial w}{\partial t} + A(w)\frac{\partial w}{\partial x} = 0 \qquad \text{avec } w(x, 0) = w_0(x)}$$

où la matrice $A(w) = f'(w)$ est la jacobienne de f : $A_{ij} = \dfrac{\partial f_i(w)}{\partial w_j}$.
A est diagonalisable pour tout w à valeurs propres réelles.

Exemple : La forme non-conservative de l'équation de Burgers est :

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0 \qquad \text{avec } u(x,0) = u_0(x)$$

III.4.6 Discontinuité - relation de saut

Même dans le cas où la fonction de flux f et la solution initale sont régulières, il peut apparaître des discontinuités et des solutions non régulières (formation de chocs). Les solutions du système ne sont pas nécessairement de classe C^1, on parle alors de **solutions faibles** du système.

Pour une solution faible (discontinue), on peut introduire **des relations de saut** au travers de la discontinuité (par exemple les relations de Rankine-Hugoniot en fluide parfait compressible).

Supposons qu'une solution faible w soit discontinue le long d'une courbe Γ régulière dans le plan (x,t). Cette courbe Γ sépare le plan en deux régions Ω_1 et Ω_2. On note w_1 et w_2 les restrictions de w à Ω_1 et Ω_2 que l'on suppose régulières. On désigne par $\vec{n}$ la normale à Γ dirigée vers l'extérieur de Ω_1 de composante (n_x, n_t). La relation de saut de l'inconnue w et de la fonction de flux au travers de la courbe Γ s'écrit :

$$(w_1 - w_2)n_t + (f(w_1) - f(w_2))\, n_x = 0$$

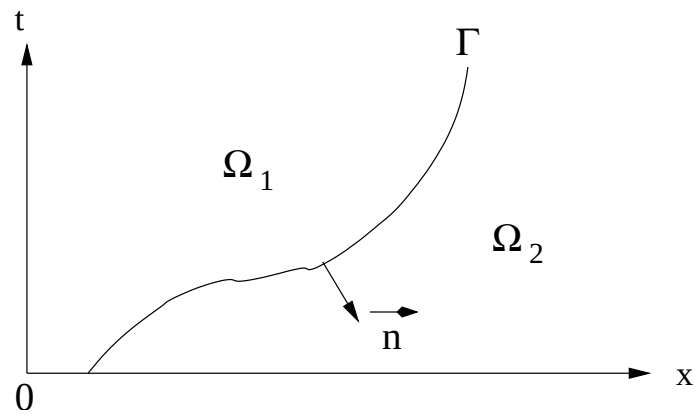

FIG. III.2 – Solution discontinue le long d'une courbe

Chapitre IV

RESOLUTION DES EDO

Les équations différentielles ordinaires (EDO) apparaissent très souvent dans la modélisation de la physique et des sciences de l'ingénieur. Trouver la solution d'une EDO ou d'un système d'EDO est ainsi un problème courant, souvent difficile ou impossible à résoudre de façon analytique. Il est alors nécessaire de recourir à des méthodes numériques pour résoudre ces équations différentielles.

IV.1 DEFINITION DES EDO

Soit une fonction $y(x)$ définie sur un intervalle de $\mathcal{R}$ et de classe C_p (continûment dérivable d'ordre p). On appelle équation différentielle d'ordre p une équation de la forme :

$$F(x, y, y', y'', ..., y^{(p)}) \; = \; 0$$

On appelle forme canonique d'une EDO une expression du type :

$$y^{(p)} \; = \; f(x, y, y', y'', ..., y^{(p-1)})$$

Seul ce type d'équations sera considéré dans ce chapitre.

Toute équation différentielle canonique peut être écrite comme un système d'équations différentielles du premier ordre en introduisant $p-1$ fonctions définies comme :

$$\begin{cases} y_1 & = & y \\ y_2 & = & y' \\ \dots & & \dots \\ y_p & = & y^{(p-1)} \end{cases}$$

L'équation canonique se met sous la forme du système d'EDO d'ordre 1 suivant :

$$\begin{cases} y_1' & = & y_2 \\ y_2' & = & y_3 \\ \dots & & \dots \\ y_p' & = & f(x, y, y_1, y_2, ..., y_p) \end{cases}$$

IV.2 RAPPEL - SOLUTIONS D'EDO SIMPLES

Nous rappelons ici les principes de résolution des équations différentielles simples. Dans ce qui suit, la variable indépendante sera notée t et la variable dépendante $x(t)$.

ORDRE 1	SOLUTION GENERALE
$\dot{x} = ax + b$	$x(t) = Ce^{at} - \dfrac{b}{a}$
$\dot{x} = a(t)x + b(t)$	$x(t) = e^{A(t)}\left(B(t) + C\right)$ avec $A(t) = \int a(t)\,dt$ et $B(t) = \int \exp\left(-A(t)\right)b(t)\,dt$
$t\dot{x} - x = f(\dot{x})$	$x(t) = Ct + g(C)$
ORDRE 2	
équation de Bernoulli $\dot{x} + a(t)x + b(t)x^{\alpha} = 0$	effectuer le changement de fonction $x(t) = z(t)^{\frac{1}{1-\alpha}}$ l'équation devient : $\dot{z} = (\alpha - 1)\left(a(t)z + b(t)\right)$
équation de Ricatti $\dot{x} + a(t)x + b(t)x^2 + c(t) = 0$	effectuer le changement de fonction $z(t) = x(t) - x_p$ où x_p est une solution particulière de l'équation on se ramène à une équation de Bernoulli avec $\alpha = 2$
$\ddot{x} + a\dot{x} + bx = c$	r_1 et r_2 racines de l'équation caractéristique $r^2 + ar + b = 0$ $x(t) = C_1e^{r_1t} + C_2e^{r_2t} + c/b$ si $r = s + ip$ alors $x(t) = e^{st}\left(D_1cos(pt) + D_2sin(pt)\right) + c/b$
équation d'Euler $t^2\ddot{x} + at\dot{x} + bx = c(t)$	changements de variable $t = e^u$ et de fonction $x(t) = z(u)$ l'équation devient : $z'' + (a-1)z' + bz = c(e^u)$
ORDRE 3	
$\dddot{x} + a\ddot{x} + b\dot{x} + cx = 0$	r_1, r_2, r_3 racines de l'équation caractéristique $r^3 + ar^2 + br + c = 0$ $x(t) = C_1e^{r_1t} + C_2e^{r_2t} + C_3e^{r_3t}$

IV.3 LE PROBLEME DE CAUCHY

On appelle problème de *Cauchy* ou problème à la valeur initiale le problème qui consiste à trouver une fonction $y(x)$ définie sur l'intervalle $[a, b]$ telle que :

$$\begin{cases} y'(x) & = & f(x, y(x)) & ; & \forall x \in [a, b] \\ y(a) & = & y_0 \end{cases}$$

Si la fonction f est continue et vérifie une condition de Lipschitz par rapport à la deuxième variable alors le problème admet une solution unique. On dit que le problème est bien posé.

<u>Attention</u> : un problème bien posé ne signifie pas qu'il peut se résoudre numériquement !

IV.4 PRINCIPE GENERAL DES METHODES NUMERIQUES

Pour obtenir une approximation numérique de la solution $y(x)$ sur l'intervalle $[a, b]$, nous allons estimer la valeur de cette fonction en un nombre fini de points x_i, pour $i = 0, 1, ..., n$, constituants les noeuds du maillage. La solution numérique discrète obtenue aux points x_i est notée $y_i = y(x_i)$.

L'écart entre deux abscisses, noté h, est appelé *pas de discrétisation*. Ce pas, dans les méthodes les plus simples, est constant, mais il peut être judicieux de travailler avec un pas variable $h_i = x_i - x_{i-1}$. Le choix du maillage et de la répartitiondes noeuds peuvent s'avérer crucial.

Les techniques de résolution des EDO sont basées sur :

- l'approximation géométrique de la fonction
- les formules d'intégration numérique (rectangle, trapèze, Simpson...)
- les développements de Taylor au voisinage de x_i

IV.5 PROPRIETES DES METHODES NUMERIQUES

Plusieurs notions mathématiques sont introduites lors de la résolution d'EDO au moyen de leurs équivalents discrétisés. Les trois principales sont **la convergence**, **la stabilité** et **la consistance** (cf. cours sur la discrétisation des EDP), permettant de relier la solution exacte des équations continues à la solution exacte des équations discrétisées et à la solution numérique obtenue.

A ces propriétés, il convient d'ajouter la notion de précision ainsi que des aspects informatiques comme la facilité de mise en oeuvre, les coûts CPU et mémoire.

Consistance d'une méthode

La consistance est la propriété qui assure que la solution exacte de l'équation discrétisée tende vers la solution exacte de l'équation continue lorsque le pas de discrétisation h tend vers 0.

Stabilité d'une méthode

C'est la propriété qui assure que la différence entre la solution numérique obtenue et la solution exacte des équations discrétisées reste bornée. Le stabilité indique si l'erreur augmente ou non au cours du calcul.
Une méthode peut être stable sous condition (elle sera dite conditionnellement stable) ou toujours stable (elle sera dite inconditionnellement stable).

Ordre de précision d'une méthode

L'erreur de troncature ε est définie comme la différence entre la solution exacte $\tilde{y}$ et l'approximation numérique obtenue y_n, soit : $\varepsilon_n = \mid \tilde{y}(x_n) - y_n \mid = \mathcal{O}(h^p)$. L'ordre de précision de la méthode est donnée par l'entier p.

Convergence et taux de convergence d'une méthode

Une méthode est convergente si, lorsque le pas de discrétisation tend vers 0, la solution numérique tend vers la solution exacte de l'équation continue.
Une méthode est convergente à l'ordre l ssi : $\lim\limits_{n\to\infty} \max\limits_{i} \mid \varepsilon_i \mid = \mathcal{O}(h^l)$

Résultat théorique : Une méthode stable et consistante est convergente.

IV.6 LES PRINCIPALES METHODES NUMERIQUES

Les principales méthodes de résolution numérique des EDO sont séparées en deux grands types :

- les méthodes à un pas

  Pour ces méthodes, le calcul de la valeur discrète y_{n+1} au noeud x_{n+1} fait intervenir la valeur y_n obtenue à l'abscisse précédente. Les principales méthodes sont :
  - Méthodes d'Euler explicite et implicite
  - Méthode d'Euler amélioré
  - Méthode d'Euler-Cauchy
  - Méthode de Crank-Nicholson
  - Méthodes de Runge et Kutta

- les méthodes à pas multiples

  Pour ces méthodes, le calcul de la valeur discrète y_{n+1} au noeud x_{n+1} fait intervenir plusieurs valeurs $y_n, y_{n-1}, y_{n-2}, ...$ obtenues aux abscisses précédentes. Les principales méthodes sont :
  - Méthode de Nystrom ou saute-mouton
  - Méthodes d'Adams-Bashforth-Moulton
  - Méthodes de Gear

IV.7 METHODES A UN PAS

La formulation générale des méthodes à un pas explicite est :

$$\begin{cases} & y_0 \text{ donné} \\ y_{n+1} & = y_n + \phi(x_n, y_n, h) \end{cases}$$

où la fonction ϕ définit la méthode utilisée.

La formulation générale des méthodes à un pas implicite est :

$$\begin{cases} & y_0 \text{ donné} \\ y_{n+1} & = y_n + \phi(x_n, y_n, y_{n+1}, h) \end{cases}$$

L'obtention de la solution à chaque abscisse nécessite la résolution d'une équation.

Ces méthodes sont obtenues en intégrant l'équation différentielle et en utilisant des formules d'intégration numérique pour le second membre. L'ordre du schéma est égal au degré du polynôme pour lequel l'intégration est exacte + 1.

Résultats théoriques :
1) Si la fonction ϕ est lipschitzienne par rapport à la deuxième variable alors les méthodes explicites sont stables.
2) Les méthodes implicites sont toujours stables.
3) Les méthodes sont consistantes ssi $\forall x \in [a, b]$, $\phi(x, y, 0) = f(x, y)$.

IV.7.1 Méthodes d'Euler explicite et implicite

Méthode explicite, d'ordre 1, dont l'algorithme est :

$$\begin{cases} & y_0 \text{ donné} \\ y_{n+1} & = y_n + hf(x_n, y_n) \end{cases}$$

La méthode peut s'interpréter de plusieurs manières :
1) Via les formules d'intégration numérique : la méthode est le résultat de l'application de la formule des rectangles basée au point x_n.
2) Géométriquement : la méthode revient à remplacer localement en chaque point x_n la courbe solution par sa tangente.
3) Via les développements de Taylor : la méthode provient du développement de Taylor d'ordre 1 de la fonction y au voisinage de x_n.

Méthode implicite, d'ordre 1, dont l'algorithme est :

$$\begin{cases} & y_0 \text{ donné} \\ y_{n+1} & = y_n + hf(x_{n+1}, y_{n+1}) \end{cases}$$

IV.7.2 Méthode d'Euler amélioré

Méthode explicite dont l'algorithme est :

$$\left\{\begin{array}{lcl} & y_0 & \text{donné} \\ y^*_{n+1} & = & y_n + \dfrac{h}{2} f(x_n, y_n) \\ y_{n+1} & = & y_n + hf(x_n + \dfrac{h}{2}\ ,\ y^*_{n+1}) \end{array}\right.$$

Géométriquement, la méthode consiste à remplacer dans la méthode d'Euler la pente de la tangente en (x_n, y_n) par la valeur corrigée au milieu de l'intervcalle $[x_n, x_{n+1}]$.

IV.7.3 Méthode d'Euler-Cauchy

Méthode explicite dont l'algorithme est :

$$\left\{\begin{array}{lcl} & y_0 & \text{donné} \\ y^*_{n+1} & = & y_n + hf(x_n, y_n) \\ y_{n+1} & = & y_n + \dfrac{h}{2}\left[f(x_n, y_n) + f(x_{n+1}, y^*_{n+1})\right] \end{array}\right.$$

Géométriquement, la méthode consiste à remplacer dans la méthode d'Euler la pente de la tangente en (x_n, y_n) par la moyenne de cette pente avec la valeur corrigée en x_{n+1}.

IV.7.4 Méthode de Crank-Nicholson

Méthode implicite, d'ordre 2, dont l'algorithme est :

$$\left\{\begin{array}{lcl} & y_0 & \text{donné} \\ y_{n+1} & = & y_n + \dfrac{h}{2}\left[f(x_n, y_n) + f(x_{n+1}, y_{n+1})\right] \end{array}\right.$$

Elle est obtenue en utilisant la formule d'intégration numérique des trapèzes.

IV.7.5 Méthodes de Runge et Kutta

Ce sont des méthodes d'ordre élevé, obtenues à partir des formules d'intégration numérique plus précises que la formule des rectangles.

Une méthode explicite, d'ordre 2, peut être obtenue par l'utilisation de la formule des trapèzes. L'algorithme, noté RK2, s'écrit :

$$\left\{\begin{array}{lcl} & y_0 & \text{donné} \\ y^*_{n+1} & = & y_n + hf(x_n, y_n) \\ y_{n+1} & = & y_n + \dfrac{h}{2}\left[f(x_n, y_n) + f(x_{n+1}, y^*_{n+1})\right] \end{array}\right.$$

Une méthode explicite, d'ordre 4, peut être obtenue par l'utilisation de la formule de Simpson. L'algorithme, noté RK4, s'écrit :

$$\left\{\begin{array}{rcl} & y_0 & \text{donné} \\ k_1 & = & hf(x_n, y_n) \\ k_2 & = & hf\left(x_n + \dfrac{h}{2}\ ,\ y_n + \dfrac{k_1}{2}\right) \\ k_3 & = & hf\left(x_n + \dfrac{h}{2}\ ,\ y_n + \dfrac{k_2}{2}\right) \\ k_4 & = & hf(x_n + h\ ,\ y_n + k_3) \\ y_{n+1} & = & y_n + \dfrac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \end{array}\right.$$

IV.7.5.1 Forme générale des méthodes de Runge et Kutta

On cherche à résoudre le problème de Cauchy suivant :

$$\left\{\begin{array}{rcl} y'(x) & = & f\left(x, y(x)\right) \quad ; \quad \forall x \in [0,1] \\ y(0) & = & y_0 \end{array}\right.$$

Introduisons q points intermédiaires dans chaque intervalle $[x_n, x_{n+1}]$ notés $x_{n,1}, x_{n,2}, ..., x_{n,q}$. On se donne $c_1, c_2, ..., c_q$ réels dans l'intervalle $[0,1]$ et on pose $x_{n,i} = x_n + c_i h$ pour $i = 1$ à q.

La valeur discrète $y_{x_{n,i}}$ vérifie :

$$\begin{array}{rcl} y_{x_{n,i}} & = & y(x_n) + \displaystyle\int_{x_n}^{x_{n,i}} f(t, x(t))\, dt \\ & = & y(x_n) + h\displaystyle\int_0^{c_i} f\left(x_n + \tau h\ ,\ y(x_n + \tau h)\right)\, d\tau \\ y_{x_{n+1}} & = & y(x_n) + h\displaystyle\int_0^{1} f\left(x_n + \tau h\ ,\ y(x_n + \tau h)\right)\, d\tau \end{array}$$

On choisit $q+1$ méthodes d'intégration à q points pour approximer ces intégrales. Ce qui revient à se donner $q(q+1)$ paramètres $(a_{ij})_{1\leq i,j\leq q}$ et $(b_i)_{1\leq i\leq q}$ tels que :

$$\begin{array}{rcl} \displaystyle\int_0^{c_i} f\left(x_n + \tau h, y(x_n + \tau h)\right)\, d\tau & \rightarrow & \displaystyle\sum_{j=1}^{q} a_{ij} f\left(x_n + c_j h\ ,\ y(x_n + c_j h)\right) \\ \displaystyle\int_0^{1} f\left(x_n + \tau h, y(x_n + \tau h)\right)\, d\tau & \rightarrow & \displaystyle\sum_{i=1}^{q} b_i f\left(x_n + c_i h\ ,\ y(x_n + c_i h)\right) \end{array}$$

Au bilan la formulation de la méthode est :

$$\left\{\begin{array}{rcl} & y_0 & \text{donné} \\ y_{n,i} & = & y_n + h\displaystyle\sum_{j=1}^{q} a_{ij} f(x_{n,j}, y_{n,j}) \\ y_{n+1} & = & y_n + h\displaystyle\sum_{i=1}^{q} b_i f(x_{n,i}, y_{n,i}) \end{array}\right.$$

La méthode se caractérise par les paramètres (a_{ij}), (b_i) et (c_i). On construite le tableau qui rassemble ces paramètres :

$$\begin{array}{c|cccc}
c_1 & a_{11} & a_{12} & \cdots & a_{1q} \\
c_1 & a_{21} & a_{22} & \cdots & a_{2q} \\
\vdots & \vdots & & & \vdots \\
c_q & a_{q1} & a_{q2} & \vdots & a_{qq} \\
\hline
 & b_1 & b_2 & \cdots & b_q
\end{array}$$

Par exemple, les paramètres de deux méthodes d'ordre 4 (dont l'algorithme RK4) :

0	0	0	0	0
1/3	1/3	0	0	0
2/3	-1/3	1	0	0
1	1	-1	1	0
	1/8	3/6	3/6	1/8

0	0	0	0	0
1/2	1/2	0	0	0
1/2	0	1/2	0	0
1	0	0	1	0
	1/6	2/6	2/6	1/6

<u>Résultat théorique</u> : Si $\sum_{i=1}^{q} b_i = 1$ alors la méthode est consistante.

IV.7.5.2 Méthodes de Runge et Kutta implicites

Des méthodes de Runge et Kutta implicites (ou méthodes de Radau) peuvent aussi être construites à partir des techniques d'intégration numériques.

Une méthode implicite, d'ordre 3, peut être obtenue :

$$\begin{cases}
y_0 & \text{donné} \\
k_1 & = & hf\left(x_n + \dfrac{h}{3}\ ,\ y_n + \dfrac{h}{12}(5k_1 - k_2)\right) \\
k_2 & = & hf\left(x_n + h\ ,\ y_n + \dfrac{h}{4}(3k_1 + k_2)\right) \\
y_{n+1} & = & y_n + \dfrac{1}{4}(3k_1 + k_2)
\end{cases}$$

Il est nécessaire de résoudre un système linéaire pour évaluer k_1 et k_2.

Par exemple, les paramètres (a_{ij}), (b_i) et (c_i) pour les méthodes d'ordre 3 et 5 sont :

1/3	5/12	-1/12
1	3/4	1/4
	3/4	1/4

$$\begin{array}{c|ccc}
\frac{4-\sqrt{6}}{10} & \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\
\frac{4-\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\
1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \\
\hline
 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9}
\end{array}$$

IV.7.5.3 Application à un système

Considérons le systèmes d'EDO aux valeurs initales suivant :

$$\begin{cases} y'(x) &= f\left(x, y(x), z(x)\right) \quad ; \quad x, z \in [a, b] \\ z'(x) &= g\left(x, y(x), z(x)\right) \\ y(a) &= y_0 \quad \text{et} \quad z(a) = z_0 \end{cases}$$

L'algorihme de Runge-Kutta RK2 s'écrit, pour ce système :

$$\begin{cases} & y_0, z_0 \text{ donnés} \\ y^*_{n+1} &= y_n + hf(x_n, y_n, z_n) \\ z^*_{n+1} &= z_n + hg(x_n, y_n, z_n) \\ y_{n+1} &= y_n + \dfrac{h}{2}\left(f(x_n, y_n, z_n) + f(x_n, y^*_{n+1}, z^*_{n+1})\right) \\ z_{n+1} &= z_n + \dfrac{h}{2}\left(g(x_n, y_n, z_n) + g(x_n, y^*_{n+1}, z^*_{n+1})\right) \end{cases}$$

L'algorihme de Runge-Kutta RK4 s'écrit, pour ce système :

$$\begin{cases} & y_0, z_0 \text{ donnés} \\ k_1 &= hf(x_n, y_n, z_n) & l_1 &= hg(x_n, y_n, z_n) \\ k_2 &= hf\left(x_n + \dfrac{h}{2}\ ,\ y_n + \dfrac{k_1}{2}\ ,\ z_n + \dfrac{l_1}{2}\right) & l_2 &= hg\left(x_n + \dfrac{h}{2}\ ,\ y_n + \dfrac{k_1}{2}\ ,\ z_n + \dfrac{l_1}{2}\right) \\ k_3 &= hf\left(x_n + \dfrac{h}{2}\ ,\ y_n + \dfrac{k_2}{2}\ ,\ z_n + \dfrac{l_2}{2}\right) & l_3 &= hg\left(x_n + \dfrac{h}{2}\ ,\ y_n + \dfrac{k_2}{2}\ ,\ z_n + \dfrac{l_2}{2}\right) \\ k_4 &= hf(x_n + h\ ,\ y_n + k_3\ ,\ z_n + l_3) & l_4 &= hg(x_n + h\ ,\ y_n + k_3\ ,\ z_n + l_3) \\ y_{n+1} &= y_n + \dfrac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) & z_{n+1} &= z_n + \dfrac{1}{6}\left(l_1 + 2l_2 + 2l_3 + l_4\right) \end{cases}$$

IV.8 METHODES A PAS MULTIPLES

IV.8.1 Méthode de Nystrom ou saute-mouton

Méthode à 2 pas dont l'algorithme est :

$$\begin{cases} y_0 \text{ donné} \\ y_1 \text{ calculé avec une méthode à un pas} \\ y_{n+1} = y_{n-1} + 2hf(x_n, y_n) \end{cases}$$

Géométriquement : on considère la droite de pente $f(x_n, y_n)$ passant par le point (x_{n-1}, y_{n-1}) parallèle à la tangente passant par (x_n, y_n). La valeur y_{n+1} est l'ordonnée du point de cette droite d'abscisse x_{n+1}.

IV.8.2 Méthodes d'Adams-Bashforth-Moulton

Méthodes basées sur des techniques d'intégration numérique, dont la formulation générale est :

$$y_{n+1} = y_n + h \sum_{j=-1}^{p} \beta_j f(x_{n-j}, y_{n-j})$$

Méthode d'Adams-Bashforth à 2 pas, explicite, d'ordre 2 :

$$\begin{cases} y_0 \text{ donné} \\ y_1 \text{ calculé avec une méthode à un pas} \\ y_{n+1} = y_n + \dfrac{h}{2}\left(3f(x_n, y_n) - f(x_{n-1}, y_{n-1})\right) \end{cases}$$

Méthode d'Adams-Bashforth à 3 pas, explicite, d'ordre 3 :

$$\begin{cases} y_0 \text{ donné} \\ y_1, y_2 \text{ calculés avec une méthode à un pas} \\ y_{n+1} = y_n + \dfrac{h}{12}\left(23f(x_n, y_n) - 16f(x_{n-1}, y_{n-1}) + 5f(x_{n-2}, y_{n-2})\right) \end{cases}$$

Méthode d'Adams-Bashforth à 4 pas, explicite, d'ordre 4 :

$$\begin{cases} y_0 \text{ donné} \\ y_1, y_2, y_3 \text{ calculés avec une méthode à un pas} \\ y_{n+1} = y_n + \dfrac{h}{24}\left(55f(x_n, y_n) - 59f(x_{n-1}, y_{n-1}) + 37f(x_{n-2}, y_{n-2}) - 9f(x_{n-3}, y_{n-3})\right) \end{cases}$$

Méthode d'Adams-Moulton à 1 pas, implicite, d'ordre 2 :

$$\begin{cases} y_0 \text{ donné} \\ y_{n+1} = y_n + \dfrac{h}{2}\left(f(x_n, y_n) + f(x_{n+1}, y_{n+1})\right) \end{cases}$$

Méthode d'Adams-Moulton à 2 pas, implicite, d'ordre 3 :

$$\begin{cases} y_0 \text{ donné} \\ y_1 \text{ calculé avec une méthode à un pas} \\ y_{n+1} = y_n + \dfrac{h}{12}\left(5f(x_{n+1}, y_{n+1}) + 8f(x_n, y_n) - f(x_{n-1}, y_{n-1})\right) \end{cases}$$

Méthode d'Adams-Moulton à 3 pas, implicite, d'ordre 4 :

$$\begin{cases} y_0 \text{ donné} \\ y_1, y_2 \text{ calculé avec une méthode à un pas} \\ y_{n+1} = y_n + \dfrac{h}{24}\left(9f(x_{n+1}, y_{n+1}) + 19f(x_n, y_n) - 5f(x_{n-1}, y_{n-1}) + f(x_{n-2}, y_{n-2})\right) \end{cases}$$

Pour rendre les méthodes d'Adams-Moulton explicite, on remplace le y_{n+1} "génant" par son estimation par la méthode d'Adams-Bashforth. Sont construits ainsi des schémas explicites dits prédicteur-correcteur, par exemple un schéma d'ordre 2 :

$$\begin{cases} y_0 & \text{donné} \\ y_1 & \text{calculé avec une méthode à un pas} \\ y^*_{n+1} & = y_n + \dfrac{h}{24}\left(3f(x_n, y_n) - f(x_{n-1}, y_{n-1})\right) \\ y_{n+1} & = y_n + \dfrac{h}{2}\left(f(x_n, y_n) + f(x_{n+1}, y^*_{n+1})\right) \end{cases}$$

Le schéma prédicteur-correcteur d'ordre 4 est :

$$\begin{cases} y_0 & \text{donné} \\ y_1, y_2 & \text{calculé avec une méthode à un pas} \\ y^*_{n+1} & = y_n + \dfrac{h}{24}\left(55f(x_n, y_n) - 59f(x_{n-1}, y_{n-1}) + 37f(x_{n-2}, y_{n-2}) - 9f(x_{n-3}, y_{n-3})\right) \\ y_{n+1} & = y_n + \dfrac{h}{24}\left(9f(x_{n+1}, y^*_{n+1}) + 19f(x_n, y_n) - 5f(x_{n-1}, y_{n-1}) + f(x_{n-2}, y_{n-2})\right) \end{cases}$$

IV.8.3 Méthodes de Gear

Les méthodes de Gear ne sont pas construites à partir de techniques d'intégration numérique mais directement à partir de polynôme d'interpolation passant par p points (x_{i+1}, y_{i+1}) , (x_i, y_i),..., (x_{i-p+1}, y_{i+1}).

Méthode de Gear à 2 pas, implicite, d'ordre 2 :

$$\begin{cases} y_0 \text{ donné} \\ y_1 \text{ calculé avec une méthode à un pas} \\ y_{n+1} = \frac{4}{3}y_n - \frac{1}{3}y_{n-1} + \frac{2h}{3}f(x_{n+1}, y_{n+1}) \end{cases}$$

Méthode de Gear à 3 pas, implicite, d'ordre 3 :

$$\begin{cases} y_0 \text{ donné} \\ y_1, y_2 \text{ calculé avec une méthode à un pas} \\ y_{n+1} = \frac{18}{11}y_n - \frac{9}{11}y_{n-1} + \frac{2}{11}y_{n-2} + \frac{2h}{11}f(x_{n+1}, y_{n+1}) \end{cases}$$

Méthode de Gear à 4 pas, implicite, d'ordre 4 :

$$\begin{cases} y_0 \text{ donné} \\ y_1, y_2, y_3 \text{ calculé avec une méthode à un pas} \\ y_{n+1} = \frac{48}{25}y_n - \frac{36}{25}y_{n-1} + \frac{16}{25}y_{n-2} - \frac{3}{25}y_{n-3} + \frac{12h}{25}f(x_{n+1}, y_{n+1}) \end{cases}$$

IV.9 LES DIFFERENCES FINIES

Il est possible de résoudre une EDO ou un système d'EDO en utilisant une discrétisation de type différences finies (cf. cours sur la discrétisation des EDP).
Un maillage du domaine de définition de la fonction inconnue est construit. Les dérivées de la fonction inconnue sont approximées par un schéma aux différences finies ce qui permet d'obtenir une relation de récurrence sur les inconnues discrètes.

IV.10 CONDITION DE STABILITE

Considérons un problème à la valeur initiale :

$$\begin{cases} y'(x) &= f\left(x, y(x)\right) \quad ; \quad x \in [0, T] \\ y(0) &= y_0 \end{cases}$$

Si $\dfrac{\partial f}{\partial y} < 0$, il existe un pas de discrétisation seuil h_{max} à partir duquel une méthode explicite sera stable. La condition de stabilité s'écrit :

$$h \, < \, h_{max} = \frac{2}{\max \mid \frac{\partial f}{\partial y} \mid}$$

Si $\dfrac{\partial f}{\partial y} < 0$, les methodes explicites sont instables quelque soit le pas de discrétisation h. Les méthodes sont dites inconditionnellement instables.

Dans certains cas, la dérivée $\dfrac{\partial f}{\partial y}$ peut changer de signe. Le comportement de la méthode est instable dans certains intervalles et stable dans d'autres.

Cas particulier d'une EDO linéaire

Dans le cas particulier d'une EDO linéaire, la solution numérique satisfait une équation récurrente linéaire à $p+1$ niveaux :

$$a_p y_{n+p} + a_{p-1} y_{n+p-1} + ... + a_0 y_n \, = \, b_n$$

Soient r_1, r_2, ..., r_p les racines complexes de l'équation caractéristique associée :

$$a_p r^p + a_{p-1} r^{p-1} + ... + a_1 r + a_0 \, = \, 0$$

La condition de stabilité s'écrit : $\forall i, \mid r_i \mid < 1$

Chapitre V

RESOLUTION DES SYSTEMES LINEAIRES

V.1 INTRODUCTION

Considérons un système d'équations linéaires de la forme $AX = B$ avec A une matrice inversible de dimension $n \times n$, B un vecteur connu et X le vecteur des inconnus.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

Il existe deux grandes familles de méthodes de résolution :

- **Les méthodes directes** qui permettent de résoudre le système soit par triangularisation ou soit par factorisation de la matrice A. Les principales méthodes sont :
  - Le pivot de Gauss
  - La factorisation LU
  - La factorisation de Cholesky
  - Les factorisations de Householder et QR

  Ces méthodes sont utilisées pour les matrices pleines et les petits systèmes (n peu elevé).

- **Les méthodes itératives** qui introduisent une notion de convergence vers la solution. Les principales méthodes sont :
  - Méthode de Jacobi
  - Méthode de Gauss-Seidel (avec ou sans relaxation)
  - Méthode du gradient conjugué (avec ou sans précondionnement)

  Ces méthodes sont utilisées pour les matrices creuses et les grands systèmes.

V.2 PIVOT DE GAUSS

La méthode d'élimination de Gauss a pour but de transformer le système de départ en un système ayant la même solution de la forme $UX = B'$ où U est une matrice triangulaire supérieure et B' un vecteur. La résolution est en deux étapes :

- Triangularisation de la matrice A.
- Résolution du système triangulaire en cascade.

V.2.1 Triangularisation de Gauss

Posons $A^{(1)} = A$, $B^{(1)} = B$ et introduisons le multiplicateur $l_i^1 = \dfrac{a_{i1}^{(1)}}{a_{11}^{(1)}}$. L'inconnue x_1 peut s'éliminer des lignes $i = 2, ..., n$ en leur retranchant l_i^1 fois la première ligne. En faisant, la même chose pour le membre de droite, on définit :

$$\begin{aligned} a_{ij}^{(2)} &= a_{ij}^{(1)} - l_i^1 a_{1j}^{(1)} \quad \text{pour } i, j = 2, ..., n \\ b_i^{(2)} &= b_i^{(1)} - l_i^1 b_1^{(1)} \end{aligned}$$

Un nouveau système est obtenu, de la forme $A^{(2)}X = B^{(2)}$, équivalent au système de départ :

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \cdots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(2)} \end{bmatrix}$$

Ce système peut à nouveau être transformé de façon à éliminer l'inconnue x_2 des lignes $3, ..., n$. En pousuivant ainsi, on obtient une suite de système equivalents : $A^{(k)}X = B^{(k)}$. Pour $k = n$, on aboutit au système triangulaire supérieur $A^{(n)}X = B^{(n)}$:

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n-1}^{(1)} & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n-1}^{(2)} & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & a_{nn}^{(n)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(3)} \\ \vdots \\ b_n^{(n)} \end{bmatrix}$$

On note U (upper) la matrice triangulaire supérieure $A^{(n)}$. Les termes $a_{kk}^{(1)}$ sont appelés *pivots* et doivent être évidemment non nuls.
Pour passer du k-ième système au $k+1$-ième système, les formules sont :

$$\begin{aligned} l_i^k &= \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} \qquad \text{pour } i = k+1, ..., n \\ a_{ij}^{(k+1)} &= a_{ij}^{(k)} - l_i^k a_{kj}^{(k)} \quad \text{pour } i, j = k+1, ..., n \\ b_i^{(k+1)} &= b_i^{(k)} - l_i^k b_k^{(k)} \quad \text{pour } i = k+1, ..., n \end{aligned}$$

Après triangularisation, le système $UX = B'$ se résout en cascade en commençant par x_n.

V.2.2 Coût de la méthode

Pour effectuer l'élimination de Gauss, $2(n-1)n(n+1)/3 + n(n-1)$ opérations ou flops (pour floating operations) sont nécessaires, auxquels il faut ajouter n^2 flops pour la résolution par remontée du système triangulaire. Pour n grand, la méthode requiert environ $2n^3/3$ flops.

V.2.3 Pivot nul et choix du pivot

La méthode de Gauss n'est correctement définie que si les pivots $a_{kk}^{(k)}$ sont non nuls pour $k = 1, ..., n-1$. Si le terme diagonal $a_{kk}^{(k)}$ est nul, il faut choisir un autre élément non nul de la colonne k pour calculer le multiplicateur l_i^k en permutant l'ordre des lignes de la matrice.

Attention aussi au choix du pivot, de grosses erreurs peuvent en découler. Si le terme diagonal a_{kk} est très petit, il faut choisir un autre terme pour évaluer le pivot l_i^k en permutant l'ordre des lignes de la matrice.

V.3 FACTORISATION LU

La factorisation LU (Lower Upper) pour une matrice A inversible consiste à déterminer deux matrices triangulaires, l'une inférieure L et l'autre supérieure U telles que $A = L \times U$.
Cette décomposition, **unique**, existe ssi les n mineurs de A sont non nuls.

La matrice U est obtenue par la méthode d'élimination de Gauss. Et la matrice L fait intervenir les pivots successifs de l'algorithme de Gauss l_i^k :

$$L = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ l_2^1 & 1 & 0 & \cdots & 0 \\ \vdots & \dots & \ddots & \ddots & \vdots \\ l_{n-1}^1 & l_{n-1}^2 & \cdots & 1 & 0 \\ l_n^1 & l_n^2 & \cdots & l_n^{n-1} & 1 \end{bmatrix}$$

La factorisation LU est utile lors de la résolution d'une suite de systèmes de même matrice A où seul le vecteur B change (exemple : calcul d'une même structure soumise à différents cas de charge).

Une fois calculée L et U, résoudre le système de départ consiste à résoudre successivement les deux systèmes triangulaires $LY = B$ puis $UX = Y$.

V.4 FACTORISATION DE CHOLESKY

Méthode de factorisation pour une matrice A **définie positive** et **symétrique**.
Il existe une **unique** matrice R triangulaire inférieure dont les termes diagonaux sont strictement positifs telle que $A = R \times^t R$. Les éléments de R sont, pour $i = 1$ à n :

$$\begin{cases} r_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} r_{ik}^2} \\ r_{ji} = \dfrac{1}{r_{ii}} \left(a_{ij} - \sum_{k=1}^{i-1} r_{ik} r_{jk} \right) \qquad \text{; pour j=i+1 à n} \end{cases}$$

Le système à résoudre $AX = B$ se ramène alors à la résolution de deux systèmes triangulaires :

$$RY = B \qquad \text{et} \qquad {}^tRX = Y$$

Coût de la méthode pour n grand $\rightarrow n^3/3$ flops.

V.5 FACTORISATIONS DE HOUSEHOLDER ET QR

V.5.1 Transformation de Householder

Soit v un vecteur quelconque de composante v_i. Soit le vecteur u de composantes u_i telles que :

$$u_1 = v_1 + \| v \| \qquad \text{et} \qquad u_i = v_i, \text{ pour } i = 2, ..., n.$$

On appelle matrice de Householder, la matrice H symétrique et orthogonale, définie par :

$$H = I_n - 2 \frac{u \times^t u}{{}^tu.u}$$

La multiplication de H par v transforme le vecteur v en un vecteur dont toutes les composantes sont nulles sauf la première :

$$H \times v = v - 2 \frac{u \times^t u}{{}^tu.u} v = \begin{bmatrix} \| v \| \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

V.5.2 Triangularisation de Householder

Posons $A^{(1)} = A$, $B^{(1)} = B$ et effectuons la transformation de Householder du premier vecteur colonne de $A^{(1)}$. Notons $H^{(1)}$ la matrice de Householder. Une nouvelle matrice et un nouveau vecteur sont construits par multiplication par cette matrice : $A^{(2)} = H^{(1)}A^{(1)}$ et $B^{(2)} = H^{(1)}B^{(1)}$.

Le nouveau système linéaire $A^{(2)}X = B^{(2)}$ est équivalent au précédent (c'est-à-dire de même solution). La matrice $A^{(2)}$ a sa première colonne nulle sauf la première composante :

$$A^{(2)} = \begin{bmatrix} a_{11}^{(2)} & a_{12}^{(2)} & \cdots & a_{1n}^{(2)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \cdots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix}$$

Puis on effectue une nouvelle transformation de Householder à partir du deuxième vecteur colonne $a_{i2}^{(2)}$ pour $i = 2, ..., n$. Notons $\tilde{H}^{(2)}$ la matrice de Householder de dimension $n-1$. On calcule la matrice $H^{(2)}$ de dimension n de la façon suivante :

$$H^{(2)} = \left[\begin{array}{c|ccc} 1 & 0 & \cdots & 0 \\ \hline 0 & & & \\ \vdots & & \tilde{H}^{(2)} & \\ 0 & & & \end{array}\right]$$

On construit une nouvelle matrice et un nouveau vecteur par multiplication par cette matrice : $A^{(3)} = H^{(2)}A^{(2)}$ et $B^{(3)} = H^{(2)}B^{(2)}$. Le nouveau système linéaire $A^{(3)}X = B^{(3)}$ est équivalent aux deux précédents. La matrice $A^{(3)}$ a ses deux premières colonnes nulles sous la diagonale :

$$A^{(3)} = \begin{bmatrix} a_{11}^{(2)} & a_{12}^{(2)} & a_{13}^{(2)} & \cdots & a_{1n}^{(2)} \\ 0 & a_{22}^{(3)} & a_{23}^{(3)} & \cdots & a_{2n}^{(3)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{3n}^{(3)} & \cdots & a_{nn}^{(3)} \end{bmatrix}$$

Après $k-1$ itérations du même type, on a déterminé une matrice $A^{(k)}$ et un vecteur $B^{(k)}$. On effectue une nouvelle transformation de Householder à partir du k-ième vecteur colonne $a_{ik}^{(k)}$ pour $i = k, ..., n$. Notons $\tilde{H}^{(k)}$ la matrice de Householder de dimension $n-k+1$. On calcule la matrice $H^{(k)}$ de dimension n de la façon suivante :

$$H^{(k)} = \left[\begin{array}{c|c} I_{k-1} & 0 \\ \hline 0 & \tilde{H}^{(k)} \end{array}\right]$$

On construit une nouvelle matrice et un nouveau vecteur par multiplication par cette matrice : $A^{(k+1)} = H^{(k)}A^{(k)}$ et $B^{(k+1)} = H^{(k)}B^{(k)}$. Le nouveau système linéaire $A^{(k+1)}X = B^{(k)}$ est équivalent aux k précédents.

Après $n-1$ itérations, on aboutit au système linéaire $A^{(n)}X = B^{(n)}$ où la matrice $A^{(n)}$ est triangulaire supérieure. Le système se résout alors en cascade en commençant par x_n.

V.5.3 Factorisation QR

Méthode de factorisation pour une matrice A quelconque basée sur la triangularisation de Householder. Il existe une **unique** décomposition $A = Q \times R$ en une matrice R triangulaire supérieure et une matrice Q orthogonale.

Les matrices R et Q sont déterminées à partir des $n-1$ matrices de Householder $H^{(1)}, H^{(2)}, ..., H^{(n-1)}$ et de la matrice $A^{(n)}$: $R = A^{(n)}$ et $Q = H^{(1)} \times H^{(2)} \times ... \times H^{(n-1)}$

Le système $AX = B$ se ramène alors à la résolution d'un système triangulaire : $RX = {}^tQB$

V.6 METHODES ITERATIVES

Ces méthodes induisent un processus itératif par construction d'une suite de vecteur qui converge vers la solution du système. A chaque itération $k+1$, un vecteur $X^{(k+1)}$ est évalué à partir du vecteur de l'itération précédente $X^{(k)}$, ce qui peut s'écrire sous forme matricielle :

$$MX^{(k+1)} = NX^{(k)} + B$$

où les matrices M et N vérifient $M - N = A$.

Il se pose plusieurs problèmes :

- Initialisation de la méthode, choix du vecteur $X^{(0)}$.
- Condition de convergence.
- Critère de convergence et nombre d'itérations à effectuer.
- Vitesse de convergence et efficacité de la méthode.

Le vecteur $R^{(k)} = B - AX^{(k)}$ s'appelle le vecteur résidu. Il tend vers le vecteur nul lorsque la méthode converge.

V.6.1 Méthode de Jacobi

1) Initialisation avec un $X^{(0)}$ arbitraire.
2) A l'itération $k+1$, le vecteur $X^{(k+1)}$ est calculé par la relation :

$$\begin{cases} x_1^{(k+1)} = \dfrac{1}{a_{11}} \left[b_1 - \displaystyle\sum_{j=2}^{n} a_{1j} x_j^{(k)} \right] \\ \vdots \\ x_i^{(k+1)} = \dfrac{1}{a_{ii}} \left[b_i - \displaystyle\sum_{j=1, j\neq i}^{n} a_{ij} x_j^{(k)} \right] \\ \vdots \\ x_n^{(k+1)} = \dfrac{1}{a_{nn}} \left[b_n - \displaystyle\sum_{j=1}^{n-1} a_{nj} x_j^{(k)} \right] \end{cases}$$

Sous forme matricielle, en décomposant la matrice A en trois matrices D, E et F respectivement diagonale, triangulaire inférieure et triangulaire supérieure :

$$X^{(k+1)} = D^{-1}(E+F)X^{(k)} + D^{-1}B$$

V.6.2 Méthode de Gauss-Seidel

1) Initialisation avec un $X^{(0)}$ arbitraire.
2) A l'itération $k+1$, le vecteur $X^{(k+1)}$ est calculé par la relation :

$$\begin{cases} x_1^{(k+1)} = \dfrac{1}{a_{11}}\left[b_1 - \displaystyle\sum_{j=2}^{n} a_{1j}x_j^{(k)}\right] \\ \vdots \\ x_i^{(k+1)} = \dfrac{1}{a_{ii}}\left[b_i - \displaystyle\sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right] \\ \vdots \\ x_n^{(k+1)} = \dfrac{1}{a_{nn}}\left[b_n - \displaystyle\sum_{j=1}^{n-1} a_{nj}x_j^{(k+1)}\right] \end{cases}$$

Sous forme matricielle : $X^{(k+1)} = (D-E)^{-1}FX^{(k)} + (D-E)^{-1}B$

V.6.3 Méthode de Gauss-Seidel avec sur- ou sous-relaxation

La méthode permet d'accélérer la convergence par rapport à la méthode de Gauss-Seidel. Elle consiste à pondérer, à chaque itération, le résultat obtenu par la méthode de Gauss-Seidel et le résultat de l'itération précédente, par l'intermédiaire d'un paramètre de relaxation ω compris entre 0 et 2.

Si l'on note $x_{i,GS}^{(k+1)}$ la valeur de x_i à l'itération $k+1$ évaluée par la méthode de Gauss-Seidel. La valeur à l'itération $k+1$ est :

$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \omega x_{i,GS}^{(k+1)}$$

Pour $\omega = 1$, on retrouve la méthode Gauss-Seidel.
Pour $\omega > 1$, on parlera de sur-relaxation.
Pour $\omega < 1$, on parlera de sous-relaxation.

Sous forme matricielle :

$$X^{(k+1)} = \left(\frac{D}{\omega} - E\right)^{-1}\left[\frac{1-\omega}{\omega}D + F\right]X^{(k)} + \left(\frac{D}{\omega} - E\right)^{-1}B$$

V.6.4 Condition de convergence

Une condition nécessaire et suffisante de convergence d'une méthode itérative est que le rayon spectral de la matrice $\mathcal{L} = M^{-1} \times N$ (où M et N dépendent de la méthode itérative choisie) soit inférieur à 1.

$$\rho(\mathcal{L}) < 1 \quad \Leftrightarrow \quad \text{convergence}$$

En pratique cette condition est difficile à utiliser car elle nécessite de connaître le maximum en module des valeurs propres de la matrice $\mathcal{L}$.

Pour les méthodes de Jacobi et Gauss-Seidel, il existe une condition suffisante de convergence plus pratique à vérifier, à savoir que la matrice A soit à **diagonale dominante**.

$$\text{pour toutes les colonnes, } \sum_{i=1, i\neq j}^{n} \mid a_{ij} \mid < a_{jj} \quad \Rightarrow \quad \text{convergence}$$

$$\text{pour toutes les lignes, } \sum_{j=1, j\neq i}^{n} \mid a_{ij} \mid < a_{ii} \quad \Rightarrow \quad \text{convergence}$$

<u>En conséquence</u> : modifier l'ordre des lignes ou des colonnes de la matrice A pour obtenir une diagonale dominante.

Pour la méthode de Gauss-Seidel avec sur- ou sous-relaxation, il existe une autre condition nécessaire et suffisante dans le cas où la matrice A est symétrique :

$$A \text{ définie positive} \quad \Leftrightarrow \quad \text{convergence pour } A \text{ symétrique}$$

V.7 METHODE DU GRADIENT CONJUGUE

Méthode "itérative" pour une matrice A **définie positive** et **symétrique**. L'algorithme converge en au plus n itérations. En théorie c'est une méthode directe. En pratique à cause des erreurs d'arrondi on la considère comme une méthode itérative.

Résoudre le système $AX = B$ est équivalent à chercher le minimum de la fonction quadratique $J(X)$ définie par : $J(X) = < AX, X > -2 < B, X >$

Le minimum est obtenu en annulant le gradient de J (d'où le nom de méthode du gradient).

La forme matricielle de la méthode itérative est : $X^{(k+1)} = X^{(k)} + \alpha_k P^{(k)}$
où $P^{(k)}$ est un vecteur et α_k un scalaire.
On note $R^{(k)} = B - AX^{(k)}$ le vecteur résidu à l'itération k.

V.7.1 L'algorithme

1) Initialisation avec un $X^{(0)}$ arbitraire, résidu $R^{(0)} = B - AX^{(0)}$ et vecteur $P^{(0)} = R^{(0)}$
2) Itération k variant de 0 à $n-1$, fin de l'algorithme si le résidu $R^{(k+1)}$ est nul.

$$\begin{cases} \alpha_k & = & \dfrac{\| R^{(k)} \|^2}{< AP^{(k)}, P^{(k)} >} \\ X^{(k+1)} & = & X^{(k)} + \alpha_k P^{(k)} \\ R^{(k+1)} & = & R^{(k)} - \alpha_k AP^{(k)} \\ \beta_{k+1} & = & \dfrac{\| R^{(k+1)} \|^2}{\| R^{(k)} \|^2} \\ P^{(k+1)} & = & R^{(k+1)} + \beta_{k+1} P^{(k)} \end{cases}$$

V.7.2 Coût de la méthode

Le coût de la méthode, pour n grand, est de $2n^3$ flops. C'est bien supérieur à Cholesky.

Pour diminuer le coût on introduit un préconditionnement de la matrice A. Le coût devient alors inférieur à n opérations.

V.8 GRADIENT CONJUGUE PRECONDITIONNE

Le principe consiste à remplacer le système initial $AX = B$ par le système $C^{-1}AX = C^{-1}B$ avec une matrice de préconditionnement C^{-1}bien choisie. Cette méthode est l'une des mieux adaptées à la résolution de grand système linéaire dont la matrice est symétrique, définie positive et creuse.

V.8.1 L'algorithme

1) Initialisation avec un $X^{(0)}$ arbitraire, résidu $R^{(0)} = B - AX^{(0)}$. Le vecteur $P^{(0)}$ est obtenu en résolvant le système $CP^{(0)} = R^{(0)}$. On introduit le vecteur $Z^{(0)} = R^{(0)}$.

2) Itération k variant de 0 à $n-1$

$$\begin{cases} \alpha_k & = & \dfrac{< R^{(k)}, Z^{(k)} >}{< AP^{(k)}, P^{(k)} >} \\ X^{(k+1)} & = & X^{(k)} + \alpha_k P^{(k)} \\ R^{(k+1)} & = & R^{(k)} - \alpha_k AP^{(k)} \\ CZ^{(k+1)} & = & R^{(k+1)} \\ \beta_{k+1} & = & \dfrac{< R^{(k+1)}, Z^{(k+1)} >}{< R^{(k)}, Z^{(k)} >} \\ P^{(k+1)} & = & Z^{(k+1)} + \beta_{k+1} P^{(k)} \end{cases}$$

A chaque itération, il faut résoudre le systèle linéaire $CZ = R$.

Il existe plusieurs préconditionnements, citons par exemple :
- SSOR d'Evans
- préconditionnement basé sur Cholesky $C = T\ {}^tT$

V.8.2 Comparaison avec Cholesky

		Cholesky		Gradient Conjugué		flops(Chol)/	Mem(Chol/
n	m/n^2	flops	mémoire	flops	mémoire	flops(GC)	Mem(GC)
47	0.12	$8.05\,10^3$	464	$1.26\,10^4$	228	0.64	2.04
83	0.07	$3.96\,10^4$	1405	$3.03\,10^4$	533	1.31	2.64
150	0.04	$2.01\,10^5$	4235	$8.86\,10^4$	1245	2.26	3.4
225	0.03	$6.39\,10^5$	9260	$1.95\,10^5$	2073	3.27	4.47
329	0.02	$1.74\,10^6$	17974	$3.39\,10^5$	3330	5.15	5.39
424	0.02	$3.78\,10^6$	30185	$5.49\,10^5$	4513	6.88	6.83
530	0.01	$8.31\,10^6$	50785	$8.61\,10^5$	5981	9.65	8.49
661	0.01	$1.19\,10^7$	68468	$1.11\,10^6$	7421	10.66	9.23

TAB. V.1 – Coût de calcul (flops) et de la mémoire occupée (bytes) entre les méthodes de Cholesky et du Gradient Conjugué pour des matrices creuses $n \times n$ avec m éléments non nuls.

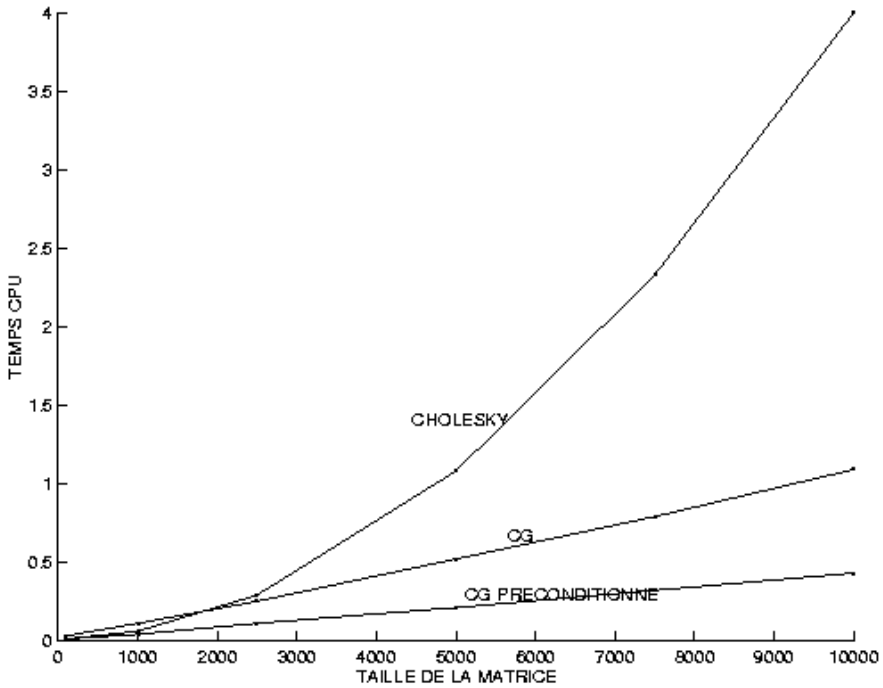

FIG. V.1 – Coût de calcul CPU entre les méthodes de Cholesky et du Gradient Conjugué pour des matrices creuses $n \times n$.

Chapitre VI

RESOLUTION DES SYSTEMES NON LINEAIRES

VI.1 EQUATIONS NON LINEAIRES

Le problème consiste à trouver les solutions d'une équation non-linéaire de la forme $F(x) = 0$. Il peut aussi se formuler comme la recherche du point fixe de la fonction $G(x) = F(x) + x$.

Les méthodes qui seront étudiées ici procèdent par itérations. Partant d'une valeur initiale x_0, un algorithme itératif permet de calculer les itérés successifs $x_1, x_2, x_3, ...$ Le calcul est arrêté lorsqu'une précision suffisante est atteinte.
Parfois, la convergence n'est garantie que pour certaines valeurs de x_0. Parfois encore, l'algorithme peut ne pas converger.

Quelles sont les difficultés des méthodes itératives ?

- Le choix de x_0
- La convergence (locale, globale) de la méthode
- La vitesse de convergence et l'efficacité de la méthode
- La propagation des erreurs numériques et la précision

Les principales méthodes itératives :

- Méthode du point fixe
- Méthode de Newton ou Newton-Raphson
- Méthode de la sécante (ou regula-falsi)
- Méthode de la parallèle
- Méthode de Steffensen
- Méthode de Bairstow (racine d'un polynôme)

VI.1.1 Vitesse de convergence

Soit une suite (x_n), issue d'une méthode itérative, qui converge vers X. On s'intéresse à la vitesse de convergence de cette suite. On dit que :

• la convergence est d'ordre 1 ou linéaire si il existe $\beta \in]0,1[$ tel que

$$\lim_{n\to\infty} \frac{\| x_{n+1} - X \|}{\| x_n - X \|} = \beta$$

Si $\beta = 0$ la convergence est dite super linéaire.

• la convergence est d'ordre p si il existe $\beta > 0$ tel que

$$\lim_{n\to\infty} \frac{\| x_{n+1} - X \|}{\| x_n - X \|^p} = \beta$$

Si $p = 2$, la convergence d'ordre 2 est dite quadratique.
Si $p = 3$, la convergence d'ordre 3 est dite cubique.

Remarque : p n'est pas forcément un nombre entier.

La suite convergence d'autant plus rapidement que la valeur de p est elevée.

VI.1.2 Méthode du point fixe

Méthode pour résoudre $G(x) = x$

L'algorithme

1) Choix d'un x_0.
2) Itération $x_{n+1} = G(x_n)$ jusqu'à convergence.

Convergence de la méthode

Notons r l'unique point fixe de $G(x)$ sur l'intervalle [a,b].

si $x_0 \in [a,b]$ et si $| G'(r) | < 1$ alors la méthode converge vers r.

On appelle bassin d'attraction du point fixe r le voisinage de r pour lequel la méthode du point fixe converge vers r quelque soit la valeur initiale x_0 appartenant à ce voisinage.

$| G'(r) | < 1 \quad \rightarrow \quad$ le point fixe est dit attractif
$| G'(r) | > 1 \quad \rightarrow \quad$ le point fixe est dit répulsif
$| G'(r) | < 1 \quad \rightarrow \quad$ cas indéterminé

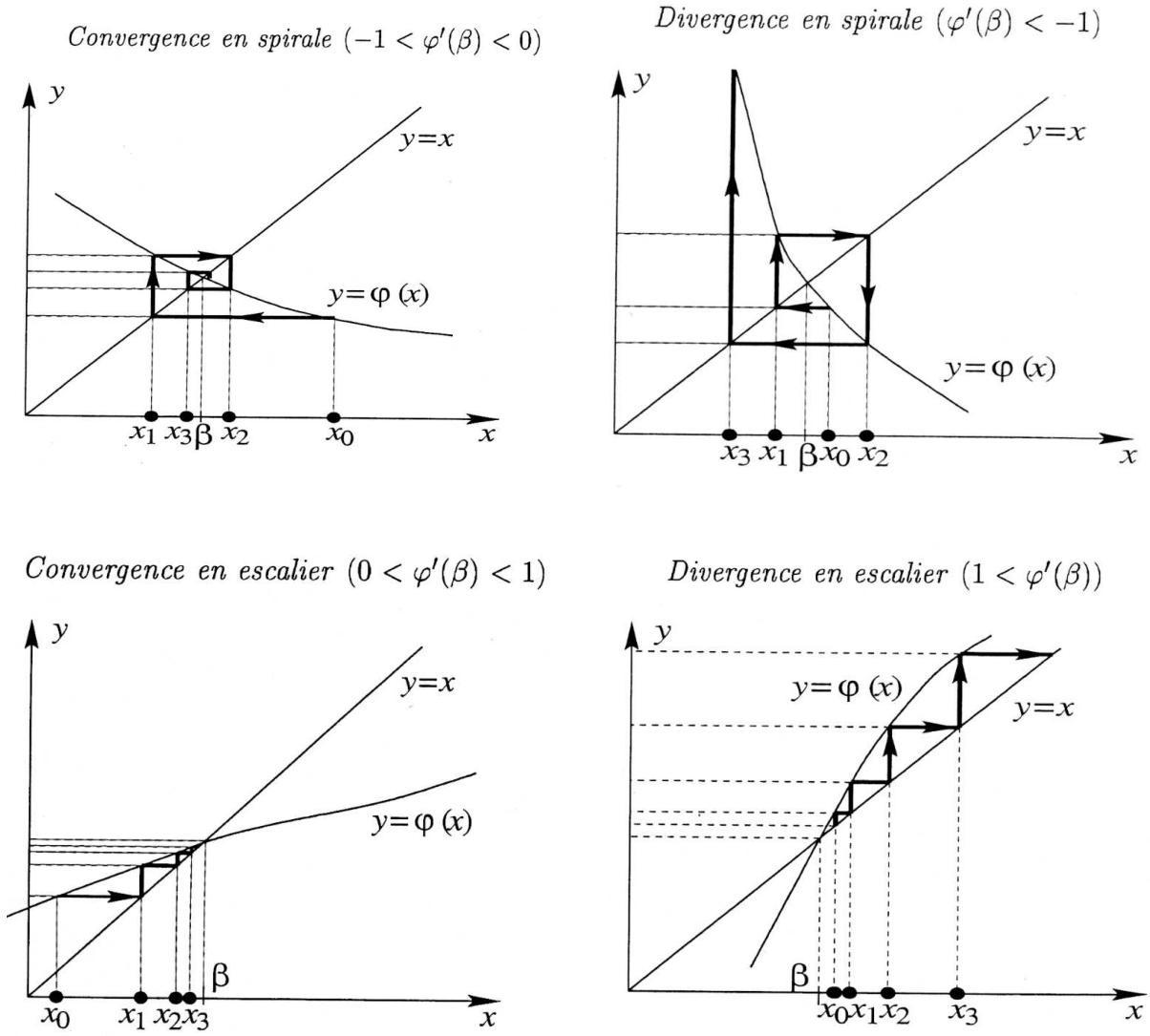

FIG. VI.1 – Point fixe - concergence et divergence

VI.1.3 Méthode de Newton

Méthode très utilisée pour résoudre $F(x) = 0$

L'algorithme

1) Choix d'un x_0.
2) Itération $x_{n+1} = x_n - \dfrac{F(x_n)}{F'(x_n)}$ jusqu'à convergence.

Convergence de la méthode

Notons X l'unique racine de $F(x)$. Si x_0 est suffisamment proche de X alors la méthode converge vers X et la convergence est quadratique.
Dans le cas d'une racine multiple, la convergence est linéaire.

Les problèmes de la méthode

1) Une première difficulté apparaît lorsque la dérivée $F'(x_i)$ est nulle, la formule n'est alors plus applicable.
2) Lorsque la valeur de la pente de la tangente $F'(x_i)$ est non nulle mais très petite, le point x_{i+1} peut se retrouver très loin de x_i. Ceci peut entraîner une convergence, par accident, vers une racine très éloignée de x_0.
3) Des situations de bouclage peuvent se produire si des termes de la suite se répétent.

VI.1.4 Méthode de la parallèle

Méthode pour résoudre $F(x) = 0$

L'algorithme

1) Choix d'un x_0 et d'un réel λ.
2) Itération $x_{n+1} = x_n - \lambda F(x_n)$ jusqu'à convergence.
Le réel λ est souvent choisi égale à $1/F'(x_0)$

Pour x_0 suffisament proche de la racine X, la méthode convergence linéairement.

VI.1.5 Méthode de la sécante

Méthode pour résoudre $F(x) = 0$

L'algorithme

1) Choix d'un x_0.
2) Calcul de x_1 par la méthode de Newton.
3) Itération $x_{n+1} = x_n - F(x_n)\dfrac{x_n - x_{n-1}}{F(x_n) - F(x_{n-1})}$ jusqu'à convergence.

Pour x_0 suffisament proche de la racine X, la méthode convergence à l'ordre 1.618.

VI.1.6 Méthode de Steffensen

Méthode pour résoudre $G(x) = x$

L'algorithme

1) Choix d'un x_0.
2) Itération $x_{n+1} = \dfrac{G\left(G(x_n)\right)x_n - G(x_n)^2}{G\left(G(x_n)\right) - 2G(x_n) + x_n}$ jusqu'à convergence.

Si x_0 est suffisamment proche de X alors la convergence de la méthode est quadratique.

VI.1.7 Racines de polynômes

VI.1.7.1 Réduction polynomiale

La recherche de racines d'un polynôme se construit de la manière suivante : soit $P_n(x)$ un polynôme de degré n, si on obtient une première racine x_1, on peut écrire

$$P_n(x) \;=\; (x - x_1)P_{n-1}(x)$$

où $P_{n-1}(x)$ est un polynôme de degré $n-1$.

Ainsi, une fois obtenue une première racine, on peut recommencer la recherche d'une autre racine pour un polynôme de degré strictement inférieur. On poursuit cette procédure jusqu'à l'obtention des n racines complexes du polynôme.

VI.1.7.2 Méthode de Bairstow

Soit $P(x)$ un polynôme de degré n : $P(x) \;=\; a_0x^n + a_1x^{n-1} + ... + a_{n-1}x + a_n$

La méthode consiste à calculer les facteurs quadratiques de P (les polynômes de degré 2) :

Si n est pair ($n = 2p$), $\qquad P(x) \;=\; a_0 \prod_{j=1}^{p} \left(x^2 + p_jx + q_j\right)$

Si n est impair ($n = 2p+1$), $P(x) \;=\; a_0\,(x-\alpha) \prod_{j=1}^{p} \left(x^2 + p_jx + q_j\right)$

L'algorithme

On introduit le polynôme $Q(x)$ de degré $n-2$ et les fonctions $R(p,q), S(p,q)$ obtenus par division de $P(x)$ par le polynôme $x^2 + px + q$ de degré 2 :

$$P(x) \;=\; Q(x)\left(x^2 + px + q\right) \;+\; xR(p,q) \;+\; S(p,q)$$

avec $Q(x) \;=\; b_0x^{n-2} + b_1x^{n-3} + ... + b_{n-3}x + b_{n-2}$
Si $x^2 + px + q$ est un diviseur de $P(x)$, on a :

$$\begin{cases} R(p,q) & = & 0 \\ P(p,q) & = & 0 \end{cases}$$

Pour trouver p et q, il suffit de résoudre ce système. Par exemple par la méthode Newton :

$$\begin{bmatrix} p^{(i+1)} \\ \\ q^{(i+1)} \end{bmatrix} = \begin{bmatrix} p^{(i)} \\ \\ q^{(i)} \end{bmatrix} - \begin{bmatrix} \dfrac{\partial R}{\partial p} & \dfrac{\partial R}{\partial q} \\ \dfrac{\partial S}{\partial p} & \dfrac{\partial S}{\partial q} \end{bmatrix}^{-1}_{(p^{(i)},q^{(i)})} \times \begin{bmatrix} R\left(p^{(i)},q^{(i)}\right) \\ \\ S\left(p^{(i)},q^{(i)}\right) \end{bmatrix}$$

On obtient en identifiant :

$$\begin{array}{lcl} R(p,q) = b_{n-1} & ; & S(p,q) = b_n + pb_{n-1} \\ \dfrac{\partial R}{\partial p}(p,q) = -c_{n-2} & ; & \dfrac{\partial S}{\partial p}(p,q) = b_{n-1} - c_{n-1} - pc_{n-2} \\ \dfrac{\partial R}{\partial p}(p,q) = -c_{n-3} & ; & \dfrac{\partial S}{\partial p}(p,q) = -c_{n-2} - pc_{n-3} \end{array}$$

avec
$b_{-1} = 0$, $b_0 = a_0$, $c_{-1} = 0$, $c_0 = b_0$

$b_k = a_k - pb_{k-1} - qb_{k-2}$ pour $k = 1, ..., n$

$c_k = b_k - pc_{k-1} - qc_{k-2}$ pour $k = 1, ..., n-1$

VI.2 SYSTEMES D'EQUATIONS NON LINEAIRES

La méthode de Newton vue précédemment pour une équation non linéaire peut être généralisée pour résoudre des systèmes à n équations non linéaires :

$$\begin{cases} f_1(x_1, x_2, ..., x_n) & = & 0 \\ f_2(x_1, x_2, ..., x_n) & = & 0 \\ \vdots & & \vdots \\ f_n(x_1, x_2, ..., x_n) & = & 0 \end{cases}$$

Ou encore, chercher un vecteur $X = (x_1, x_2, ..., x_n)$ vérifiant $F(X) = 0$

L'algorithme

1) Choix d'un vecteur inital $X^{(0)}$.
2) Itération $X^{(k+1)} = X^{(k)} - \left[J(X^{(k)})\right]^{-1} F(X^{(k)})$ jusqu'à convergence.
où $J(X^{(k)})$ est la matrice jacobienne évalué en $X^{(k)}$.

Ce qui s'écrit encore :

$$\begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{bmatrix} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix} - \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}^{-1}_{(x_1^{(k)}, x_2^{(k)}, ..., x_n(k))} \times \begin{bmatrix} f_1(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)}) \\ f_2(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)}) \\ \vdots \\ f_n(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)}) \end{bmatrix}$$

En posant $\Delta X^{(k)} = X^{(k+1)} - X^{(k)}$, le problème revient à résoudre le système linéaire suivant, à chaque itération k :

$$J(X^{(k)})\,\Delta X^{(k)} \;=\; -F(X^{(k)})$$

La méthode de Newton linéarise ainsi le système. Il faut alors, à chaque itération, résoudre un système linéaire jusqu'à ce que le vecteur X soit suffisament proche de la solution.

Pour une seule équation non linéaire et une suite de réels, le critère d'arrêt est basé sur l'erreur $\varepsilon_k =\mid x_{k+1} - x_k \mid$ entre le résultats de deux itérations successives (par exemple, $\varepsilon < 10^{-5}$).
Pour un système d'équations et une suite de vecteurs, le critère d'arrêt est basé sur la norme entre le résultats de deux itérations successives : $\mid\mid X^{(k+1)} - X^{(k)} \mid\mid$, qui doit être inférieure à une valeur donnée.

En calcul numérique, trois normes sont fréquemment utilisées :

1) La norme L_1 : $\mid\mid X \mid\mid_1 = \sum_{i=1}^{n} \mid x_i \mid$

2) La norme euclidienne ou L_2 : $\mid\mid X \mid\mid_2 = \sqrt{x_1^2 + x_2^2 + ... + x_n^2}$

3) La norme infinie : $\mid\mid X \mid\mid_\infty = \max_i \mid x_i \mid$

Chapitre VII

INTERPOLATION ET APPROXIMATION

VII.1 GENERALITES

VII.1.1 Le problème

Etant donné une fonction f, définie soit de façon discrète, soit de façon continue, l'objectif est de déterminer une autre fonction g, de forme donnée, qui, en un certain sens, approche le mieux possible la fonction f.

VII.1.2 Les 3 grandes classes d'approximation fonctionnelle

Il existe trois grandes classes de méthodes :

- **l'interpolation**
  On approche la fonction f par une fonction g appartenant à une classe de fonctions approximantes (souvent des polynômes) et qui coïncide avec f en n points discrets.

- **l'approximation aux moindres carrés**
  On minimise la somme des carrés des erreurs en n points $x_1, x_2, ..., x_n$, c'est-à-dire qu'on cherche g qui minimise la quantité : $\sum_{i=1}^{n} || f(x_i) - g(x_i) ||^2$

- **l'approximation uniforme**
  On minimise le maximum de l'amplitude de l'erreur entre la fonction f et son approximation Les points de collocation s'expriment en fonction des racines des polynômes de Tchebychev.

VII.1.3 Les 3 grandes familles de fonctions approximantes

Les trois grandes familles de fonctions approximantes sont :

- les polynômes (théorème de Stone-Weierstrass)
- les fractions rationnelles (approximants de Padé)
- les fonctions trigonométriques (série de Fourrier)

VII.2 INTERPOLATION

VII.2.1 Le théorème de Stone-Weierstrass

Toute fonction numérique continue définie sur un compact de $\mathcal{R}^n$ est limite d'une suite de polynômes qui converge uniformément sur ce compact.

Toute fonction continue à valeurs complexes, périodique de période 2π, est limite d'une suite de polynômes trigonométriques qui converge uniformément sur $\mathcal{C}$.

VII.2.2 Méthode de Lagrange

Considérons une fonction f connue en $n+1$ points $x_0, x_1, ..., x_n$ (appelés points de collocation). On cherche un polynôme de degré inférieur ou égal à n qui passe par ces $n+1$ points.

Il existe un UNIQUE polynôme $P_n(x)$ qui coïncide avec $f(x)$ aux $n+1$ points, à savoir $P_n(x_i) = f(x_i)$. Ce polynôme a pour expression :

$$P_n(x) = \sum_{i=0}^{n} f(x_i)\, L_i(x) \quad \text{où} \quad L_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}$$

Les polynômes $L_i(x)$ sont les polynômes de Lagrange et $P_n(x)$ est le polynôme d'interpolation de Lagrande de f aux ponts x_i.

Coût de l'interpolation : pour n grand $\rightarrow 3n^2$ opérations.

VII.2.3 Méthode de Neville-Aitken

Le polynôme d'interpolation $P_n(x)$ est calculé à l'aide d'une formule de récurrence.

Initialisation : $P_{i,i}(x) = f(x_i)$, pour $i = 0$ à n

Itération : $P_{i,i}(x) = \dfrac{(x - x_i)P_{i+1,j}(x) - (x - x_j)P_{i,j-1}(x)}{x_j - x_i}$, pour $i = 0$ à n, $j > i$

Le polynôme d'interpolation est $P_n(x) = P_{0,n}(x)$.

VII.2.4 Méthode de Newton

On introduit les polynômes de Newton $Q_i(x)$ définis par : $Q_i(x) = \prod_{j<i}(x - x_j)$ et $Q_0(x) = 1$

On appelle **différences divisées d'ordre** k de la fonction f aux points $x_0, x_1, ..., x_k$ la quantité notée $f[x_0, x_1, ..., x_k]$ définie par récurrence :

$$\begin{cases} f[x_i] &= f(x_i) \text{ pour i=0 à n} \\ f[x_0, x_1, ..., x_k] &= \dfrac{f[x_1, x_2, ..., x_k] - f[x_0, x_1, ..., x_{k-1}]}{x_k - x_0} \end{cases}$$

Le polynôme $P_k(x)$ d'interpolation de Newton de la fonction f aux points $x_0, x_1, ..., x_{k-1}$ est défini par récurrence :

$$P_k(x) \; - \; P_{k-1}(x) \; = \; f[x_0, x_1, ..., x_k]Q_k(x)$$

Le polynôme $P_n(x)$ d'interpolation de Newton de la fonction f aux points $x_0, x_1, ..., x_n$ est :

$$P_n(x) \; = \; f(x_0) \; + \sum_{k=1}^{n} f[x_0, x_1, ..., x_k]Q_k(x)$$

Coût de l'interpolation : pour n grand $\rightarrow n^3$ opérations.

VII.2.5 Méthode de Hermite

Plutôt que de faire coïncider la fonction f et le polynôme P_n aux points x_i, on peut chercher à faire aussi coïncider les dérivées de f et de P_n en ces points, jusqu'à un ordre fixé que l'on notera α_i aux points x_i. Posons $N = n + \alpha_0 + \alpha_1 + ... + \alpha_n$

Si f admet des dérivées d'ordre α_i aux points x_i, il existe un UNIQUE polynôme P_N tel que :

$$Ai = 0, 1, ..., n \quad Al = 0, 1, ..., \alpha_i \quad P_N^{(l)}(x_i) \; = \; f^{(l)}(x_i)$$

P_N est le polynôme d'interpolation de Hermite de f aux points x_i aux ordres α_i.
Pour $\alpha_i = 0$, on retrouve le polynôme d'interpolation de Lagrange.

VII.2.6 Interpolation par morceaux - spline cubique

Sur tout intervalle $[x_{i-1}, x_i]$, entre deux points de collocation, on effectue une interpolation LOCALE avec des polynômes de bas degré $P_k(x)$ (degré 1, 2 ou 3).
Les polynômes se raccordent aux pointx x_i ainsi que leurs dérivées d'ordre aussi élevé que possible.

Spline cubique

Le polynôme d'interpolation locale est de degré 3. Le fonction interpolante $S(x)$ est constituée de morceaux de polynômes de degré 3 qui se raccordent, ainsi que que leurs dérivées premières et secondes, aux points de collocation.

Sur chaque intervalle $[x_{i-1}, x_i]$, la restriction de $S(x)$ est un polynôme $P_i(x)$ de degré 3 tel que :

$$\left\{ \begin{array}{rcl} P_i(x_{i-1}) & = & f(x_{i-1}) \quad \text{et} \quad P_i(x_i) \; = \; f(x_i) = f_i \\ P_i'(x_i) & = & P_{i+1}'(x_i) \; = \; f'(x_i) \\ P_i''(x_i) & = & P_{i+1}''(x_i) \; = \; f''(x_i) \end{array} \right.$$

Les polynômes $P_i(x)$ ont pour expression :

$$P_i(x) \; = \; \alpha_{i-1}\frac{(x_i - x)\left((x_i - x)^2 - h_i^2\right)}{6h_i} + \alpha_i\frac{(x - x_{i-1})\left((x - x_{i-1})^2 - h_i^2\right)}{6h_i} + \frac{f_{i-1}(x_i - x)}{h_i} + \frac{f_i(x - x_{i-1})}{h_i}$$

Avec : $h_i = x_i - x_{i-1}$ et les α_i solutions du système suivant :

$$\frac{h_i}{6}\,\alpha_{i-1} + \frac{h_i + h_{i+1}}{3}\,\alpha_i + \frac{h_{i+1}}{6}\,\alpha_{i+1} = \frac{f_{i+1} - f_i}{h_{i+1}} - \frac{f_i - f_{i-1}}{h_i}$$

où les paramètres $\alpha_0 = f''(x_0)$ et $\alpha_n = f''(x_n)$ sont choisis arbitrairement (nuls par exemple).

VII.2.7 Limites de l'interpolation polynômiale

L'interpolation polynômiale pose plusieurs problèmes :

→ lorsque le nombre de points de collocation est grand (instabilité de calculs, erreurs d'arrondi et coût de calculs).

→ en pratique, les valeurs discrètes résultent d'expériences ou de simulations numériques. Ce sont des valeurs approximatives. Il est souvent plus intéressant de chercher une bonne approximation plutôt qu'une interpolation.
Par exemple, un expérimentateur qui relève 100 points quasiment alignés sera plus intéressé par la droite passant "au mieux" par ces 100 points plutôt que par le polynôme de degré 99 réalisant l'interpolation exacte.

VII.3 APPROXIMATION

VII.3.1 Approximation rationnelle - approximants de Padé

On approche la fonction f par une fraction rationnelle : $F(x) = P_n(x)/Q_m(x)$
où $P_n(x)$ et $Q_m(x)$ sont des polynômes de degré n et m tels que le développement limité au voisinage de α de $f(x)Q_m(x) - P_n(x)$ ait son terme constant ainsi que les termes en $(x - \alpha)$, $(x_\alpha)^2$, ..., $(x - \alpha)^{n+m}$ nuls.
Le calcul consiste à identifier à 0 ces $n + m + 1$ termes.

VII.3.2 Approximation polynomiale au sens des moindres carrés

VII.3.2.1 Droite des moindres carrés discrets

On cherche à approcher la fonction f, connue uniquement en n valeurs discrètes, par une droite $P(x) = a_0 + a_1 x$ au sens des moindres carrés.

Soient n valeurs $y_1, y_2, ..., y_n$ de la fonction f aux n abscisses $x_1, x_2, ..., x_n$. La droite $P(x)$ qui réalise la meilleure approximation au sens des moindres carrés des valeurs y_i aux points x_i est celle qui minimise la somme des carrés des écarts entre les y_i et les $P(x_i)$ soit :

$$S(a_0, a_1) = \sum_{i=1}^{n} [y_i - (a_0 + a_1 x_i)]^2$$

La quantité $S(a_0, a_1)$ apparaît comme le carré de la norme euclidienne du vecteur $(y_i - a_0 - a_1 x_i)$.
Introduisons les vecteurs à n composantes suivants : $Y = (y_i)$, $X = (x_i)$ et U le vecteur unité.
La méthode consiste à chercher le vecteur G le plus proche du vecteur Y dans le sous-espace vectoriel de dimension 2 engendré par les vecteurs U et X, c'est-à-dire à minimiser la distance

euclidienne $\| Y - G \|^2$. Pour vérifier ceci, le vecteur $Y - G$ doit être orthogonal aux vecteurs U et X, d'où les relations suivantes :

$$\begin{cases} < Y - G, U > & = & 0 & = & \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i) \\ < Y - G, X > & = & 0 & = & \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)\, x_i \end{cases}$$

Ceci conduit au système linéaire suivant :

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \end{bmatrix}$$

VII.3.2.2 Droite des moindres carrés continus

On cherche à approcher la fonction f, connue sur l'intervalle $[a, b]$, par une droite $P(x) = a_0 + a_1 x$ au sens des moindres carrés.

La détermination des coefficients a_0 et a_1 s'effectue identiquement à la méthode des moindres carrés discrèts. Le produit scalaire utilisé n'est plus basé sur la norme euclidienne :

$$< g_1, g_2 > = \int_a^b g_1(x) g_2(x)\, dx$$

Le système linéaire à résoudre devient :

$$\begin{bmatrix} \int_a^b dx & \int_a^b x\, dx \\ \int_a^b x\, dx & \int_a^b x^2\, dx \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \int_a^b f(x)\, dx \\ \int_a^b x f(x)\, dx \end{bmatrix}$$

VII.3.2.3 Généralisation - Polynôme des moindres carrés discrets

On cherche à approcher la fonction f, connue uniquement en n valeurs discrètes, par un polynôme de degré $m < n$ $P(x) = a_0 + a_1 x + a_2 x^2 + ... + a_m x^m$ au sens des moindres carrés.

Soient n valeurs $y_1, y_2, ..., y_n$ de la fonction f aux n abscisses $x_1, x_2, ..., x_n$. Le polynôme $P(x)$ qui réalise la meilleure approximation au sens des moindres carrés des valeurs y_i aux points x_i est celle qui minimise la somme des carrés des écarts entre les y_i et les $P(x_i)$ soit :

$$S(a_0, a_1, a_2, ..., a_m) = \sum_{i=1}^{n} [y_i - (a_0 + a_1 x_i + ... + a_m x_i^m)]^2$$

La quantité $S(a_0, a_1, ..., a_m)$ apparaît comme le carré de la norme euclidienne du vecteur $(y_i - a_0 - a_1 x_i - ... - a_m x_i^m)$. Introduisons les vecteurs à n composantes suivants : $Y = (y_i)$, $X = (x_i)$, $X^2 = (x_i^2)$,..., $X^m = (x_i^m)$ et U le vecteur unité.
La méthode consiste à chercher le vecteur G le plus proche du vecteur Y dans le sous-espace vectoriel de dimension $m+1$ engendré par les vecteurs U, X, X^2,...,X^m c'est-à-dire à minimiser

la distance euclidienne $\| Y - G \|^2$. Pour vérifier ceci, le vecteur $Y - G$ doit être orthogonal ux vecteurs U, X, X^2,...,X^m d'où les relations suivantes :

$$\left\{\begin{array}{rcccl} < Y-G, U > & = & 0 & = & \displaystyle\sum_{i=1}^{n} (y_i - a_0 - a_1 x_i - ... - a_m x_i^m) \\ < Y-G, X > & = & 0 & = & \displaystyle\sum_{i=1}^{n} (y_i - a_0 - a_1 x_i - ... - a_m x_i^m)\, x_i \\ < Y-G, X^2 > & = & 0 & = & \displaystyle\sum_{i=1}^{n} (y_i - a_0 - a_1 x_i - ... - a_m x_i^m)\, x_i^2 \\ & \vdots & & & \vdots \\ < Y-G, X^m > & = & 0 & = & \displaystyle\sum_{i=1}^{n} (y_i - a_0 - a_1 x_i - ... - a_m x_i^m)\, x_i^m \end{array}\right.$$

Ceci conduit au système linéaire suivant :

$$\begin{bmatrix} n & \sum_{i=1}^n x_i & \cdots & \sum_{i=1}^n x_i^m \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \cdots & \sum_{i=1}^n x_i^{m+1} \\ \vdots & \vdots & \cdots & \vdots \\ \sum_{i=1}^n x_i^{m-1} & \sum_{i=1}^n x_i^m & \cdots & \sum_{i=1}^n x_i^{2m-1} \\ \sum_{i=1}^n x_i^m & \sum_{i=1}^n x_i^{m+1} & \cdots & \sum_{i=1}^n x_i^{2m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{m-1} \\ a_m \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \\ \vdots \\ \sum_{i=1}^n x_i^{m-1} y_i \\ \sum_{i=1}^n x_i^m y_i \end{bmatrix}$$

Posons $U_q = \displaystyle\sum_{i=1}^{n} x_i^q$ et $V_q = \displaystyle\sum_{i=1}^{n} x_i^q y_i$. Le système s'écrit alors :

$$\begin{bmatrix} U_0 & U_1 & \cdots & U_m \\ U_1 & U_2 & \cdots & U_{m+1} \\ \vdots & \vdots & \cdots & \vdots \\ U_m & U_{m+1} & \cdots & U_{2m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{bmatrix} = \begin{bmatrix} V_0 \\ V_1 \\ \vdots \\ V_m \end{bmatrix}$$

VII.3.3 Approximation trigonométrique au sens des moindres carrés

Méthode analogue au cas polynomial en utilisant des polynômes trigonométriques de degré m :

$$P_m(x) = \frac{a_0}{2} + \sum_{k=1}^{m} (a_k \cos kx + b_k \sin kx)$$

Les conditions d'orthogonalité s'appliquent aux vecteurs $U/2$, $\cos X$, $\sin X$, $\cos 2X$, $\sin 2X$, ..., $\cos mX$, $\sin mX$.
Dans le cas continu, le produit scalaire est défini par :

$$< g_1, g_2 > = \int_0^{2\pi} g_1(x) g_2(x)\, dx$$

VII.3.4 Approximation uniforme - Meilleure approximation

Soit f une fonction définie sur $[a, b]$. On souhaiterait identifier le choix des points de collocation $x_0, x_1, ..., x_n$ pour lequel le polynôme d'interpolation de Lagrange $P_n(x)$ réalise la meilleure approximation de f au sens de la norme infinie, c'est-à-dire à minimiser l'erreur d'interpolation :

$$\varepsilon(x_0, x_1, ...x_n) \;=\; \max_{x\in[a,b]} \mid f(x) - P_n(x) \mid$$

La solution de ce problème n'est pas connue en général. Cependant, lorsque la fonction $f \in C^{n+1}([a, b])$, on peut introduire la définition suivante de meilleure approximation :
On appelle meilleure approximation de f par un polynôme de degré au plus égal à n, le polynôme d'interpolation de Lagrange $P_n(x)$ de f, associé aux points de collocation $x_0, x_1, ..., x_n$ pour lesquels la quantité suivante est minimale :

$$\max_{x\in[a,b]} \mid \prod_{i=0}^{n}(x - x_i) \mid$$

On montre que les points de collocation x_i s'expriment en fonction des racines du polynôme de Tchebychev T_{n+1} et vérifient :

$$x_i \;=\; \frac{a+b}{2} \;+\; \frac{b-a}{2}\cos\left(\frac{(2i+1)\pi}{2n+2}\right) \quad ; \quad \text{pour}\, i = 0, 1, ..., n$$

Polynôme de Tchebychev : polynôme de degré n noté $T_n(x)$ défini sur l'intervalle $[-1, 1]$ par :

$$T_n(x) = \cos\left(n \arccos x\right) \qquad \text{ou par recurrence} \qquad T_{n+1}(x) + T_{n-1}(x) \;=\; 2xT_n(x)$$

avec $T_0(x) = 1$ et $T_1(x) = x$.

VII.3.5 Approximation polynomiale dans une base de polynômes orthogonaux

On cherche une approximation de la fonction f par un polynôme exprimée dans une base de polynômes orthogonaux.

Il existe de nombreuses bases de polynômes orthoginaux : Legendre, Jacobi, Tchebychev, Laguerre, Hermite...

Chapitre VIII

RECHERCHE DE VALEURS PROPRES

VIII.1 INTRODUCTION

Nous nous intéressons à la détermination des valeurs propres (le spectre) et/ou des vecteurs propres correspondants d'une matrice A de dimension n. Les n valeurs propres complexes λ sont les racines du polynôme caractéristique de degré n : $P_n(\lambda) = \det(A - \lambda I_n)$.

La détermination des racines de ce polynôme n'est pas efficace du point de vue numérique. Des méthodes plus adaptées sont utilisées, elles sont séparées entre deux types de méthodes :

- Les méthodes globales
  Ces méthodes permettent d'évaluer le spectre entier de la matrice A. Elles utilisent une transformation de la matrice en une matrice semblable dont on calcule ensuite les éléments propres. Les principales méthodes sont :
  - Méthode de Jacobi
  - Méthode QR
  - Transformation sous forme de matrice de Hessenberg
  - Méthode de Lanczos
  - Méthode de bissection

- Les méthodes partielles
  Ces méthodes visent à évaluer la plus grande ou la plus petite valeur propre ou encore la valeur propre la plus proche d'une valeur donnée. Les principales méthodes sont :
  - Méthode de la puissance
  - Méthode de déflation

Applications : un problème aux valeurs propres émerge d'études d'oscillateurs physiques (systèmes masse-ressort, systèmes vibratoires, systèmes quantiques, ...), d'études de dynamique des structures, de flambage de poutre ainsi que d'études de stabilité des écoulements de fluides (transition laminaire/turbulent)...

VIII.2 METHODE DE JACOBI

Méthode pour une matrice symétrique A.

La méthode revient à effectuer une suite de transformation de type rotation planaire qui permet d'annuler un élément (p, q), où p et q sont deux entiers, de la matrice A. Chaque rotation élémentaire fait intervenir une matrice orthogonale P_{pq}. On construit ainsi une suite de matrices symétriques $A^{(k)}$ qui tend vers la matrice diagonale D semblable à A.
On pose $A^{(1)} = A$ et, à chaque transformation k, on construit la matrice $A^{(k+1)}$ définie par :

$$A^{(k+1)} = {}^tP_{pq}^{(k)} \times A^{(k)} \times P_{pq}^{(k)}$$

Les éléments de la matrice symétrique $A^{(k+1)}$ sont donnés par :

$$\begin{cases} a_{ij}^{(k+1)} = a_{ij}^{(k)} & \text{pour } i \neq p, q \text{ et } j \neq p, q \\ a_{ip}^{(k+1)} = a_{ip}^{(k)} \cos\theta - a_{iq}^{(k)} \sin\theta & \text{pour } i \neq p, q \\ a_{iq}^{(k+1)} = a_{ip}^{(k)} \sin\theta + a_{iq}^{(k)} \cos\theta & \text{pour } i \neq p, q \\ a_{pp}^{(k+1)} = a_{pp}^{(k)} \cos^2\theta + a_{qq}^{(k)} \sin^2\theta - 2a_{pq}^{(k)} \cos\theta \sin\theta & \\ a_{qq}^{(k+1)} = a_{pp}^{(k)} \sin^2\theta + a_{qq}^{(k)} \cos^2\theta + 2a_{pq}^{(k)} \cos\theta \sin\theta & \end{cases}$$

où l'angle $\theta \in]-\frac{\pi}{4}, 0[\cup]0, \frac{\pi}{4}[$ vérifie : $\tan 2\theta = \dfrac{2a_{pq}^{(k)}}{a_{qq}^{(k)} - a_{pp}^{(k)}}$

Deux méthodes pour le choix des entiers p et q :
1) Les deux entiers peuvent être choisis de façon cyclique : $p = 1, q = 2$ puis $(p = 1, q = 3)$, etc..
2) Les deux entiers peuvent être choisis tel que l'élément $\mid a_{pq}^{(k)} \mid$ soit la plus grande valeur hors éléments de la diagonale.

VIII.3 METHODE QR

Méthode pour une matrice quelconque A.

La méthode consiste à effectuer des factorisations QR successives sur une suite de matrices $A^{(k)}$ qui convergent vers une matrice triangulaire supérieure semblable à A.
Posons $A^{(1)} = A$. A la k-ième étape, appelons $Q^{(k)}$, $R^{(k)}$ les matrices issues de la factorisation QR de $A^{(k)}$. La matrice $A^{(k+1)}$ est définie par : $A^{(k+1)} = R^{(k)} \times Q^{(k)}$.

Cette méthode est plus efficace que la méthode de Jacobi. Il existe de nombreuses variantes (QL, QZ...).

VIII.4 TRANSFORMATION EN MATRICE DE HESSENBERG

Méthode pour une matrice quelconque A.

Avec la transformation $X = T.Y$ où T est une matrice inversible, le problème $A.X = \lambda X$ devient $(T^{-1}AT).Y = \lambda Y$. Les matrices A et $T^{-1}AT$ sont semblables. Le but est de trouver une matrice T telle que la matrice $T^{-1}AT$ devienne "plus simple".
Une possibilité est de chercher la matrice T tel que $T^{-1}AT$ soit une matrice de Hessenberg H, c'est-à-dire vérifiant $h_{ij} = 0$ pour $i > j + 1$:

$$T^{-1}AT = H = \begin{bmatrix} * & * & \cdots & \cdots & * \\ * & * & \ddots & & \vdots \\ & * & \ddots & \ddots & * \\ & & \ddots & \ddots & * \\ & & & * & * \end{bmatrix}$$

Pour arriver à ce but, il existe plusieurs algorithmes :

- transformations élémentaires par élimination de Gauss.
- transformations orthogonales (Householder, Schur, Lanczos...).

VIII.5 METHODE DE LANCZOS

Méthode itérative pour une matrice quelconque A.

La méthode consiste à calculer les puissances successives de A en construisant un suite de vecteurs orthogonaux U_k selon :

$$\beta_{i+1}U_k = A.U_{k-1} - \alpha_k U_{k-1} - \beta_k U_{k-2} \quad \text{pour}\, k = 2, ..., n-1$$

avec : $\begin{cases} \alpha_k & = & {}^tU_{k-1}.A.U_{k-1} \\ \beta_{k+1} & = & \sqrt{{}^tU_{k-1}.A^2.U_{k-1} - \alpha_k^2 - \beta_k} \end{cases}$

Les vecteurs $U_0, U_1, ..., U_{n-1}$ forment une base, appelée base de Krylov. Dans cette base, la matrice représentant A, ayant les mêmes valeurs propres que A, est tridiagonale :

$$A' = \begin{bmatrix} \alpha_1 & \beta_2 & 0 & 0 & \cdots & 0 \\ \beta_2 & \alpha_2 & \beta_3 & 0 & \cdots & 0 \\ 0 & \beta_3 & \alpha_3 & \beta_4 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \beta_{n-1} & \alpha_{n-1} & \beta_n \\ 0 & \cdots & \cdots & 0 & \beta_n & \alpha_n \end{bmatrix}$$

La détermination des valeurs propres peut s'effectuer avec une méthode de type QR ou une méthode de bissection.

Initialisation de l'algorithme :
Choix d'un vecteur initial U_0 normalisé.
Calculs de : $\alpha_1 =^t U_0.A.U_0$ et $\beta_2 = \sqrt{^tU_0.A^2.U_0 - \alpha_1^2}$
Le vecteur U_1 est défini par : $\beta_2 U_1 = A.U_0 - \alpha_1 U_0$

VIII.6 METHODE DE BISSECTION

Méthode pour une matrice tridiagonale A.

Appelons a_i, b_i et c_i respectivement les termes diagonaux, supérieurs et inférieurs de la matrice A. La méthode consiste à calculer les valeurs propres à partir du polynôme caractéristique $P_n(\lambda) = \det(A - \lambda I_n)$. Un développement du déterminant par rapport à la dernière ligne permet d'écrire la relation de récurrence suivante :

$$P_n(\lambda) = (a_n - \lambda)P_{n-1}(\lambda) - c_{n-1}b_{n-1}P_{n-2}(\lambda)$$

Le calcul des racines de $P_n(\lambda)$ s'opère de la manière suivante : chercher un intervalle où $P_n(\lambda)$ change de signe et localiser une racine par bissection.

VIII.7 METHODE DE LA PUISSANCE

Méthode itérative pour une matrice quelconque A qui permet d'évaluer son rayon spectral $\rho(A)$.

Le principe de la méthode repose sur le fait qu'en appliquant un grand nombre de fois la matrice sur un vecteur initial quelconque, les vecteurs successifs vont prendre une direction qui se rapproche du vecteur propre de la plus grande valeur propre (en valeur absolue).

L'algorithme :
1) Choisir un vecteur X_0.
2) Itérer : $X_{k+1} = \dfrac{AX_k}{\| AX_k \|}$ jusqu'à convergence $\| X_{k+1} - X_k \| < \varepsilon$.

La suite des vecteurs (X_k) convergent vers le vecteur propre de la plus grande valeur propre et la suite des normes $\| AX_k \|$ converge vers le rayon spectral $\rho(A)$.

Condition de convergence :
Si le vecteur de départ X_0 n'appartient pas au sous-espace vectoriel engendré par $n-1$ vecteurs propres de A alors la méthode converge.

VIII.8 METHODE DE DEFLATION

Méthode itérative pour une matrice quelconque A.

Supposons connu le rayon spectral de A, $\lambda_1 = \rho(A)$ et le vecteur propre correspondant X_1 (calculés par la méthode de la puissance). Il est possible de calculer λ_2, la valeur propre de module immédiatement inférieur à λ_1, et d'autres valeurs propres par la suite.
La méthode consiste à transformer la matrice A en une matrice $A^{(1)}$, ayant les mêmes valeurs propres que A excepté λ_1 remplacée par une valeur propre nulle, puis d'appliquer de nouveau à $A^{(1)}$ la méthode de la puissance.

L'algorithme :
1) Application de la méthode de la puissance pour déterminer λ_1 et X_1.
2) Choix d'un vecteur W tel que $< W, X_1 >= 1$.
3) Calcul de la matrice $A^{(1)} = A - \lambda_1 X_1 \, {}^tW$ qui a les mêmes valeurs propres que A sauf λ_1 remplacée par une valeur propre nulle.
4) Application de la méthode de la puissance à $A^{(1)}$ pour déterminer λ_2 et le vecteur propre associé Y_2 (vecteur propre de $A^{(1)}$). Le vecteur propre de A correspondant est donné par : $X_2 = Y_2 + \dfrac{\lambda_1}{\lambda_2 - \lambda_1} < W, Y_2 > X_1$

En itérant la déflation sur $A^{(1)}$, $A^{(2)}$,..., d'autres valeurs propres de la matrice A peuvent être déterminées.

Bibliographie

[1] J. Baranger. Analyse numérique. Hermann, 1988.

[2] J. Beuneu. Algorithmes pour le calcul scientifique. Cours de l'Ecole Universitaire d'Ingénieurs de Lille, 1999.

[3] J.A. Desideri. Introduction à l'analyse numérique. INRIA, 1998.

[4] E. Hairer. Méthodes numériques. Cours de l'université de Genéve, 2004.

[5] C. Hirsch. Numerical computation of internal and external flows, volume I : fundamentals of numerical discretization. John Wiley & Sons, Chichester, New York, 1986.

[6] P. Lascaux and R. Theodor. Analyse numérique matricielle appliquée à l'art de l'ingénieur, tomes I et II. Masson, Paris, 1986.

[7] V. Legat. Mathématique et méthodes numériques. Cours de l'université catholique de Louvain, 2004.

[8] B. Mohammadi and J.H. Saiac. Pratique de la simulation numérique. Dunod, 2003.

[9] N. Point and J.H. Saiac. Equations aux dérivées partielles - mathématiques et méthodes numériques. Cours de l'ESCPI, 2005.

[10] J. Reveillon. Simulation et modélisation numérique. Cours de l'université de Rouen, 2003.

[11] D. Senechal. Histoire des sciences. Cours de l'université de Sherbrooke, 2004.

[12] A. Quarteroni. Analyse numérique. Cours de l'EPFL, 2003.

[13] P. Viot. Méthodes d'analyse numériques. Cours de DEA de Jussieu, 2003.