Gabriel Stoltz

Introduction au Calcul Scientifique

20 juin 2015

Ecoles des Mines - Tronc commun - 2ème année

Avant-propos

Un ordinateur donne toujours un résultat numérique, mais est-ce le bon ?

Ce cours est, comme son titre l'indique, une introduction au calcul scientifique. Son premier objectif est donc déjà de définir ou au moins de donner une idée de ce qu'est le calcul scientifique, si possible par des exemples précis afin de plus marquer les esprits ; et également de présenter quelques concepts généraux et transversaux, rencontrés dans toutes les applications ou presque du calcul scientifique.

Mettons en garde le lecteur contre le fait que ce cours n'est pas un ènième cours de mathématiques : nous allons nous concentrer sur des aspects pratiques de la science mathématique et de ses applications. Par ailleurs, insistons également sur le fait que nombre d'entre vous seront recrutés précisément pour des compétences telles que celles qui sont évoquées dans ce cours, et ce, quelque soit votre domaine d'activité par la suite : vos employeurs imagineront toujours (à tort, et si possible à raison) que, en tant qu'ingénieur diplômé d'une grande école scientifique, vous saurez établir la pertinence et la crédibilité de résultats et d'études scientifiques dont les rapports stratégiques sont truffés. En particulier, il y aura toujours une étude numérique des impacts ou conséquences envisageables, qui fera intervenir des modèles physiques ou économiques brièvement décrits, et une méthode de simulation subliminalement évoquée. Comment, à l'aune de ces maigres informations, donner une vraie valeur ajoutée aux résultats de simulation en tant que non-expert du domaine ?

Si tout se passe bien, vous saurez à l'issue de cours quel crédit accorder (ou pas) à un résultat numérique, en distinguant les types d'erreur possibles ; connaitrez quelques méthodes numériques employées quotidiennement (de manière explicite ou implicite), leurs succès et leurs limitations ; saurez implémenter informatiquement des algorithmes simples et en éprouver les limites. Du moins, c'est l'objectif le plus ambitieux que l'on puisse se fixer... Il faudra, pour ceux que le domaine intéresse, affiner vos compétences par des modules complémentaires. Pour les autres, il faudra vous souvenir au moins vaguement des limitations évoquées le jour (pas si lointain que ça, croyez-moi) où vous aurez vous-même à produire un résultat numérique pour les besoins d'une étude dans un domaine scientifique non-mathématique, la simulation numérique étant alors, comme souvent, un moyen et non pas une fin.

Aspects pratiques

J'ai essayé de construire ce cours en montrant en amphi beaucoup d'exemples informatiques pratiques. Il est essentiel d'insister sur le fait que le calcul scientifique n'est pas un sport de spectateur ! Il faut pratiquer la simulation et pousser les codes et méthodes dans leurs derniers retranchements pour vraiment comprendre ce qui se passe, ce qui peut être simulable, et ce qui ne l'est pas. Tout cela a toujours l'air beaucoup trop simple lorsque c'est l'enseignant qui le fait... Alors, à vous de jouer !

D'un point-de-vue plus pragmatique, la validation se fait par le rendu en séance de deux compte-rendus de TP (réalisés en binômes), et d'un exercice rédigé en fin de TD (par groupe de 4).

L'équipe enseignante de cette année est composée de Ahmed-Amine Homman, Francois Madiot, Jean-Léopold Vié, tous trois en thèse (respectivement au CEA/DAM, au CERMICS, le laboratoire de mathématiques appliquées de l'Ecole des Ponts, et entre le CMAP de l'Ecole polytechnique, Renault et le CERMICS). Le cours est structuré en trois demi-journées :

(1) Un cours d'introduction générale : démarche générale du calcul scientifique, modélisation, sources d'erreurs, illustrés par des exemples simples (amphi) ; session de TDs (conditionnement, etc) ; erreurs d'arrondis informatique (amphi).

(2) Deux séances spécifiques : calcul d'intégrales (motivé par des applications en physique statistique numérique) et résolution d'équations différentielles ordinaires (pour intégrer par exemple la dynamique des planètes). Ces séances débuteront par un cours d'amphi, et seront suivies d'un TD et d'un TP informatique.

Ce document est la troisième version du poly, et je demande donc votre indulgence pour les nombreuses erreurs que je n'ai toujours pas repérées, et les présentations parfois un peu laconiques ou cavalières de certains concepts. Nul doute que vos remarques acérées me permettront d'améliorer grandement ce document pour vos successeurs !

Table des matières

1

Concepts généraux du calcul scientifique

1.1 Qu'est-ce que le calcul scientifique ?

Le calcul scientifique est une discipline aux contours pas toujours franchement définis, mais qui regroupe un ensemble de champs mathématiques et informatiques permettant la simulation numérique des phénomènes de la physique, chimie, biologie, et sciences appliquées en général.

1.1.1 Anatomie d'un champ scientifique

L'approche d'un problème par le biais du calcul scientifique est une démarche globale, qui se passe en plusieurs temps successifs (avec en pratique des aller-retours d'une étape à l'autre), et dont toutes les étapes sont nécessaires :

(i) cela commence par la modélisation du système, qui consiste à décrire les phénomènes observés par le biais d'équations mathématiques, souvent en collaboration avec les scientifiques des disciplines applicatives concernées ;

(ii) vient ensuite l'analyse théorique du modèle, et l'étude de ses propriétés (existence/unicité de la solution). Ceci peut faire intervenir des résultats profonds d'analyse, de théorie spectrale, de théorie des probabilités, etc ;

(iii) on propose ensuite une méthode numérique adaptée aux propriétés théoriques du modèle (préservant certains invariants par exemple), et on en fait l'analyse numérique. Ceci permet de déterminer la vitesse de convergence de la méthode numérique, sa stabilité. On peut également chercher des estimations d'erreurs *a priori* et *a posteriori* (voir Section 1.2).

(iv) vient enfin l'implémentation informatique de la méthode (avec éventuellement sa parallélisation sur un gros cluster de calcul), et sa validation sur des cas tests académiques pour vérifier le comportement de la méthode dans des situations bien connues ;

(v) si le travail s'arrête souvent là pour les mathématiciens, c'est en revanche à ce stade que commence la vraie aventure scientifique pour les chercheurs ou ingénieurs des domaines d'application, qui vont utiliser la nouvelle méthode sur des cas réels (et possiblement jusque dans ses derniers retranchements).

Comme cette sommaire description l'indique, le calcul scientifique est par essence un domaine interdisciplinaire, tant au sein des sciences en général qu'au sein des mathématiques (puisqu'il repose sur des champs aussi divers que l'analyse, l'analyse numérique, la théorie des probabilités, etc). Il est donc important d'avoir une bonne culture scientifique générale pour travailler dans ce domaine, et les étudiants des écoles d'ingénieurs françaises sont particulièrement qualifiés pour cela !

1.1.2 Moyen d'action : les méthodes numériques

Au coeur d'une démarche de calcul scientifique se trouve une méthode numérique, qui permet de calculer de manière approchée une propriété d'intérêt. Une telle méthode est fondée sur un algorithme implémenté informatiquement, son bras armé en quelque sorte. Un algorithme est une suite de tâches élémentaires qui s'enchaînent selon des règles précises, et qui est exécuté automatiquement par un langage informatique. Ce n'est pas une recette de cuisine ! Un élément important d'un algorithme est sa complexité, qui est une mesure du temps d'exécution. On évalue en particulier le nombre d'opérations arithmétiques élémentaires et le coût du stockage en mémoire.

1.1.3 Exemples d'applications

Un premier champ d'action pour le calcul scientifique concerne les situations où l'on ne peut pas (complètement) réaliser une expérience : ce qui se passe dans une centrale nucléaire qui s'emballe, la résistance de ladite centrale à un crash d'avion, la simulation du fonctionnement des armes nucléaires en l'absence d'essais, la conception d'un centre de stockage définitif des déchets nucléaires, le calcul de trajectoires de satellites, fusées. Dans toutes ces situations, une bonne simulation numérique permettra de donner quelques indications sur le comportement attendu de l'objet de la simulation – sans garantie totale que tout se passe comme prévu, bien sûr...

Il y a également des situations où il est moins cher de réaliser des tests numériques préliminaires : la simulation moléculaire des principes actifs pour l'industrie pharmaceutique, les tests de résistance mécanique (crash tests) dans l'industrie automobile, la synthèse de nouveaux matériaux pour l'industrie (alliages, polymères). Dans ces cas, la simulation numérique ne remplace pas une expérience, mais elle la complète, en suggérant des processus ou des comportements à tester spécifiquement de manière expérimentale.

Signalons enfin les situations où l'on cherche à anticiper des événements par le biais de la simulation. Cela concerne les méthodes numériques pour la finance (trading), et plus traditionnellement, la prévision météorologique ou climatique.

1.1.4 Objectifs du calcul scientifique

En fonction du problème que l'on cherche à résoudre, on peut avoir des objectifs différents. Pour mettre un peu de corps sur une phrase aussi générique, distinguons quatre points de vue :

(i) on peut souhaiter assurer la convergence de la méthode numérique : l'erreur sur le résultat final peut être rendue arbitrairement petite en y mettant les moyens ;

(ii) on peut lui préférer la précision : assurer que les erreurs que l'on commet sont petites par rappport à une tolérance fixée ;

(iii) on peut plutôt privilégier la fiabilité, qui est moins contraignante que la précision. Dans ce cas, on souhaite simplement garantir que l'erreur globale est en dessous d'une certaine tolérance. Ceci demande typiquement une validation de la méthode sur des cas tests ;

(iv) on peut enfin se concentrer sur l'efficacité de la méthode, et assurer que son coût de calcul est aussi petit que possible.

Evidemment, on souhaiterait avoir des résultats aussi convergés que possibles, et qui soient fiables dans tous les cas. Ce n'est cependant pas toujours possible, notamment lorsque l'on cherche à faire des estimations en temps réel (ou presque). Dans ce dernier cas, une fiabilité minimale mais une efficacité maximale seront plus opportunes.

1.2 Les différentes sources d'erreur

1.2.1 Formulation abstraite du problème

Commençons par formuler de manière abstraite les problèmes du calcul scientifique sous la forme : chercher x tel que, pour des données d, on ait

$$F(x, d) = 0.$$

L'inconnue x et les données d peuvent être des nombres, des vecteurs, des fonctions, etc. On notera par la suite D l'ensemble des données possibles, et X l'espace dans lequel on cherche les solutions. Dans les exemples ci-dessous, le lecteur est incité à préciser qui sont x et d, ainsi que leurs domaines de définition.

Exemple 1.1. Chercher les racines de $p(t) = a_2t^2 + a_1t + a_0$.

Exemple 1.2. Trouver une fonction u à valeurs dans $\mathbb{R}^m$ telle que $-\mathrm{div}(A\nabla u) = f$ où $A(x)$ est une matrice symétrique réelle de taille $m \times m$ avec $A \geqslant \alpha > 0$ (au sens des matrices symétriques définies positives) et f est un forçage donné.

Exemple 1.3. Calculer $y(T)$ où $y(t)$ vérifie $\dot{y}(t) = g(t, y(t))$ avec $y(0) = 1$.

La première question que l'on doit se poser est la suivante : le problème $F(x, d) = 0$ est-il bien posé ? Cela demande de vérifier que la solution x existe, est unique, et dépend continûment des données, au sens suivant : pour d donné, il existe $K > 0$ et $\varepsilon > 0$ tels que, si $\|\delta d\| \leqslant \varepsilon$, alors $\|\delta x\| \leqslant K\varepsilon$ (avec $\delta x = x(d + \delta d) - x(d)$). Insistons lourdement sur le fait qu'une méthode numérique ne peut pas résoudre un problème mal posé (il faut le régulariser dans ce cas).

Exemple 1.4. Chercher le nombre de racines réelles de $t^4 - (2a-1)t^2 + a(a-1)$ est un problème mal posé ! En effet, on a 4 racines si $a \geqslant 1$, 2 si $a \in [0, 1[$, et aucune si $a < 0$. Le nombre de racines (un élément de $\mathbb{N}$) ne dépend pas continûment de la donnée $a \in \mathbb{R}$.

1.2.2 Analyse des sources d'erreurs

Le premier objectif à atteindre pour analyser les erreurs d'une simulation numérique est déjà de reconnaitre les différentes sources d'erreurs possibles ! On peut penser déjà aux erreurs dues à la modélisation mathématique du problème, résultants d'approximations dans la physique du problème – par exemple, négliger la viscosité et travailler avec les équations d'Euler plutôt que Navier-Stokes si le fluide est peu visqueux. Ces erreurs peuvent être évaluées lors de discussions avec les scientifiques des domaines applicatifs concernés.

On distingue également les erreurs dans les données d'entrée du problème : paramètres estimés par une mesure expérimentale ou par un autre modèle mathématique, conditions initiales, etc.

Vous n'y pouvez rien *a priori*, mais il faut quand même faire quelque chose, ne serait-ce qu'étudier comment les incertitudes sur les données d'entrée se répercutent sur les données de sortie (valeurs des inconnues).

Mentionnons enfin les erreurs dans les algorithmes et méthodes numériques que l'on utilise : en pratique, on résout un problème approché $F_n(x_n, d_n) = 0$, l'approximation résultant par exemple de la discrétisation d'un problème continu. Dans cette situation, nous avons des outils pour nous aider à quantifier précisément les erreurs introduites :

(a) les erreurs d'arrondi dues à la représentation machine des nombres et aux opérations arithmétiques effectuées (voir Section 1.3) ;

(b) les erreurs d'approximation des méthodes numériques, qui est le gros du travail pour un mathématicien appliqué. C'est le domaine par excellence de l'analyse numérique ;

(c) ne pas oublier les erreurs humaines... Mêmes développées et implémentées par des professionnels qualifiés, il se peut que les méthodes numériques soient entâchées d'un bug interne ! Pour éviter cela, on ne peut que recommander la validation des résultats de simulation par des approches variées et complémentaires.

Cette section décrit les concepts généraux associés à la discussion précédente sur les erreurs. Nous allons donc successivement évoquer le caractère bien posé des problèmes et notamment étudier leur conditionnement ; puis définir les notions de stabilité (robustesse face aux perturbations) et de convergence ; et finir en présentant l'analyse *a priori* (directe et rétrograde) et *a posteriori* des erreurs de simulation.

1.2.3 Les erreurs dans les données : le conditionnement

Des erreurs sur les données d'entrées du problèmes sont inévitables, ne serait-ce que du fait des imprécisions dans les mesures physiques (auquel cas ces erreurs peuvent être systématiques ou aléatoires), et de la troncature des nombres dans la représentation machine. La question importante est de savoir si ces erreurs sont propagées, et si elles sont amplifiées ou pas. Notons que cette amplification peut être intrinsèque au modèle mathématique continu, et l'amplification liée à la méthode numérique sera au mieux du même ordre en général.

Un moyen de quantifier cette propagation des erreurs est de calculer le conditionnement du problème. On considère pour cela une perturbation des données $d+\delta d \in D$ et on regarde l'inconnue $x+\delta x$ telle que $F(x+\delta x, d+\delta d) = 0$. Le conditionnement relatif, pour une donnée d fixée, mesure la variation relative des inconnues pour une variation relative des données :

$$K_{\text{rel}}(d) = \limsup_{\delta d \to 0} \left\{ \frac{\|\delta x\|_X / \|x\|_X}{\|\delta d\|_D / \|d\|_D} \right\}.$$

Dans certaines situations, on ne peut pas utiliser cette notion, notamment lorsque $x = 0$. On utilise dans ce cas le conditionnement absolu :

$$K_{\text{abs}}(d) = \limsup_{\delta d \to 0} \left\{ \frac{\|\delta x\|_X}{\|\delta d\|_D} \right\}$$

On dit qu'un problème est mal conditionné si $K(d)$ est "grand". Les guillemets indiquent le flou de cette notion en général... Pour donner un sens précis à cette affirmation, il faut donner des ordres de grandeur typiques, qui sont contingents au problème considéré. Notons également qu'une reformulation du problème (par le biais d'un changement de variable) peut réduire ou augmenter le conditionnement.

Exemple 1.5 (Résolution d'un système linéaire d'équation). On considère un système d'équations $Ax = b$ avec une matrice $A \in \mathbb{R}^{m\times m}$ inversible et $b \in \mathbb{R}^m \backslash \{0\}$. Les données de ce problèmes sont la matrice A et le second membre b. On se limite à des perturbations du second membre seulement. Dans ce cas, en utilisant la forme explicite de la solution, à savoir, $x = A^{-1}b$, on obtient le conditionnement relatif

$$K_{\rm rel}(b) = \limsup_{\delta b \to 0} \frac{\|A^{-1}\delta b\|/\|A^{-1}b\|}{\|\delta b\|/\|b\|} \leqslant \sup_{\delta b \neq 0} \frac{\|A^{-1}\delta b\|\,\|Ax\|}{\|\delta b\|\,\|x\|} \leqslant \|A^{-1}\|\|A\| = \kappa(A),$$

où on a noté $\|A\|$ la norme matricielle induite par la norme $\|\cdot\|$ sur $\mathbb{R}^m$:

$$\|A\| = \sup_{y \neq 0} \frac{\|Ay\|}{\|y\|}.$$

Noter que $\kappa(A) \geqslant \|AA^{-1}\| = 1$ dans tous les cas.

Exercice 1.1. *On se place dans le cadre de l'exemple 1.5. Choisissons maintenant la norme euclidienne et supposons que A est symétrique positive. Montrer que le conditionnement $\kappa(A) = \|A\|\|A^{-1}\|$ d'une matrice inversible symétrique réelle est le rapport des valeurs propres extrêmes (en module). Etablir également que $\kappa(\alpha A) = \kappa(A)$ pour tout $\alpha \in \mathbb{R}^*$.*

Exercice 1.2 (Conditionnement d'une solution d'équation différentielle). *On considère le problème suivant : pour $T > 0$ et $(\alpha, y^0) \in \mathbb{R}^2$ donnés, calculer la valeur $y(T)$ de la solution de l'équation différentielle ordinaire*

$$\frac{dy(t)}{dt} = \alpha y(t),$$

avec la condition initiale $y(0) = y^0$. Calculer les conditionnements relatif et absolu de l'application $y^0 \mapsto y(T)$, ainsi que de l'application $\alpha \mapsto y(T)$. Discuter les valeurs des paramètres pour lesquelles le problème est bien conditionné.

Exercice 1.3 (Evaluation approchée d'une fonction). *On souhaite calculer le plus précisément possible $f(\sqrt{2})$ pour $f(x) = (x-1)^6$, en utilisant la valeur approchée* 1.4 *(voire 1 !) pour $\sqrt{2}$. Parmi les expressions suivantes (qui sont mathématiquement équivalentes !),*

$$\left(\frac{1}{\sqrt{2}+1}\right)^6;\ (3-2\sqrt{2})^3;\ \left(\frac{1}{3+2\sqrt{2}}\right)^3;\ 99-70\sqrt{2};\ \frac{1}{99+70\sqrt{2}};$$

laquelle donne le meilleur résultat ? Et laquelle donne le pire ? Motiver ces observations par un calcul de conditionnement, en montrant au préalable que le conditionnement relatif lié au calcul de $f(x_0)$ est

$$K_{\rm rel}(x_0) = \left|\frac{x_0 f'(x_0)}{f(x_0)}\right|.$$

Exercice 1.4 (Compensation des erreurs). *On veut calculer $y = z_1 + z_2$ avec $z_1 = \sqrt{x^2+1}$ et $z_2 = 200 - x$, avec une valeur de $x = 100 \pm 1$. Estimer les erreurs sur z_1, z_2 et sur y.*

1.2.4 Les erreurs d'approximation

Discutons à présent les erreurs d'approximation, celles qui résultent directement de la conception des méthodes numériques.

Consistance

Rappelons pour commencer qu'approcher numériquement la solution de $F(x,d) = 0$ se fait en résolvant un problème numérique $F_n(x_n, d_n) = 0$. On s'attend à ce que $x_n \to x$ en un certain sens. Pour cela, il faut que $d_n \to d$ et que F_n "approche" F : c'est précisément la notion de consistance. Insistons sur le fait que les données sont également approchées en général (par exemple, une donnée d'entrée qui est une fonction devra nécessairement, d'un point de vue informatique, être représentée de manière discrète, par exemple par ses valeurs en certains points). On notera D_n l'espace des données numériques possibles.

Pour donner corps à la discussion qui va suivre, présentons quelques méthodes numériques très simples :

- le calcul de l'intégrale $\displaystyle\int_0^1 h(x)\,dx$ peut être approché par la somme discrète $\displaystyle\frac{1}{n+1}\sum_{i=0}^{n} h\left(\frac{i}{n}\right)$;
- la recherche de solutions de l'équation $f(x) = 0$ pour $f : \mathbb{R} \to \mathbb{R}$ peut se faire avec une méthode itérative de type Newton : $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$. Dans ce cas, la méthode numérique est arrêtée au bout d'un nombre fini n de pas, ce qu'on peut écrire comme $F_n(x_n, x_{n-1}, \ldots, x_0, d) = 0$.

Définition 1.1 (Consistance). *On dit qu'une méthode numérique est consistante si $F_n(x, d) \to 0$ lorsque $n \to +\infty$; et fortement consistance si $F_n(x, d) = 0$ pour tout n.*

Exemple 1.6. L'approximation d'une intégrale par une somme de Riemman est une méthode numérique consistante ; alors que la méthode de Newton est fortement consistante.

Stabilité

Considérons une méthode numérique bien posée (pour laquelle on a existence et unicité de la solution, et une dépendance continue de la solution par rapport aux données). Si on considère une perturbation δd_n de la donnée numérique d_n, telle que $d_n + \delta d_n \in D_n$, alors la solution x_n est également perturbée, et devient $x_n + \delta x_n$ (définie par le fait que $F_n(x_n + \delta x_n, d_n + \delta d_n) = 0$). On voudrait assurer une certaine forme de robustesse ou stabilité par rapport aux perturbations. Mathématiquement, la notion de stabilité signifie que, pour tout $\eta > 0$ il existe $M > 0$ tel que, pour une perturbation $\|\delta d_n\| \leqslant \eta$, on a

$$\|\delta x_n\| \leqslant M\|\delta d_n\|. \tag{1.1}$$

Notons que η et M dépendent de d_n en général. Cette notion peut être quantifiée par le conditionnement de la méthode numérique (relatif, ici) :

$$K_{\mathrm{rel}}^{\mathrm{num}}(d) = \lim_{k\to+\infty} \sup_{n\geqslant k} \left(\limsup \left\{ \frac{\|\delta x_n\|/\|x_n\|}{\|\delta d_n\|/\|d_n\|} \;\middle|\; \delta d_n \to 0,\ d + \delta d_n \in D_n \right\} \right).$$

Convergence

Une méthode numérique est convergente si, pour tout $\varepsilon > 0$, il existe $N \geqslant 1$ et $\eta > 0$ tels que, pour tout $n \geqslant N$,

$$\|\delta d_n\| \leqslant \eta \Rightarrow \|x(d) - x_n(d + \delta d_n)\| \leqslant \varepsilon.$$

Comme $F(x, d)$ est un problème bien posé, il suffit en fait de montrer des estimations du type

$$\|x(d + \delta d_n) - x_n(d + \delta d_n)\| \leqslant \frac{\varepsilon}{2} \tag{1.2}$$

pour $\|\delta d_n\| \leqslant \eta$ assez petit.

Conditions nécessaires et suffisantes de convergence

La stabilité est une condition nécessaire de convergence : on ne considérera donc par la suite que des méthodes numériques stables ! En effet, supposons que la méthode numérique soit convergente. On a alors

$$\begin{aligned}\|\delta x_n\| &= \|x_n(d + \delta d_n) - x_n(d)\| \\ &\leqslant \|x_n(d) - x(d)\| + \|x(d) - x(d + \delta d_n)\| + \|x(d + \delta d_n) - x_n(d + \delta d_n)\|.\end{aligned}$$

Fixons $0 < \varepsilon \leqslant \|\delta d_n\| \leqslant \eta$. Pour n assez grand, le premier et le dernier termes sont plus petits que $\varepsilon/2$ par convergence ; alors que, par continuité de la solution en fonction des données, le

second terme est plus petit que $K\eta$ pour une certaine constante $K > 0$. Au final, on a (1.1) avec $M = K + 1$.

Un second résultat, très important, est le suivant : la stabilité et la consistance impliquent la convergence. C'est un méta-théorème de l'analyse numérique, qui doit cependant être vérifié au cas par cas. Esquissons la structure de la preuve (nous verrons un cas particulier en Section 3.3.2). On a

$$\|x(d) - x_n(d + \delta d_n)\| \leqslant \|x_n(d) - x_n(d + \delta d_n)\| + \|x(d) - x_n(d)\|.$$

Le premier terme peut se majorer par $M\|\delta d_n\|$ (par stabilité). Pour le dernier terme, on a formellement

$$F_n(x(d), d) - F_n(x_n(d), d) = \left.\frac{\partial F_n}{\partial x}\right|_{(x^*,d)} \Big(x(d) - x_n(d)\Big)$$

pour un certain x^*, d'où on déduit que

$$\|x(d) - x_n(d)\| \leqslant \left\| \left.\left(\frac{\partial F_n}{\partial x}\right)^{-1}\right|_{(x^*,d)} \right\| \|F_n(x(d), d) - F_n(x_n(d), d)\|.$$

On conclut en remarquant que le dernier facteur du membre de droite tend vers 0 par consistance (noter que $F_n(x_n(d), d) = 0$). C'est bien sûr cette dernière étape de la majoration, tout à fait formelle ici, qu'il s'agit de rendre rigoureuse au cas par cas.

1.2.5 Analyse d'erreur

Nous verrons précisément comment on peut analyser l'erreur d'approximation d'une méthode numérique dans les chapitres 2 et 3. Décrivons simplement ici l'esprit des techniques mathématiques utilisées :

(a) l'analyse *a priori* directe permet de répondre à la question suivante : à quel point la solution calculée est-elle proche de la solution exacte ? Cela demande d'estimer ou d'obtenir des bornes sur $\delta x_n = x_n(d) - x(d)$ en fonction de $d_n - d$ et des erreurs de la méthode numérique (paramètre n). Un exemple d'application est de majorer la perturbation de la solution d'une EDO en fonction d'une perturbation de la condition initiale ;

(b) l'analyse *a priori* rétrograde repose sur une vision différente : on se demande plutôt à quel point la solution calculée satisfait le problème initial. Cela demande d'obtenir des bornes sur la perturbation δd_n à appliquer pour que $F(x_n, d + \delta d_n) = 0$. Au lieu de voir la solution numérique comme la solution approchée du problème (exact) de départ, on la considère comme la solution d'un problème approché (ou voisin) ;

(c) enfin, l'analyse *a posteriori* permet d'estimer de l'erreur $x - x_n$ à partir des quantités effectivement calculées (telles que le résidu $r_n = F(x_n, d)$). Cela permet de songer à des stratégies de contrôle d'erreur adaptatif (rétroaction), où on va ajuster n pour obtenir la précision souhaitée.

Illustrons à présent l'esprit de ces différents types d'analyse sur des exemples.

Exemple 1.7. Pour le calcul des racines α_i du polynôme $p(t) = \sum_{m=0}^{M} a_m t^m$, l'analyse *a priori* directe estimerait l'erreur entre les racines calculées numériquement (notées $\widehat{\alpha}_i$ par la suite) et les racines exactes en fonction de l'erreur sur les coefficients du polynôme et possiblement des erreurs de la méthode numérique. L'analyse *a priori* rétrograde estimerait quant à elle les perturbations δa_m à apporter aux coefficients du polynôme pour que les racines effectivement calculées $\widehat{\alpha}_i$ soient les solutions exactes α_i. Enfin, la tâche de l'analyse *a posteriori* serait d'estimer $\alpha_i - \widehat{\alpha}_i$ en fonction de $p_M(\widehat{\alpha}_i)$.

Exemple 1.8. Considérons à présent la résolution du système linéaire $Ax = b$ (où A est une matrice inversible de taille $m \times m$) par une méthode itérative. On note $\widehat{x}_n$ la solution calculée numériquement (par exemple au bout de n itérations). L'erreur *a priori* directe demande de contrôler $\|x - \widehat{x}_n\|$ en fonction de n, b et A. L'erreur *a priori* rétrograde est obtenue en écrivant $A\widehat{x}_n = b + \delta b_n$ ou $(A + \delta A_n)\widehat{x}_n = b$, et en estimant $\|\delta b_n\|$ ou $\|\delta A_n\|$. L'erreur *a posteriori* demande d'estimer $x - \widehat{x}_n$ en fonction de $b - A\widehat{x_n}$.

Exercice 1.5 (Calcul de la plus grande valeur propre d'une matrice). *On cherche une méthode numérique permettant de calculer la plus grande valeur propre d'une matrice, sans la diagonaliser (une opération très coûteuse...). On considère une matrice symétrique, définie positive $A \in \mathbb{R}^{m\times m}$, et on munit $\mathbb{R}^m$ de la norme euclidienne. On note $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_N$ ses valeurs propres (avec $x_1, \ldots, x_N$ les vecteurs propres associés) et on suppose que $\lambda_1 > \lambda_2$. On va commencer par établir deux estimations (une a priori, une a posteriori), avant d'analyser la convergence d'une méthode particulière.*

(i) Soit E une matrice symétrique réelle. On note $\mu_1, \ldots, \mu_N$ les valeurs propres de $A + E$. Montrer l'estimation a priori *:*

$$\max_{1\leqslant j\leqslant N} \min_{1\leqslant i\leqslant N} \{|\mu_j - \lambda_i|\} \leqslant \|E\|.$$

(ii) Soient $\widehat{x}, \widehat{\lambda}$ un vecteur propre et une valeur propre calculés par une méthode numérique (comme celle que vous allez étudier ci-dessous...). On note $\widehat{r} = A\widehat{x} - \widehat{\lambda}\widehat{x}$ le résidu. Montrer l'estimation a posteriori

$$\min_{1\leqslant i\leqslant N} \left\{\left|\widehat{\lambda} - \lambda_i\right|\right\} \leqslant \frac{\|\widehat{r}\|}{\|\widehat{x}\|}.$$

(iii) On considère la méthode numérique suivante (méthode de la puissance) : pour q^0 donné tel que $\alpha_1 = x_1^T q^0 \neq 0$, et pour $n \geqslant 1$,

$$q^n = \frac{Aq^{n-1}}{\|Aq^{n-1}\|}, \qquad \mu^n = (q^n)^T A q^n.$$

Montrer que $q^n \to \mathrm{sign}(\alpha_1)x_1$ et $\mu^n \to \lambda_1$ en précisant le taux de convergence.

1.3 Arithmétique des nombres flottants

On va étudier dans cette section les erreurs numériques directement liées à la représentation des nombres réels dans nos ordinateurs. C'est une erreur assez subtile et qui est souvent négligée, voire totalement ignorée dans la majorité des cas. Et pourtant, il est fondamental de comprendre que certaines opérations arithmétiques vont bien se passer alors que d'autres vont avoir des conséquences catastrophiques... On va présenter les idées les plus simples, et on renvoie lecteur à [3] pour une présentation plus complète (et très pédagogique).

1.3.1 Les erreurs d'arrondi, est-ce vraiment important ?

Toute opération effectuée par un ordinateur est entachée par des erreurs d'arrondis – à quel point ? Parfois beaucoup... Donnons quelques exemples historiques importants pour motiver le sujet :

- à la la bourse de Vancouver, du fait d'une troncature au lieu d'un arrondi au plus proche sur le dernier chiffre significatif, on a vu, en quelques mois, l'index passer de 1000 à 520, alors qu'une règle d'arrondi correcte donnait une indice de 1089 (voir l'analyse dans [6]) ;
- dans un célèbre accident dû à un mauvais tir de missile Patriot, un facteur de conversion de 0.1 représenté approximativement sur 24 bits a donné lieu à un décalage d'horloge de 0.3433 s au bout de 100 h, soit environ 500 m à la vitesse de Mach 5 (voir l'article [9]) ;
- le vol Ariane 5 du 4 juin 1996 a explosé du fait d'un overflow lié à une conversion malheureuse *64-bit floating point* vers *16-bit signed integer* (voir par exemple [7]) ;

D'autres exemples sont présentés interactivement lors de l'amphi, par exemple le calcul de la variance d'un ensemble de N valeurs $\{y_1, \ldots, y_N\}$. Il existe deux formules mathématiquement équivalentes pour calculer la variance :

$$\sigma_1 = \frac{1}{N-1}\sum_{i=1}^{N} y_i^2 - \left(\frac{1}{N}\sum_{i=1}^{N} y_i\right)^2, \qquad \sigma_2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(y_i - \frac{1}{N}\sum_{k=1}^{N} y_k\right)^2 ;$$

mais qui ont des comportements bien différents d'un point-de-vue numérique...

1.3.2 Représentation des nombres en machine

La première remarque est qu'on ne peut représenter qu'un sous-ensemble fini de $\mathbb{R}$. Ces nombres sont représentés avec une notation positionnelle, dans une base $\beta \geqslant 2$:

$$x = (-1)^s \sum_{k=-m}^{n} x_k \beta^k, \qquad s = 0, 1,$$

ce qu'on peut écrire

$$x_\beta = (-1)^s [x_n\, x_{n-1} \ldots x_1\, x_0\, .\, x_{-1} \ldots x_{-m}], \qquad x_n \neq 0$$

En pratique, on emploie souvent la base $\beta = 2$.

Travailler avec un format fixe, et notamment une virgule fixe, limite énormément la magnitude des nombres que l'on peut représenter, ou alors demande une mémoire excessive. On emploie donc en pratique une représentation en virgule flottante :

$$x = (-1)^s \cdot (0.a_1\, a_2 \ldots a_t) \cdot \beta^e = (-1)^s \cdot m \cdot \beta^{e-t}$$

où t est le nombre de chiffres significatifs, e est l'exposant, et m la mantisse. Les chiffres a_i sont tels que $0 \leqslant a_i \leqslant \beta - 1$ avec $a_1 \neq 0$. On a également des bornes sur les exposants : $L \leqslant e \leqslant U$. Dans cette représentation, la précision relative est

$$\frac{\Delta x}{x} = \frac{\Delta m}{m} \leqslant \frac{1}{2} \varepsilon_{\text{machine}} = \frac{1}{2} \beta^{1-t}.$$

Les deux formats les plus standards à ce jour sont définis dans la norme IEEE 754 de 1985 :

- nombres en précision simple (*float*). On utilise pour ce faire un codage sur 32 bits : 1 pour le signe, 8 pour l'exposant dont un bit de signe, 23 pour la mantisse (voir la Figure 1.1). Dans ce cas, le plus petit nombre qui peut être représenté est $x_{\min} \simeq 10^{-38}$, et le plus grand $x_{\max} = 10^{38}$. La précision relative est de 10^{-7} ;

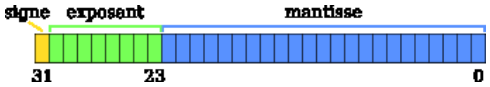

Fig. 1.1. Représentation schématique d'un nombre en précision simple (*float*).

- nombres en double précision (*double*), via un codage sur 64 bits : 1 pour le signe, 11 pour l'exposant dont un bit de signe, 52 pour la mantisse (voir la Figure 1.2). Dans ce cas, $x_{\min} \simeq 10^{-308}$, $x_{\max} = 10^{308}$, et $\varepsilon_{\text{machine}} \simeq 10^{-16}$.

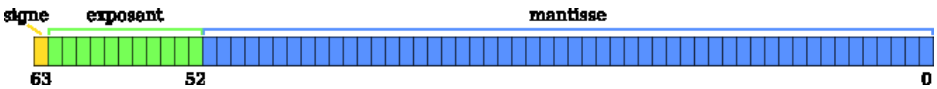

Fig. 1.2. Représentation schématique d'un nombre en double précision (*double*).

Dans tous les cas, le cardinal de l'ensemble $\mathbb{F}$ des nombres que l'on peut représenter est $2(\beta - 1)\beta^{t-1}(U - L + 1) < +\infty$. En effet, le facteur 2 tient compte du signe, le facteur $\beta - 1$ correspond aux valeurs possibles de a_1 (rappelons que ce chiffre doit être non nul) alors qu'il y a β^{t-1} choix pour les autres chiffres de m ; enfin, l'exposant peut prendre $U - L + 1$ valeurs. Notons également que ces nombres ne sont pas répartis uniformément !

Pour assigner de manière univoque un élément de $\mathbb{F}$ à tout nombre réel, on utilise la règle d'arrondi suivante (bien noter que ce n'est pas une troncature) : étant donné $x \in [x_{\min}, x_{\max}]$, que l'on peut écrire comme la somme infinie

$$x = \beta^{e-1} \sum_{i=0}^{+\infty} a_i \beta^{-i},$$

on définit

$$\mathrm{fl}(x) = (-1)^s \cdot (0.a_1\, a_2 \ldots \widetilde{a}_t) \cdot \beta^e, \qquad \widetilde{a}_t = \begin{cases} a_t & \text{si } a_{t+1} < \beta/2 \\ a_t + 1 & \text{si } a_{t+1} \geqslant \beta/2 \end{cases}.$$

Insistons sur la limitation $x \in [x_{\min}, x_{\max}]$. Si cette condition n'est pas vérifiée, on parle d'*overflow* lorsque $|x| > x_{\max}$, et d'*underflow* pour $|x| < x_{\min}$.

L'application fl est monotone : si $x \leqslant y$ alors $\mathrm{fl}(x) \leqslant \mathrm{fl}(y)$. Une autre propriété importante est que

$$\frac{|\mathrm{fl}(x) - x|}{|x|} \leqslant \frac{1}{2} \varepsilon_{\text{machine}}$$

si $x_{\min} \leqslant |x| \leqslant x_{\max}$.

Il existe plein d'autres règles supplémentaires pour gérer les exceptions et les nombres hors de la plage des nombres représentables (NaN, nombres dénormalisés, etc). On se reportera à la bibliographie pour plus de précisions.

1.3.3 Arithmétique machine

Nous allons à présent définir les opérations élémentaires $\widehat{\circ} : \mathbb{R} \times \mathbb{R} \to \mathbb{F}$ (où $\circ \in \{+, -, \times, /\}$) par l'égalité

$$x \,\widehat{\circ}\, y = \mathrm{fl}\Big(\mathrm{fl}(x) \circ \mathrm{fl}(y)\Big).$$

On voit que cela revient à transformer chacun des nombres réel dans la représentation machine, à effectuer l'opération souhaitée, puis à remettre le résultat dans la représentation machine. Cette dernière opération est vraiment nécessaire ! Les différents arrondis intervenant dans la fonction de représentation fl sont à la l'origine de la perte de certaines propriétés des opérations arithmétiques standards, telle que l'associativité. On peut en donner un exemple simple : si a est un "grand" nombre et $b > 0$ un petit "nombre", tel que $a \widehat{+} b = a$ (par exemple parce que b est plus petit que le dernier chiffre significatif de a), alors

$$\big(\ldots \big(\big(a \widehat{+} b\big) + b\big) \widehat{+} \ldots\big) \widehat{+} b = a,$$

alors que, en effectuant d'abord les additions des nombres b suffisamment de fois (en gros, a/b fois),

$$a \widehat{+} \big(b \widehat{+} \big(b \widehat{+} \ldots\big)\big) > a.$$

Noter également que l'on peut perdre des chiffres significatifs lors de ces opérations. Un cas d'école est celui où l'on calcule $a - b$ avec a, b très proches.

Malgré ces désagréments, on conserve certaines propriétés de stabilité, notamment

$$x \,\widehat{\circ}\, y = (x \circ y)(1 + \delta) \tag{1.3}$$

avec $|\delta| \simeq \varepsilon_{\text{machine}}$. Faisons la preuve pour l'addition. On considère $(x, y) \in \mathbb{R}^2$, et on note $\Delta(x \circ y) = x \,\widehat{\circ}\, y - x \circ y$. On introduit $x' = \mathrm{fl}(x)$ et $y' = \mathrm{fl}(y)$, avec $|x' - x| \leqslant \varepsilon_{\text{machine}}|x|$. On décompose la somme $x + y$ comme $x + y = (x - x') + (x' + y') + (y - y')$. L'objectif est de montrer que $\Delta(x' + y') \leqslant \varepsilon_{\text{machine}}(|x| + |y|)$. Pour ce faire, on écrit (en supposant $|x| \geqslant |y|$, ce qui est toujours possible quitte à échanger x et y)

$$x = 0, a_1\, a_2 \ldots a_t \cdot \beta^p, \quad y = 0, b_1\, b_2 \ldots b_t \cdot \beta^q, \quad p \geqslant q.$$

On voit donc que l'on perd les $p - q$ derniers chiffres de y (correspondants aux termes $\leqslant \beta^{-t+p}$), d'où $|\Delta(x' + y')| \leqslant \beta^{-t+p}$; alors que par ailleurs $|x| \sim \beta^{-1+p}$.

Ce genre de calculs permet de montrer la propriété (1.3) pour les autres opérations. Notons toutefois (et c'est un point sur lequel nous avons glissé) qu'il faut en fait un chiffre supplémentaire pour des opérations quasi-dégénérées, appelé chiffre de garde. Dans le cas de l'addition, cela correspond au cas où $x + y \simeq 0$.

1.3.4 En conclusion

On retiendra de cette section qu'il faut faire attention aux opérations arithmétiques que l'on fait, et à l'ordre dans lequel on les fait... Il y a trois écueils principaux :

(1) l'accumulation des erreurs d'arrondi (qui est souvent un très petit effet, sauf si les sommes contiennent vraiment beaucoup de termes) ;

(2) la perte de chiffres significatifs par compensation (soustraction, renormalisation) ;

(3) l'amplification des erreurs d'arrondi par un mauvais conditionnement du problème.

En particulier, on ne peut jamais faire complètement confiance à un résultat numérique dont on ne sait pas comment il a été produit ![1] Il est toutefois possible de réduire ces problèmes en modifiant les algorithmes utilisés. Donnons un exemple simple pour l'écueil (ii) : le calcul des racines de $x^2 - 2px + 1 = 0$ pour p grand, les racines étant $x_\pm = p \pm \sqrt{p^2 - 1}$. On privilégie dans ce cas le calcul de x_- par $x_- = 1/x_+$ et non pas $p - \sqrt{p^2 - 1}$, qui correspond à une soustraction de deux grands nombres du même ordre de grandeur, et donc à une perte de précision.

Il y a bien sûr plein de détails techniques supplémentaires, dont nous n'avons pas parlé dans cette première approche – et notamment les effets liés au compilateur et aux fonctions standards tabulées (log, exp, etc). Le lecteur intéressé se reportera avec plaisir à [3] (et aux références citées par cet ouvrage) pour tous ces passionnants aspects, et bien d'autres.

1. C'est une leçon important pour ceux et celles d'entre vous qui souhaitent exercer des activités de conseil : il y a aura toujours une étude numérique, on vous en donnera les conclusions, et vous serez peut-être la caution scientifique devant approuver les résultats... et il est bon dans ces cas de se souvenir de ce qui peut ne pas marcher.

2

Intégration numérique

Il est nécessaire, dans beaucoup de domaines applicatifs, de calculer numériquement des intégrales. Ce travail ne semble pas passionnant *a priori*, d'autant plus que les logiciels de calcul scientifique tels que Matlab ou Scilab font très bien tout ça tout seul dans la majorité des cas... Oui, mais justement : il peut arriver que les solveurs par défaut n'arrivent pas à calculer l'intégrale voulue, ou alors, de manière moins dramatique, que l'on se demande quelle est la fiabilité du résultat. Il est donc important de comprendre comment est calculée l'approximation d'une intégrale. On peut également motiver la pertinence de l'intégration numérique par des applications actuelles et dynamiques, comme le calcul de quantités moyennes en physique statistique numérique (voir Section 2.1).

Des méthodes déterministes simples sont présentées en Section 2.2 (voir [2, 8]). Elles permettent un calcul précis en petite dimension. Dans la majorité des cas, la brique de base est une formule de quadrature interpolaire, utilisant des points équirépartis (méthode de Newton-Cotes) ou des points de Gauss (qui vérifient certaines propriétés d'optimalité). Il est cependant conseillé de leur superposer des méthodes d'extrapolation de type Richardson/Romberg ou des méthodes automatiques (adaptatives ou non) pour assurer que les quantités calculées le sont avec une précision suffisante.

Pour les problèmes en grande dimension, les méthodes déterministes ne sont plus utilisables, et on leur préfère des méthodes stochastiques (voir Section 2.3). Ces méthodes stochastiques peuvent être des méthodes directes (auquel cas il faut souvent les améliorer avec des techniques de réduction de variance), des méthodes de quasi-Monte Carlo, ou, dans les cas les plus compliqués, des méthodes fondées sur des chaînes de Markov ergodiques.

2.1 Motivation : calcul de propriétés moyennes en physique statistique

Présentons pour commencer une situation physique intéressante dans laquelle il s'agit de calculer des intégrales de fonctions. A l'échelle microscopique, la matière n'est pas continue mais est

composée d'atomes, qui ont des positions et des vitesses données. Un système classique[1] de N atomes est décrit par sa configuration microscopique ou microétat :

$$(q,p) = (q_1,\dots,q_N,\ p_1,\dots,p_N) \in \mathcal{D}^N \times \mathbb{R}^{dN},$$

où les positions q_i sont dans un domaine $\mathcal{D}$ (typiquement, une boite, avec des conditions de bord périodiques) et p_i est l'impulsion de la particule i (vitesse multipliée par la masse m_i). L'énergie du système dans cette configuration est donnée par le Hamiltonien

$$H(q,p) = \sum_{i=1}^{N} \frac{p_i^2}{2m_i} + V(q_1,\dots,q_N).$$

L'expression de l'énergie cinétique est la même pour tous les systèmes : toute la physique est contenue dans l'expression de V. Rappelons également quelques ordres de grandeur. Les distances interatomiques typiques sont de quelques Å (soit 10^{-10} m), les énergies à température ambiante de l'ordre de $k_{\mathrm{B}}T \simeq 4 \times 10^{-21}$ J, et le nombre de molécules dans un échantillon macroscopique de l'ordre de $\mathcal{N}_A = 6,02 \times 10^{23}$ atomes ! On ne peut donc pas considérer tous les degrés de libertés d'un système macroscopique car ils sont bien trop nombreux ; et c'est également inutile car souvent un comportement moyen se dégage. Pour toutes ces raisons, on décrit le système par le biais d'un macroétat ou ensemble thermodynamique. Mathématiquement, c'est une mesure de probabilité sur l'ensemble des configurations accessibles.

Les propriétés moyennes ou grandeurs thermodynamiques d'équilibre sont obtenues en prenant la moyenne de fonctions des microétats par rapport à la mesure de probabilité décrivant l'état du système :

$$\langle A \rangle = \int_{\mathcal{D}^N \times \mathbb{R}^{dN}} A(q,p)\, d\mu(q,p).$$

On voit donc que cela demande de calculer une intégrale en dimension très grande, ladite intégrale ne pouvant se calculer analytiquement que dans des cas très simples (et rarement pertinents pour donner une information quantitative précise, même s'ils permettent parfois de discuter qualitativement des comportements physiques). Pour donner des ordres de grandeur des calculs effectués en pratique, disons qu'on peut simuler informatiquement de nos jours des systèmes allant de quelques centaines à plusieurs milliards d'atomes.

Pour préciser cette discussion, présentons un exemple : le calcul de la loi d'état de l'argon, *i.e.*, la courbe donnant la pression en fonction de la densité et de la température. L'observable associée à la pression est

$$A(q,p) = \frac{1}{d|\mathcal{D}|} \sum_{i=1}^{N} \left(\frac{p_i^2}{m_i} - q_i \cdot \nabla_{q_i} V(q) \right).$$

Par ailleurs, la mesure canonique décrit un système à température constante et densité fixée :

$$\mu_{\mathrm{NVT}}(dq\,dp) = \frac{1}{Z} \exp\left(-\frac{H(q,p)}{k_{\mathrm{B}}T} \right),$$

où la fonction de partition Z est une constante de normalisation qui assure que μ_{NVT} est bien une mesure de probabilité. Il ne reste plus qu'à préciser qui est le potentiel d'interaction V. C'est là qu'intervient une erreur de modélisation puisqu'on remplace souvent $V(q)$, qui en toute rigueur devrait être calculé par le biais de la physique quantique comme l'énergie fondamentale de l'opérateur de Schrödinger associé aux atomes placés aux positions q_i, par une formule empirique. Pour l'argon dans les conditions thermodynamiques usuelles, une bonne approximation consiste à utiliser des interactions de paires de type Lennard–Jones :

$$V(q_1,\dots,q_N) = \sum_{1 \leqslant i < j \leqslant N} V_0(|q_j - q_i|),$$

1. Au sens de non quantique.

avec

$$V_0(r) = 4\varepsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right],$$

et $\sigma = 3,405 \times 10^{-10}$ m, $\varepsilon/k_{\mathrm{B}} = 119,8$ K. En fait, seule la partie d'interactions à longue portée en r^{-6} a un sens physique (interactions de Van der Waals), l'objectif de la partie à courte portée étant d'empêcher que deux atomes ne puissent trop se rapprocher (ce qui modélise la répulsion des nuages électroniques associés aux atomes).

2.2 Méthodes déterministes

Dans toute cette section, on se limite au cas d'intégrales à calculer en dimension 1, sur des domaines bornés $[a, b]$, et pour des fonctions régulières f. On cherche donc à approcher

$$I(f) = \int_a^b f(x)\, dx. \tag{2.1}$$

Il y a bien sûr plein d'extensions possibles pour traiter les cas plus compliqués qui ne rentrent pas dans le jeu de nos hypothèses (fonctions f à intégrer non bornées ou possédant des singularités, domaines d'intégration non bornés, etc), mais nous nous limitons volontairement au cas le plus simple pour faire ressortir le cœur des méthodes. Notons également que, quitte à faire un changement de variable affine et à remplacer la fonction à intégrer par $g(t) = f(a + (b - a)t)$, on peut se ramener à l'intégration de fonctions sur l'intervalle $[0, 1]$.

2.2.1 Principe de base des méthodes déterministes

Toute méthode numérique déterministe d'approximation d'une intégrale est fondée sur une formule de quadrature utilisant des nœuds $x_i \in [a, b]$ et des poids $\omega_i \in \mathbb{R}$:

$$\int_a^b f(x)\, dx \simeq \sum_{i=0}^{n} \omega_i f(x_i). \tag{2.2}$$

Définition 2.1 (Ordre d'une méthode de quadrature). *On dit qu'une méthode est d'ordre k si l'égalité* (2.2) *a lieu pour tout polynôme de degré au plus k.*

On définit également l'erreur de quadrature

$$E(f) = \int_a^b f(x)\, dx - \left(\sum_{i=0}^{n} \omega_i f(x_i) \right).$$

L'idée de base qui sous-tend le choix des poids et des nœuds dans (2.2) est de remplacer la fonction à intégrer par une fonction plus simple qui coïncide avec cette fonction en un certain nombre de points (fonction interpolante), les poids étant ensuite déterminés par l'intégration analytique de la fonction interpolante. Des exemples simples sont présentés ci-dessous.

Si la fonction f ou ses dérivées varient significativement entre a et b, il peut être opportun de remplacer une formule de quadrature globale telle que (2.2) par une formule dite composite : on découpe l'intégrale sur $[a, b]$ en M intégrales élémentaires

$$\int_a^b f(x)\, dx = \sum_{m=1}^{M} \int_{\alpha_{m-1}}^{\alpha_m} f(x)\, dx$$

avec $a = \alpha_0 < \alpha_1 < \cdots < \alpha_M = b$. Chaque intégrale élémentaire apparaissant dans la somme du membre de droite est ensuite approchée par une formule du type (2.2) (obtenue en pratique par un changement de variable affine de (2.2) donnée sur $[0, 1]$). Là aussi, des exemples simples sont présentés ci-dessous.

Quelques exemples simples

Commençons par présenter quelques exemples classiques qui permettront de fixer les idées, et seront également l'occasion de faire un peu d'analyse d'erreur *a priori*. Rappelons toutefois avant de commencer la formule de Taylor avec reste exact, que nous utiliserons de manière répétée : pour une fonction régulière $f : \mathbb{R} \to \mathbb{R}$,

$$f(x) = f(0) + xf'(0) + \frac{x^2}{2}f''(0) + \cdots + \frac{x^n}{n!}f^{(n)}(0) + \frac{x^{n+1}}{(n+1)!}f^{(n+1)}(\theta_x),$$

avec $\theta_x \in [0, x]$.

Exemple 2.1 (Méthode du point milieu). La fonction à intégrer est approchée par une constante, la valeur de cette constante étant la valeur de la fonction au milieu de l'intervalle. L'intégration analytique de la fonction interpolante est triviale, et donne

$$I_0(f) = (b-a)f\left(\frac{a+b}{2}\right).$$

La version composite de cette méthode est la suivante. On pose $h = (b-a)/M$ et on considère les nœuds $x_m = a + (m+1/2)h$ avec $m = 0, \ldots, M-1$. Dans ce cas, l'intégrale de la fonction est approchée par la somme discrète

$$I_{0,M}(f) = h\sum_{m=0}^{M-1} f(x_m).$$

Essayons à présent d'obtenir une estimation de l'erreur commise en remplaçant l'intégrale par $I_0(f)$ ou $I_{0,M}(f)$. On va supposer que f est assez régulière, $C^2([a,b])$ dans le cas présent. L'outil technique de base est d'utiliser une formule de Taylor avec reste exact : pour tout $x \in [a,b]$, il existe $\theta(x) \in [a,b]$ tel que

$$f(x) = f\left(\frac{a+b}{2}\right) + f'\left(\frac{a+b}{2}\right)\left(x - \frac{a+b}{2}\right) + \frac{1}{2}f''\left(\theta(x)\right)\left(x - \frac{a+b}{2}\right)^2.$$

En intégrant cette égalité sur $[a,b]$, on obtient

$$\int_a^b f(x)\,dx = (b-a)f\left(\frac{a+b}{2}\right) + \frac{1}{2}\int_a^b f''\left(\theta(x)\right)\left(x - \frac{a+b}{2}\right)^2 dx.$$

Le second terme du membre de droite peut se réécrire comme

$$\begin{aligned}\int_a^b f''\left(\theta(x)\right)\left(x - \frac{a+b}{2}\right)^2 dx &= (b-a)^3\int_0^1 f''\Big(\theta(a+(b-a)t)\Big)\left(t - \frac{1}{2}\right)^2 dt \\ &= (b-a)^3 f''(\xi)\int_0^1 \left(t - \frac{1}{2}\right)^2 dt,\end{aligned}$$

où on a utilisé le théorème de la valeur moyenne pour l'intégration pour obtenir la dernière égalité : avec $F_2(t) = f''(\theta[a+(b-a)t])$, il existe t_ξ tel que

$$\int_0^1 F_2(t)\left(t - \frac{1}{2}\right)^2 dt = F_2(t_\xi)\int_0^1 \left(t - \frac{1}{2}\right)^2 dt.$$

Au final, en posant $H = (b-a)/2$, on peut donc dire qu'il existe $\xi \in [a,b]$ tel que l'erreur de quadrature s'écrive

$$E(f) = \int_a^b f(x)\,dx - I_0(f) = \frac{H^3}{3}f''(\xi).$$

On a donc une méthode d'ordre 1. Pour la méthode composite correspondante, on montre que l'erreur de quadrature est

$$E_M(f) = \frac{(b-a)h^2}{24} f''(\eta) \tag{2.3}$$

pour un certain $\eta \in [a,b]$ (voir Exercice 2.1). Notons que l'erreur varie en $1/M^2$, où M est le nombre de nœuds de la quadrature.

Exercice 2.1 (Estimations d'erreur pour les formules composites). *On suppose que f est aussi régulière que nécessaire.*

(1) Montrer qu'il existe une constante $C > 0$ telle que $|I_{0,M}(f) - I(f)| \leqslant C\, h^2$.

(2) On souhaite à présent montrer que la borne supérieure obtenue à la question précédente ne peut être améliorée. Pour ce faire, on utilise le théorème de la moyenne discrète : soit $u \in C^0([a,b])$, $\{x_j\}_{j=0,\dots,s}$ des points de $[a,b]$ et $\{\delta_j\}_{j=0,\dots,s}$ des nombres de signe constant. Alors il existe $\eta \in [a,b]$ tel que

$$\sum_{j=0}^{s} \delta_j\, u(x_j) = u(\eta) \sum_{j=0}^{s} \delta_j.$$

Prouver ce résultat.

(3) En déduire qu'il existe $\eta \in [a,b]$ tel que $I(f) - I_{0,M}(f) = \dfrac{b-a}{24} f''(\eta)\, h^2$.

Exemple 2.2 (Méthode des trapèzes). On interpole la fonction à intégrer par une fonction affine prenant les mêmes valeurs aux extrémités de l'intervalle. Une intégration analytique de la fonction interpolante donne alors

$$I_1(f) = \frac{b-a}{2}\Big(f(a) + f(b)\Big).$$

La formule composite correspondante utilise les nœuds $x_m = a + mh$ pour $m = 0, \dots, M$ et $h = (b-a)/M$:

$$I_{1,M} = h\left(\frac{1}{2}f(x_0) + f(x_1) + \cdots + f(x_{M-1}) + \frac{1}{2}f(x_M)\right). \tag{2.4}$$

On montre (voir Exercice 2.2) que les erreurs de quadrature sont respectivement

$$E_1(f) = \int_a^b f(x)\,dx - I_1(f) = -\frac{(b-a)^3}{12} f''(\xi), \qquad E_{1,M}(f) = -\frac{(b-a)h^2}{12} f''(\eta), \tag{2.5}$$

pour des nombres $\xi, \eta \in [a,b]$. On obtient donc une méthode numérique d'ordre 1 aux performances très similaires à celle de la méthode du point milieu.

Exercice 2.2 (Estimation d'erreur pour la formule des trapèzes). *Montrer qu'il existe $\xi \in [a,b]$ tel que*

$$E_1(f) = \int_a^b f(x)\,dx - I_1(f) = -\frac{(b-a)^3}{12} f''(\xi),$$

puis qu'il existe $\eta \in [a,b]$ tel que

$$E_{1,M}(f) = \int_a^b f(x)\,dx - I_{1,M}(f) = -\frac{(b-a)}{12} f''(\eta) h^2,$$

Indications : on posera

$$g(x) = f(x) - \left\{\frac{f(a)+f(b)}{2} + \left(x - \frac{a+b}{2}\right)\frac{f(b)-f(a)}{b-a}\right\},$$

et on évaluera $\int_a^b g(x)\,dx$ *(utiliser une intégration par parties) ; on utilisera également la formule de la moyenne intégrale : si* $u \in C^0([a,b])$*, alors*

$$\int_a^b u(x)\,dx = (b-a)u(\xi)$$

pour un certain $\xi \in [a,b]$*.*

Exemple 2.3 (Méthode de Cavalieri–Simpson). C'est une méthode d'ordre 3, qui est à la base de nombre de techniques plus raffinées. Son principe de base est d'approcher la fonction à intégrer par une parabole en utilisant les trois points d'interpolation $a, (a+b)/2, b$. L'intégration analytique de la fonction interpolante donne (voir aussi Exercice 2.3)

$$I_2(f) = \frac{b-a}{6}\Big(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\Big).$$

On peut montrer que l'erreur de quadrature est

$$E_2(f) = \int_a^b f(x)\,dx - I_2(f) = -\frac{(b-a)^5}{90} f^{(4)}(\xi).$$

La formule composite correspondante utilise $2M+1$ nœuds $x_m = a + mh/2$ pour $m = 0, \ldots, 2M$ et $h = (b-a)/M$:

$$I_{2,M}(f) = \frac{h}{6}\left(f(x_0) + 2\sum_{r=1}^{M-1} f(x_{2r}) + 4\sum_{s=0}^{M-1} f(x_{2s+1}) + f(x_{2M})\right). \tag{2.6}$$

L'erreur de quadrature associée

$$E_{2,M}(f) = -\frac{(b-a)h^4}{2880} f^{(4)}(\eta), \qquad \eta \in [a,b], \tag{2.7}$$

est ici proportionnelle à $1/M^4$. On a intérêt à utiliser cette quadrature par rapport à la méthode du point milieu ou à la méthode des trapèzes si la fonction à intégrer est plus régulière (dérivée quatrième pas trop grande).

Points équidistants

Nous décrivons rapidement ici le principe des méthodes dites de Newton–Cotes, fondées sur l'interpolation de Lagrange en des points équidistants. On note x_j $(0 \leqslant j \leqslant n)$ les nœuds de quadrature : $x_j = a + jh$ avec $h = (b-a)/n$, et P_j les polynômes d'interpolation valant 1 en x_j et 0 aux autres nœuds. La fonction interpolante approchant f est alors

$$f_n(x) = \sum_{j=0}^{n} f(x_j) P_j(x), \qquad P_j(x) = \prod_{k \neq j} \frac{x - x_k}{x_j - x_k},$$

et on obtient alors une approximation de l'intégrale à calculer en évaluant analytiquement

$$\int_a^b f_n(x)\,dx.$$

On construit ainsi des méthodes d'ordre arbitrairement élevé. Prévenons toutefois le lecteur que la montée en ordre peut toutefois être limitée en pratique par des instabilités numériques (liées à l'apparition de poids négatifs pour les formules d'ordres élevés), et ne fait sens que si la fonction est assez régulière, avec des dérivées pas trop grandes.

Exercice 2.3. *Calculer les coefficients* α, β, γ *tels que la méthode d'intégration numérique*

$$I_2(f) = (b-a)\left(\alpha f(a) + \beta f\left(\frac{a+b}{2}\right) + \gamma f(b)\right)$$

soit une méthode d'ordre 3, et montrer que l'on retrouve la méthode de Cavalieri–Simpson. On remarquera pour commencer que l'on peut se ramener, par un changement de variable idoine, au cas où $a = -1$ *et* $b = 1$*.*

Points de Gauss

Les nœuds dans la méthode de Gauss sont les racines de polynômes orthogonaux d'ordre croissant. Cette classe de quadrature est importante car elle satisfait une certaine propriété d'optimalité : on peut montrer qu'elle est d'ordre $2n-1$ pour n nœuds. On peut obtenir des formules d'ordre arbitrairement élevé, qui sont numériquement stables (au sens où les poids qui interviennent dans la quadrature sont toujours positifs). En revanche, l'expression analytique des nœuds et des poids est de plus en plus compliquée et coûteuse à évaluer. Un exemple important est la méthode de Gauss d'ordre 3, obtenue par une quadrature à 2 nœuds (voir Exercice 2.4) :

$$I(f) = \int_{-1}^{1} f(x)\,dx \simeq I_{\mathrm{G}}(f) = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right).$$

On peut bien sûr transformer cette quadrature élémentaire en une quadrature sur un intervalle général $[a,b]$ par une transformation affine :

$$I(f) = \int_{a}^{b} f(x)\,dx \simeq I_{\mathrm{G}}(f) = \frac{b-a}{2}\left[f\left(\frac{a+b}{2} - \frac{1}{\sqrt{3}}\frac{b-a}{2}\right) + f\left(\frac{a+b}{2} + \frac{1}{\sqrt{3}}\frac{b-a}{2}\right)\right].$$

Exercice 2.4. *Trouver les valeurs de α, β et $a \in [0,1]$ telle que la méthode suivante soit d'ordre 3 :*

$$\int_{-1}^{1} f(x)\,dx \simeq I(f) = 2\Big(\alpha f(-a) + \beta f(a)\Big).$$

Donner la formule de quadrature dans le cas d'un intervalle $[a,b]$ général.

2.2.2 Extrapolation

Pour gagner en précision, plutôt que de travailler avec des méthodes d'ordre élevé, qui demandent beaucoup de nœuds et sont relativement lourdes à mettre en œuvre, une idée attractive est de partir d'une méthode plus simple et d'améliorer ses résultats de manière systématique par un procédé itératif. L'idée de base sous-tendant cette approche est l'extrapolation de Richardson. Son application à la quadrature numérique donne la méthode de Romberg.

Extrapolation de Richardson

Supposons que l'on sache que la fonction A admette le développement suivant pour t petit :

$$A(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \cdots + \alpha_k t^k + \mathrm{O}(t^{k+1}) \tag{2.8}$$

et qu'on souhaite calculer α_0. L'idée est d'extrapoler la valeur de la fonction A en 0 connaissant ses valeurs pour des arguments $t > 0$. On voit déjà qu'on peut combiner $A(t)$ et $A(\delta t)$ (avec $0 < \delta < 1$) pour éliminer le terme linéaire et avoir une approximation à l'ordre 2 de α_0 :

$$\frac{A(\delta t) - \delta A(t)}{1-\delta} = \alpha_0 - \alpha_2 \delta t^2 + \dots$$

On peut ensuite combiner deux approximations d'ordre 2 pour en obtenir une à l'ordre 3, etc. Cela revient à définir successivement les fonctions

$$\begin{aligned} A_0(t) &= A(t), \\ A_1(t) &= \frac{A_0(\delta t) - \delta A_0(t)}{1-\delta}, \\ A_2(t) &= \frac{A_1(\delta t) - \delta^2 A_1(t)}{1-\delta^2}, \end{aligned}$$

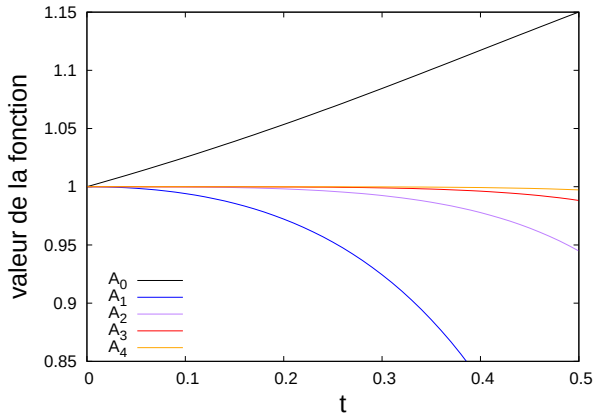

Fig. 2.1. Illustration de l'extrapolation de Richardson pour $t = 0.5$ et $\delta = 0.5$, partant d'une fonction A qui est un polynôme de degré 10 avec des coefficients aléatoirement choisis.

ce qui donne $A_2(t) = \alpha_0 + \alpha_3\delta^3(1+\delta)t^3 + \dots$ et, de manière générale,

$$A_n(t) = \frac{A_{n-1}(\delta t) - \delta^n A_{n-1}(t)}{1-\delta^n}.$$

On élimine ainsi successivement les termes en t, t^2,..., t^n. Cette procédure est illustrée sur la Figure 2.1, où l'on voit que les fonctions successives $A_1, A_2, \dots$ sont de plus en plus "plates".

On peut réécrire cette procédure de façon à faire apparaitre plus explicitement les quantités que l'on doit effectivement calculer, à savoir les valeurs de la fonction de base $A_0 = A$. Pour ce faire, on définit tout d'abord, pour une valeur t fixée,

$$\mathcal{A}_{m,0} = A(\delta^m t), \qquad m = 0, \dots, n,$$

puis, pour des valeurs croissantes de q, on combine ces valeurs selon

$$\mathcal{A}_{m,q+1} = \frac{\mathcal{A}_{m,q} - \delta^{q+1}\mathcal{A}_{m-1,q}}{1-\delta^{q+1}} = \frac{\delta^{-(q+1)}\mathcal{A}_{m,q} - \mathcal{A}_{m-1,q}}{\delta^{-(q+1)} - 1}, \qquad q = 0, \dots, n-1.$$

Une illustration graphique permet de mieux comprendre ce qui se passe. On calcule les termes de la première colonne, et on en déduit les termes diagonaux, qui sont ceux qui nous intéressent :

$$\begin{array}{lccccccc}
\mathcal{A}_{0,0} = A(t) & & & & & & & \\
 & \searrow & & & & & & \\
\mathcal{A}_{1,0} = A(\delta t) & \to & \mathcal{A}_{1,1} & & & & & \\
 & \searrow & & \searrow & & & & \\
\mathcal{A}_{2,0} = A(\delta^2 t) & \to & \mathcal{A}_{2,1} & \to & \mathcal{A}_{2,2} & & & \\
 & \searrow & & \searrow & & \searrow & & \\
\mathcal{A}_{3,0} = A(\delta^3 t) & \to & \mathcal{A}_{3,1} & \to & \mathcal{A}_{3,2} & \to & \mathcal{A}_{3,3} & \\
\vdots & \ddots & & \ddots & & \ddots & & \ddots \\
 & \searrow & & \searrow & & \searrow & & \searrow \\
\mathcal{A}_{n,0} = A(\delta^n t) & \to & \mathcal{A}_{n,1} & \to & \mathcal{A}_{n,2} & \to & \mathcal{A}_{n,3} & \dots \to \mathcal{A}_{n,n}
\end{array}$$

Une récurrence simple (voir Exercice 2.5) montre que

$$\mathcal{A}_{m,q} = A_q(\delta^{m-q}t). \tag{2.9}$$

Dans la représentation graphique ci-dessus, l'indice m correspond aux lignes, et l'indice q aux colonnes.

Considérons à présent la vitesse de convergence de cette méthode. Comme $A(t) = \alpha_0 + \mathrm{O}(t)$, on a seulement $\mathcal{A}_{m,0} = \alpha_0 + \mathrm{O}(\delta^m t)$. En revanche, on peut vérifier (voir Exercice 2.5) que

$$\mathcal{A}_{m,q} = \alpha_0 + \mathrm{O}\left(\delta^{(q+1)(m-q/2)} t^{q+1}\right). \tag{2.10}$$

En particulier, $\mathcal{A}_{n,n} = \alpha_0 + \mathrm{O}\left(\delta^{n(n+1)/2} t^{n+1}\right)$. La convergence de $\mathcal{A}_{n,n}$ est donc $(n+1)/2$ fois plus rapide que celle de $\mathcal{A}_{n,0} = \alpha_0 + \mathrm{O}(\delta^n t)$ (lorsque l'on ne considère que la vitesse de convergence par rapport à δ et que l'on ne s'intéresse pas au facteur t).

Exercice 2.5 (Extrapolation de Richardson). *L'objectif de cet exercice est de vérifier la formule* (2.10).

(1) Vérifier la formule (2.9) *en procédant par récurrence.*

(2) Pour montrer (2.10)*, on va tout d'abord obtenir une expression de* $A_n(t) = \alpha_0 + \alpha_{n+1}^{(n)} t^{n+1} + \alpha_{n+2}^{(n)} t^{n+2} + \dots$.

(a) Montrer que, pour $m \geqslant n+1$*, on a la relation de récurrence*

$$\alpha_m^{(n)} = \frac{\delta^{m-n} - 1}{1 - \delta^n} \delta^n \alpha_m^{(n-1)}.$$

(b) En déduire que

$$\left|\alpha_m^{(n)}\right| \leqslant |\alpha_m| \prod_{k=1}^{n} \frac{\delta^k}{1-\delta^k}.$$

(c) Montrer qu'il existe une constante $K_\delta > 1$ *telle que* $\displaystyle\prod_{k=1}^{n}(1-\delta^k) \geqslant \exp\left(-\frac{K_\delta \delta}{1-\delta}\right)$ *uniformément en* $n \geqslant 1$*. Indication : on utilisera le fait que, par concavité de* $x \mapsto \ln(1-x)$*, il existe* $K_\delta > 1$ *tel que, pour tout* $x \in [0, \delta]$*, on ait* $\ln(1-x) \geqslant -K_\delta x$.

(d) Montrer qu'il existe une constante $C_\delta > 0$ *telle que* $\left|\alpha_m^{(n)}\right| \leqslant C_\delta \, |\alpha_m| \, \delta^{n(n+1)/2}$.

(e) Conclure.

Intégration de Romberg

Appliquons à présent l'extrapolation de Richardson aux formules de quadrature dans le cas particulier de la formule composite des trapèzes : c'est l'intégration de Romberg. On doit d'abord établir un développement du type (2.8). Par le développement d'Euler–Maclaurin, on montre que la formule des trapèzes composite (2.4) peut s'écrire, avec $h = (b-a)/M$,

$$\begin{aligned} I_{1,M} &= h\left(\frac{1}{2}f(x_0) + f(x_1) + \dots + f(x_{M-1}) + \frac{1}{2}f(x_M)\right) \\ &= \int_a^b f(x)\,dx + \sum_{i=1}^{k} \frac{B_{2i}}{(2i)!}\Big(f^{(2i-1)}(b) - f^{(2i-1)}(a)\Big)h^{2i} + \mathrm{O}(h^{2(k+1)}), \end{aligned}$$

où les B_n sont les nombres de Bernoulli. Insistons sur le fait que les coefficients qui apparaissent dans le développement ci-dessus ne sont pas importants par eux-mêmes, c'est la forme analytique du développement qui nous importe. Ainsi,

$$I_{1,M} = T(h) = \alpha_0 + \alpha_1 h^2 + \dots$$

est une série en puissances de h^2. On retrouve donc une expression similaire à (2.8) en remplaçant t par h^2. Lorsque l'on divise le pas du maillage par 2, on doit donc utiliser un procédé d'extrapolation avec $\delta = 1/4$ dans les notations de la section précédente. L'algorithme est alors le suivant : on

commence par évaluer le résultat donné par des formules composites avec un pas divisé par 2 à chaque étape, *i.e.*,

$$\mathcal{A}_{m,0} = T\left(\frac{h}{2^m}\right), \qquad m = 0, \ldots, n.$$

On combine ensuite ces approximations selon

$$\mathcal{A}_{m,q+1} = \frac{4^{q+1}\mathcal{A}_{m,q} - \mathcal{A}_{m-1,q}}{4^{q+1} - 1}, \qquad q = 0, \ldots, n-1.$$

Le terme $\mathcal{A}_{m,q}$ approche l'intégrale (2.1) avec un erreur d'ordre $\mathrm{O}\left(2^{q(q+1)}\left(\frac{h}{2^m}\right)^{2(q+1)}\right)$ (remplacer t par h^2 et prendre $\delta = 1/4$ dans (2.9)). En particulier, $\mathcal{A}_{n,n}$ approche l'intégrale avec une erreur d'ordre $\mathrm{O}\left(\left(\frac{h}{2^n}\right)^{n+1}\right)$.

Interprétons enfin cette convergence dans le cas où $h = b - a = 1$: lorsque l'on ne fait pas d'extrapolation, et que l'on calcule $\mathcal{A}_{n,0}$, on utilise un pas d'espace

$$h_n = \frac{1}{2^n},$$

et on a ainsi une erreur de quadrature en h_n^2 au vu de l'erreur sur la formule des trapèzes composite (voir (2.5)). Lorsque l'on fait l'extrapolation, l'erreur est énormément réduite, et est plus précisément d'ordre h_n^{n+1}.

2.2.3 Méthodes automatiques

Une autre manière d'améliorer efficacement des méthodes d'ordre bas est d'utiliser des méthodes automatiques d'intégration assurant que l'erreur de quadrature est plus petite qu'un seuil donné. Ces techniques sont fondées sur des estimateurs d'erreur *a posteriori*. On peut même utiliser si on le souhaite une méthode adaptative, qui détermine de manière automatique comment répartir les nœuds, plutôt que de raffiner uniformément la grille.

Méthode automatique non-adaptative

Commençons par présenter une méthode qui détermine automatiquement le nombre de points M à utiliser dans une formule composite, en procédant par un raffinement uniforme de la grille (avec des points régulièrement espacés). On considère par exemple la formule composite de Cavalieri–Simpson (2.6), dont on rappelle l'estimation d'erreur *a priori* (2.7) :

$$E_M = I - I_M = -\frac{b-a}{2880}\left(\frac{b-a}{M}\right)^4 f^{(4)}(\eta_M).$$

La notation η_M insiste sur le fait que le point η_M pour lequel l'égalité ci-dessus est valide dépend *a priori* de M. Si on suppose toutefois que $f^{(4)}(\eta_M) \simeq f^{(4)}(\eta_{2M})$ (ce qui est effectivement le cas lorsque $M \to +\infty$), on a alors $E_{2M} \simeq E_M/16$. Un calcul simple montre que

$$E_{2M} \simeq \frac{I_{2M} - I_M}{15}. \tag{2.11}$$

Insistons sur le fait que cette estimation *a posteriori* de l'erreur est obtenue en combinant une estimation *a priori* et deux évaluations de I pour des paramètres différents. En pratique, on va donc commencer par une valeur de M_0 donnée, puis doubler cette valeur jusqu'à ce que l'erreur de quadrature estimée par (2.11) soit plus petite que la tolérance que l'on s'est initialement fixée. Au vu du caractère approché de l'estimation *a posteriori* (2.11), il est plus prudent de considérer par exemple un critère d'arrêt tel que $E_{2M} \simeq (I_{2M} - I_M)/10$.

Méthodes adaptatives

Nous concluons ici notre rapide panorama des méthodes déterministes d'intégration par une méthode de quadrature adaptative, dont des idées sont utilisées dans la définition des algorithmes standards d'intégration des logiciels comme Matlab. Etant donnée une tolérance ε préalablement fixée, l'objectif est d'obtenir une formule de quadrature avec une distribution non-uniforme de nœuds (aussi près les uns des autres qu'il le faut dans certaines régions, mais toujours en prenant soin de limiter au maximum le nombre de ces points), sans fixer le nombre de points au préalable.

Pour ce faire, on souhaite que l'erreur de quadrature estimée sur tout sous-intervalle $[\alpha, \beta]$ soit en $\varepsilon(\beta - \alpha)/(b - a)$. On part initialement de $\alpha = a$ et $\beta = b$. Pour α, β fixés, on calcule l'intégrale sur l'intervalle $[\alpha, \beta]$ avec une formule de Cavalieri–Simpson simple

$$S_f(\alpha, \beta) = \frac{\beta - \alpha}{6} \left(f(\alpha) + 4f\left(\frac{\alpha + \beta}{2}\right) + f(\beta) \right),$$

et on compare le résultat à ce que donne une formule de Cavalieri–Simpson composite avec $M = 2$ sur le même intervalle :

$$S_{f,2}(\alpha, \beta) = S_f\left(\alpha, \frac{\alpha + \beta}{2}\right) + S_f\left(\frac{\alpha + \beta}{2}, \beta\right).$$

On montre comme pour (2.11) que

$$\left| \int_\alpha^\beta f - S_{f,2}(\alpha, \beta) \right| \simeq \frac{1}{15} \left| \mathcal{E}_f(\alpha, \beta) \right|, \qquad \mathcal{E}_f(\alpha, \beta) = S_f(\alpha, \beta) - S_{f,2}(\alpha, \beta).$$

On va donc utiliser le critère d'arrêt suivant afin d'avoir une erreur de quadrature d'ordre $\varepsilon(\beta - \alpha)/(b - a)$ sur l'intervalle $[\alpha, \beta]$ (le facteur 10 ci-dessous remplaçant le 15 de la ligne précédente, pour plus de prudence) :

$$\left| \mathcal{E}_f(\alpha, \beta) \right| \leqslant 10\, \varepsilon\, \frac{\beta - \alpha}{b - a}. \tag{2.12}$$

On distingue ensuite deux cas :

- Si (2.12) est vrai, on réalise effectivement la quadrature de Cavalieri–Simpson avec les nœuds $\alpha, (\alpha+\beta)/2, \beta$, et on ajoute la contribution correspondante $S_{f,2}(\alpha, \beta)$ à l'estimation courante de l'intégrale sur l'intervalle $[a, \alpha]$. On obtient ce faisant une estimate courante de l'intégrale sur l'intervalle $[a, \beta]$. On continue ensuite en considérant l'intervalle $[\beta, b]$ sur lequel il reste à estimer l'intégrale de la fonction.
- Sinon, on raffine l'intégration sur l'intervalle $[\alpha, \beta]$ en gardant la borne gauche α et en remplaçant la borne droite par $(\alpha + \beta)/2$, et on reprend la procédure d'estimation avec le critère d'arrêt (2.12).

Il est sain dans toute cette procédure de vérifier que la taille des intervalles sur lesquels on réalise la quadrature ne dégénère pas, *i.e.*, $\beta - \alpha \geqslant h_{\min} > 0$, où $h_{\min}$ est une longueur minimale fixée par le numéricien. Un message d'erreur avertissant l'utilisateur qu'un des intervalles est trop petit est en général un signe que la fonction à intégrer a des singularités.

2.3 Méthodes stochastiques

Les méthodes déterministes sont très efficaces et tout à fait appropriées pour approcher numériquement des intégrales en dimension petite, mais ne sont pas très adaptées au calcul d'intégrales en dimension grande en général. L'argument de base est le suivant : une quadrature pour un domaine en dimension d est typiquement obtenue en considèrant le produit tensoriel de quadratures unidimensionnelles. Ainsi, pour un pas spatial h en dimension 1, on a $\mathrm{O}(h^{-d})$ noeuds de quadrature en dimension d, ce qui représente un nombre exorbitant de points où évaluer la fonction à intégrer. Par ailleurs, les points générés par des tensorisations de quadratures en dimension plus

petite sont redondants. Lorsque la dimension de l'espace est modérément grande et que la fonction à intégrer est suffisamment régulière, on peut utiliser des techniques de type *sparse grids*. Les méthodes stochastiques restent toutefois les méthodes de choix pour les problèmes en très grande dimension, comme ceux rencontrés en physique statistique numérique. On se concentrera pour simplifier dans cette section sur l'approximation d'intégrales de la forme

$$I(f) = \int_{[a,b]^d} f(x)\,dx$$

pour des fonctions f régulières. L'extension à des domaines plus généraux du type $[a_1, b_1] \times \cdots \times [a_d, b_d]$ ne pose pas de problème de principe mais alourdit les notations. En fait, à un changement de variable affine près, on peut toujours ramener une intégration sur un domaine borné quelconque à une intégration sur $[0,1]^d$ (quitte à prolonger la fonction qu'on intègre par 0).

2.3.1 Principe de la méthode

L'idée des méthodes d'intégration stochastiques est de considérer des noeuds de quadrature aléatoirement positionnés, avec des poids fixes. La consistance de ces méthodes repose sur la loi forte des grands nombres, alors que leur vitesse de convergence peut être obtenue par un théorème de la limite centrale.

Plus précisément, considérons des noeuds de quadrature x_i indépendants et identiquement distribués selon une loi uniforme sur $[a,b]^d$, *i.e.* avec une densité de probabilité

$$\mu(dx) = \frac{1}{(b-a)^d} \mathbf{1}_{[a,b]^d}(x)\,dx.$$

Définissons l'estimateur

$$I_N(f) = \frac{(b-a)^d}{N} \sum_{i=1}^{N} f(x_i). \tag{2.13}$$

Notons que cette estimation de l'intégrale est une variable aléatoire. Par la loi forte des grands nombres, on a la convergence presque sûre (lorsque $N \to +\infty$)

$$\frac{1}{N} \sum_{i=1}^{N} f(x_i) \longrightarrow \int_{\mathbb{R}^d} f\,d\mu = \int_{[a,b]^d} f(x)\,\frac{dx}{(b-a)^d},$$

et donc $I_N(f) \to I(f)$ presque sûrement, ce qui donne la consistance de la méthode numérique.

Pour obtenir une estimation de l'erreur dans le cadre d'une analyse *a priori* directe, on utilise le théorème de la limite centrale : lorsque $N \to +\infty$, on a convergence en loi vers une variable aléatoire gaussienne

$$\sqrt{N}\Big(I_N(f) - I(f)\Big) \longrightarrow \mathcal{N}\Big(0, \sigma_I(f)^2\Big),$$

où la variance de la loi Gaussienne limite est donnée par

$$\sigma_I(f)^2 = (b-a)^d \int_{[a,b]^d} f^2 - \left(\int_{[a,b]^d} f\right)^2 = (b-a)^{2d} \left[\fint_{[a,b]^d} f^2 - \left(\fint_{[a,b]^d} f\right)^2\right], \tag{2.14}$$

où

$$\fint_{\mathcal{D}} f = \frac{1}{|\mathcal{D}|} \int_{\mathcal{D}} f$$

est la valeur moyenne de f sur le domaine $\mathcal{D}$. Ceci signifie que

$$I_N(f) - I(f) \simeq \frac{\sigma_I(f)}{\sqrt{N}}\,\mathcal{N}(0,1), \tag{2.15}$$

et on a donc une erreur de quadrature qui décroit comme $1/\sqrt{N}$. En particulier, la dimension n'apparait pas explicitement dans le taux de convergence en fonction du nombre de noeuds de la quadrature, qui reste du même ordre quelque soit la dimension de l'espace. Cela dit, la dimension limite de manière implicite la vitesse de convergence par le biais de la variance $\sigma_I(f)^2$ définie en (2.14) : on s'attend à ce que cette variance augmente avec la dimension.

En pratique, la variance est bien sûr inconnue puisqu'elle est donnée par une intégrale du même type que celle que l'on cherche à évaluer. On peut toutefois l'estimer empiriquement en effectuant M réalisations indépendantes de I_N, notées $I_N^1, \dots, I_N^M$, en calculant préalablement la moyenne empirique

$$\overline{I_N} = \frac{1}{M} \sum_{m=1}^{M} I_N^m,$$

puis en évaluant par exemple

$$\Sigma_{M,N} = \frac{N}{M-1} \sum_{m=1}^{M} \left(I_N^m - \overline{I_N} \right)^2.$$

Cet estimateur de la variance $\sigma_I(f)^2$ est motivé par (2.15), qui dit que

$$I_N^m = I(f) + \frac{\sigma_I(f)}{\sqrt{N}} G_N^m,$$

les variables aléatoires G_N^m étant indépendantes, et asymptotiquement distribuées selon une loi Gaussienne standard.

Un aparté sur la génération de variables uniformes pseudo-aléatoires

La génération de variables aléatoires est à la base de tout calcul stochastique. Il faut tout d'abord savoir générer des variables aléatoires uniformes. On peut ensuite générer les autres distributions standards par (i) des changements de variables telle que la méthode de Box-Müller pour les Gaussiennes :

$$G = \sqrt{-2 \ln U_1} \cos(2\pi U_2)$$

où U_1, U_2 sont des variables uniformes sur $[0,1]$ indépendantes ; ou (ii) inversion de la fonction de répartition

$$F(x) = \int_{-\infty}^{x} f(y)\, dy.$$

En effet,

$$X = F^{-1}(U) \sim f(x)\, dx.$$

Cette méthode est utilisée pour échantillonner des lois exponentielles par exemple.

Les variables aléatoires uniformes générées informatiquement sont en fait pseudo-aléatoires puisqu'elles sont obtenues par une méthode déterministe ! On peut utiliser par exemple des générateurs linéaires congruentiels, qui produisent des suites "chaotiques" obtenues par une récursion affine du type (les variables x sont des entiers, les variables u sont des réels de l'intervalle $[0,1]$)

$$x_{n+1} = a x_n + b \mod c, \qquad u_n = \frac{x_n}{c-1}.$$

La suite (u_n) a en effet l'air aléatoire, et remplit uniformément l'intervalle $[0,1]$ pour des choix convenables de a, b, c. Cependant, à y regarder de plus près, les générateurs linéaires congruentiels ont des défauts importants (périodes courtes, alignement de points, etc).

D'autres classes de générateurs ont été proposées pour remédier à ces défauts. Citons par exemple l'algorithme de Mersenne-Twister, qui est une méthode par défaut de nombreux logiciels sérieux.

2.3.2 Réduction de variance

Le résultat de convergence (2.14) montre qu'il y a deux options pour réduire l'erreur de quadrature : augmenter le nombre de noeuds N ou essayer de faire en sorte que la variance soit plus petite. La réduction de l'erreur avec N est assez lente : pour gagner un facteur 10 (par exemple, passer de 10% à 1% d'erreur), il faut multiplier le temps de calcul par 100 ! Construire des estimateurs alternatifs à (2.13) et ayant une variance plus petite est donc souhaitable. C'est ce qu'on appelle les techniques de réduction de variance. Nous en présentons quatre célèbres exemples dans cette section, selon une présentation inspirée de [1].

Variables antithétiques

Pour simplifier les notations, on se place dans le cas où $a = -1/2$ et $b = 1/2$, le cas général s'en déduisant par un changement de variable affine. La remarque fondamentale sous-tendant la méthode est que si les noeuds sont distribués selon une loi uniforme sur $[-1/2, 1/2]^d$, *i.e.* $x_i \sim \mathcal{U}([-1/2, 1/2]^d)$, alors les noeuds $-x_i$ sont également distribués selon une loi uniforme sur $[-1/2, 1/2]^d$. Ainsi, on peut considérer l'estimateur

$$J_N = \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i) + f(-x_i)}{2},$$

les termes $f(x_i) + f(-x_i)$ étant indépendants si les x_i sont indépendants. La consistance de cet estimateur est toujours donnée par la loi forte des grands nombres. La convergence est encore donnée par un théorème de la limite centrale, la variance asymptotique étant à présent (voir Exercice 2.6)

$$\sigma_J(f)^2 = \int_{[-1/2,1/2]^d} \left(\frac{f(x) + f(-x)}{2} \right)^2 dx - \left(\int_{[-1/2,1/2]^d} f \right)^2. \tag{2.16}$$

Si le coût de calcul est compté en nombre d'évaluations de la fonction f, on voit que le calcul de J_N est deux fois plus coûteux que celui de l'estimateur simple I_N donné par (2.13). Pour que cette méthode soit efficace, il faut donc que la variance de J_N soit réduite d'au moins un facteur 2 (pour être compétitif par rapport à une stratégie simple qui consiste à utiliser l'estimateur standard (2.13) avec $2N$ tirages). Il se trouve que c'est le cas au moins si $d = 1$ et f est monotone. Des arguments heuristiques [1] et numériques confirment l'intérêt de cette méthode dans le cas général.

Exercice 2.6 (Preuve de la réduction de variance si $d = 1$ et f monotone). *On note σ_I^2 la variance asymptotique de l'estimateur I_N, et σ_J^2 la variance asymptotique de l'estimateur J_N. On commencera par prouver* (2.16)*. Montrer ensuite qu'on a toujours*

$$\sigma_J(f)^2 \leqslant \sigma_I(f)^2.$$

On cherche à présent des hypothèses sous lesquelles on a

$$\sigma_J(f)^2 \leqslant \frac{1}{2}\sigma_I(f)^2. \tag{2.17}$$

On suppose que $f : \mathbb{R} \to \mathbb{R}$ est croissante. Montrer que (2.17) *est équivalent à*

$$\int_{-1/2}^{1/2} f(x)f(-x)\,dx - \left(\int_{-1/2}^{1/2} f \right)^2 \leqslant 0.$$

Pour montrer cette dernière inégalité, on introduira la fonction

$$u(x) = \int_{-1/2}^{x} f(-y)\,dy - \left(x + \frac{1}{2} \right) \int_{-1/2}^{1/2} f$$

et on montrera que $u \geqslant 0$. Ceci donnera en effet

$$\int_{-1/2}^{1/2} f'(x)u(x)\,dx \geqslant 0,$$

et il restera à conclure.

Variable de contrôle

L'idée de base de la méthode de variable de contrôle est d'introduire une fonction g proche en un certain sens de f, et dont on sait calculer analytiquement l'intégrale. La décomposition

$$\int_{[a,b]^d} f(x)\,dx = \int_{[a,b]^d} \Big(f(x) - g(x)\Big)dx + \int_{[a,b]^d} g(x)\,dx$$

permet ainsi de proposer l'estimateur suivant : pour une suite de noeuds aléatoires $x_i \sim \mathcal{U}([a,b]^d)$ indépendants et identiquement distribués,

$$J_N(f) = I(g) + \frac{(b-a)^d}{N}\sum_{i=1}^{N} f(x_i) - g(x_i), \qquad I(g) = \int_{[a,b]^d} g(x)\,dx.$$

La loi forte des grands nombres assure la consistance de cet estimateur, alors que le théorème de la limite centrale montre que la convergence est en $\sigma_J(f)/\sqrt{N}$ avec

$$\sigma_J(f)^2 = (b-a)^d \int_{[a,b]^d} (f-g)^2 - \left(\int_{[a,b]^d} f - g\right)^2.$$

On a bien $\sigma_J(f)^2 \ll \sigma_I(f)$ lorsque g est "proche" de f (penser au cas où $g = f + h$ avec h petit par rapport à f dans une certaine norme). En revanche, il est tout à fait possible que $\sigma_J(f)^2 \geqslant \sigma_I(f)$ si on choisit mal g !

Stratification

La méthode de stratification est le pendant des méthodes composites pour le cas stochastique. On décompose le domaine d'intégration $\mathcal{D} = [a,b]^d$ en une partition de sous-domaines $\mathcal{D}_m$:

$$\mathcal{D} = \bigcup_{m=1}^{M} \mathcal{D}_m,$$

et on tire un nombre de points N_m dans chaque sous-domaine $\mathcal{D}_m$, ces points étant indépendants et identiquement distribués selon la loi uniforme sur $\mathcal{D}_m$:

$$x_i^m \sim \mathcal{U}(\mathcal{D}_m),$$

et le nombre de points par sous-domaine étant directement proportionnel à la taille du sous-domaine :

$$N_m = N\frac{|\mathcal{D}_m|}{|\mathcal{D}|}.$$

Notons que le nombre total de points est toujours fixé à N. On construit ensuite l'estimateur

$$J_N^M = \frac{|\mathcal{D}|}{N}\sum_{m=1}^{M}\sum_{i=1}^{N_m} f(x_i^m) = \sum_{m=1}^{M}\frac{|\mathcal{D}_m|}{N_m}\sum_{i=1}^{N_m} f(x_i^m).$$

Noter que, par la loi forte des grands nombres, on a la convergence presque sûre suivante :

$$\frac{|\mathcal{D}_m|}{N_m}\sum_{i=1}^{N_m} f(x_i^m) \xrightarrow[N_m\to+\infty]{} \int_{\mathcal{D}_m} f,$$

ce qui montre la consistance de l'estimateur J_N^M. Lorsque $N \to +\infty$, on peut appliquer un théorème de la limite centrale sous-domaine par sous-domaine, et obtenir ainsi que l'estimateur J_N^M est asymptotiquement normal, sa variance étant obtenue en sommant les variances sur chacun des sous-domaines.

On peut montrer que l'on diminue toujours la variance asymptotique par rapport à l'estimateur I_N. C'est d'ailleurs l'objet de l'Exercice 2.7.

Exercice 2.7 (Variance de l'estimateur stratifié). *Montrer que la variance asymptotique de la méthode de stratification (estimateur J_N^M) est toujours inférieure à celle de la méthode directe (estimateur I_N). On commencera par montrer que*

$$\frac{N}{|\mathcal{D}|}\mathrm{Var}\Big(J_N^M\Big) = \sum_{m=1}^{M}\int_{\mathcal{D}_m}\Big(f - \frac{1}{|\mathcal{D}_m|}\int_{\mathcal{D}_m} f\Big)^2.$$

Qu'est-ce que ce résultat suggère comme décomposition en sous-domaines ?

Echantillonnage d'importance

Une densité uniforme de noeuds de quadrature est probablement sous-optimale pour une fonction f fixée, surtout en dimension grande... L'idée de l'échantillonnage d'importance est de concentrer les noeuds de quadrature dans les zones qui comptent le plus pour f. Cela revient à choisir une mesure de probabilité de référence $g(x)\,dx$ (une fonction positive intégrable, d'intégrale 1) et à générer des noeuds $x_i \sim g(x)\,dx$ indépendants (ce qui demande donc que g soit suffisamment simple pour que l'on puisse facilement l'échantillonner...), puis à calculer l'estimateur

$$J_{N,g} = \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{g(x_i)}.$$

Une fois de plus, la consistance est donnée par la loi forte des grands nombres :

$$J_{N,g} \xrightarrow[N\to+\infty]{} \int_{[a,b]^d}\Big(\frac{f}{g}\Big)\,g = \int_{[a,b]^d} f.$$

La variance asymptotique ainsi que donnée par le théorème de la limite centrale est

$$\sigma_J(f)^2 = \int_{[a,b]^d}\frac{f^2}{g} - \Big(\int_{[a,b]^d} f\Big)^2.$$

On a donc $\sigma_J(f)^2 \leqslant \sigma_I(f)^2$ si

$$\int_{[a,b]^d}\frac{f^2}{g} \leqslant (b-a)^d\int_{[a,b]^d} f^2.$$

Notons toutefois que la variance peut également être augmentée par un choix malheureux de la fonction g ! Le choix optimal (au sens où la variance asymptotique est minimisée) correspond au cas où $g \propto |f|$.

Exercice 2.8 (Fonction d'importance optimale). *Montrer que la fonction d'importance optimale (qui minimise la variance de l'estimateur $J_{N,g}(f)$) est la mesure de probabilité de densité*

$$g(x) = \frac{|f(x)|}{\displaystyle\int_{[a,b]^d}|f|}.$$

On utilisera pour ce faire l'inégalité de Cauchy-Schwarz.

2.3.3 Méthodes de quasi Monte-Carlo

Les noeuds de quadrature générés par les méthodes de Monte-Carlo ne sont pas toujours bien répartis : on observe qu'il y a localement des zones mieux échantillonnées que d'autres. Bien sûr, ces déséquilibres disparaissent lorsque le nombre de points devient très grand, mais il reste un signe rémanent de cela : la vitesse de convergence en $1/\sqrt{N}$.

Les méthodes de quasi-Monte Carlo ont pour objectif de générer une suite de noeuds x_i qui ne sont pas des variables aléatoires indépendantes (en fait, ces points sont mêmes obtenus par une méthode déterministe !), mais qui sont tout de même distribués plus ou moins uniformément, et qui forment une suite qui a l'air aléatoire – un peu dans l'esprit des nombres pseudo-aléatoires évoqués à la fin de la Section 2.3.1. On souhaite en tout cas que

$$I_N(f) = \frac{1}{N}\sum_{i=1}^{N} f(x_i) \xrightarrow[N\to+\infty]{} I(f) = \int_{[0,1]^d} f(x)\,dx. \tag{2.18}$$

Exemple 2.4 (Suite de Hammersley). On fixe un nombre N de points et on se place en dimension d. On choisit des entiers $b_1, \dots, b_{d-1}$. Les éléments de la suite $\{x_1, \dots, x_N\}$ sont obtenus de la manière suivante :

$$x_i = \left(\frac{i}{N}, \varphi_{b_1}(i), \dots, \varphi_{b_{d-1}}(i)\right) \in [0,1]^d.$$

Les fonctions $\varphi_b : \mathbb{N} \to [0,1]$ sont définies de la manière suivante. Soit $i \in \mathbb{N}$, on décompose i dans la base b selon

$$i = d_0 + d_1 b + d_2 b^2 + \cdots + d_k b^k,$$

où chacun des coefficients d_j est un entier compris entre 0 et $b-1$. Ceci correspond à l'écriture (en base b) $i = d_k d_{k-1} \dots d_1 d_0$. On pose ensuite

$$\varphi_b(i) = \frac{d_0}{b} + \frac{d_1}{b^2} + \cdots + \frac{d_k}{b^{k+1}},$$

ce qui correspond à l'écriture (en base b) $\varphi_b(i) = 0.d_0 d_1 \dots d_k$. On montre facilement que $\varphi_b(i) \in [0,1]$. Notons que si on veut ajouter quelques points (passer de N à $N+1$ points), il faut recalculer toute la suite : on ne peut pas juste ajouter un point.

L'analyse *a priori* directe de (2.18) repose sur une quantification de l'équidistribution appelée discrépance : notant $\mathcal{R}$ l'ensemble des rectangles dont un sommet est 0,

$$D_N^* = \sup_{R\in\mathcal{R}} \left| \frac{1}{N}\mathrm{Card}\{x_n \in R,\ n = 1, \dots, N\} - |R| \right|.$$

Insistons sur le fait que les suites aléatoires ne sont pas équidistribuées ! L'erreur commise en approchant l'intégrale à calculer par (2.18) est donnée par l'inégalité de Koksma-Hlawka

$$|I_N(f) - I(f)| \leqslant D_N^* V(f), \tag{2.19}$$

où $V(f)$ est une quantité qui ne dépend que de la fonction à intégrer. Dans le cas simple de l'intégration en dimension 1, on a

$$V(f) = \int_0^1 |f'|.$$

Pour les dimensions supérieures, $V(f)$ est donnée par une définition récursive compliquée (impliquant notamment des dérivées d'ordre d). Insistons également sur le fait que (2.19) donne une borne supérieure déterministe (alors que les estimations d'erreurs pour les algorithmes stochastiques donnent des limites en loi), produit d'un facteur dépendant de la fonction et d'un facteur ne dépendant que du nombre de points. En travaillant bien (et en utilisant la théorie des nombres), on peut construire des séquences quasi-aléatoires, qui sont telles que

$$D_N^* \leqslant C\frac{(\ln N)^k}{N},$$

ce qui donne une vitesse de convergence qui est presque en $1/N$, au lieu de $N^{-1/2}$ pour les méthodes de Monte-Carlo.

Notons toutefois qu'une limitation importante de ces méthodes est qu'il faut fixer la valeur de N à l'avance, et qu'il faut recalculer tous les points si on change N ! Certaines méthodes de quasi Monte Carlo sont toutefois dites hiérarchiques, et permettent de limiter le recalcul des points.

2.3.4 Ouverture : méthodes fondées sur les chaines de Markov

Evoquons rapidement pour finir des techniques utiles pour le calcul de propriétés moyennes en dimension très grande, où on cherche à approcher la moyenne d'une fonction par rapport à une mesure de probabilité μ de densité g (connue à un facteur multiplicatif près), par une moyenne empirique adéquate :

$$\langle f\rangle = \int_{\mathcal{D}} f\,d\mu = \frac{\displaystyle\int_{\mathcal{D}} f(x)g(x)\,dx}{\displaystyle\int_{\mathcal{D}} g(x)\,dx} \simeq \frac{1}{N}\sum_{i=1}^{N} f(x_i).$$

Dans les systèmes en très grande dimension, il est difficile de tirer des points indépendants selon g parce que les mesures de probabilité ont typiquement des modes très concentrés, séparés par de grandes zones de probabilité très faible. Une idée plus réaliste est donc de considérer une chaîne de points corrélés, où un nouveau point est obtenu par une perturbation aléatoire du précédent. Ceci assure ainsi une bonne exploration locale mais peut aussi mener à des problèmes de métastabilité (fausse convergence) si l'ensemble des points générés reste coincé dans un des modes locaux de la mesure de probabilité.

Il faut également assurer que la perturbation que l'on applique au point est consistante avec la mesure de probabilité que l'on cherche à échantillonner. Un algorithme très classique pour ce faire est l'algorithme de Metropolis, qui consiste, partant d'un point x_0 donné, à

- proposer un nouveau point $\widetilde{x}_{i+1} = x_i + \sigma G_i$ en appliquant une perturbation Gaussienne de variance σ^2 ;
- calculer le rapport $\alpha_i = g(\widetilde{x}_{i+1})/g(x_i)$;
- accepter $\widetilde{x}_i$ avec une probabilité α_i, auquel cas on pose $x_{i+1} = \widetilde{x}_i$, et poser sinon $x_{i+1} = x_i$.

Insistons sur le fait que les points ainsi générés sont corrélés : on a ce qu'on appelle une chaîne de Markov (le nouveau point ne dépend du passé que par le biais du point précédent). On peut toutefois montrer qu'on a toujours une loi forte des grands nombres ainsi qu'un théorème de la limite centrale, avec la variance

$$\sigma^2(f) = \mathrm{Var}_\mu(f) + 2\sum_{n=1}^{+\infty} \mathbb{E}_\mu\Big[\big(f(x_0) - \mathbb{E}_\mu(f)\big)\big(f(x_n) - \mathbb{E}_\mu(f)\big)\Big],$$

les espérances ci-dessus étant par rapport à toutes les conditions initiales x_0 distribuées selon μ, et pour toutes les réalisations de la dynamique partant de ces conditions initiales.

3

Intégration numérique des équations différentielles ordinaires

L'objectif de ce chapitre est de présenter quelques méthodes numériques pour approcher des solutions d'équations différentielles ordinaires (EDOs), qui sont un problème de Cauchy de la forme suivante

$$\dot{y}(t) = f(t, y(t)), \qquad y(0) = y_0, \tag{3.1}$$

où $y : \mathbb{R}_+ \to \mathbb{R}^d$ est une fonction du temps $t \geqslant 0$ à valeurs vectorielles, et $f : \mathbb{R}^d \to \mathbb{R}^d$ est un champ de vecteurs. On peut réécrire ce problème sous une formulation intégrale, qui peut être plus agréable pour l'étude théorique car cela demande moins d'hypothèses de régularité sur les objets en jeu, et qui est également utile pour proposer des schémas numériques :

$$y(t) = y_0 + \int_0^t f(s, y(s))\, ds. \tag{3.2}$$

Cette formulation est équivalente à (3.1) si f est continue.

Nous commençons par présenter quelques applications en Section 3.1 (et notamment un problème modèle, la dynamique du système solaire). Nous nous tournons ensuite vers l'étude mathématique du problème continu (3.1) en Section 3.2, et discutons en particulier son caractère bien posé. Une fois ceci établi, on peut sereinement se tourner vers l'approximation numérique de (3.1) par le biais des méthodes à un pas, les plus simples (voir Section 3.3). Enfin, on peut mener plus loin l'étude de systèmes qui ont des propriétés ou une structure particulières : c'est l'objet de la Section 3.4.

3.1 Motivation

On a besoin dans certains cas de savoir calculer numériquement avec une grande précision la solution d'une EDO : par exemple pour déterminer la trajectoire de la fusée qui va mettre en orbite un satellite, ou de la sonde spatiale qui va passer au ras de Jupiter pour ensuite aller explorer les confins de notre univers, ou de la météorite que l'on voit arriver près de la Terre (nous touchera-t-elle ou non ?). Dans ces cas, on se donne un horizon de temps fini et on cherche à reproduire au mieux la trajectoire du système sur ce temps.

Il y a d'autres situations où on s'intéresse plutôt à un résultat en temps long, par exemple la convergence vers une trajectoire périodique ou un cycle : c'est cet élément géométrique asymptotique qui sera l'objet de nos soins. Citons par exemple l'équation de Lotka-Volterra, qui est un modèle simplifié de dynamique des populations ; ou l'intégration en temps long de la dynamique Hamiltonienne pour calculer des moyennes microcanoniques en physique statistique, ou en dynamique céleste, pour déterminer la stabilité des orbites d'un système planétaire.

Citons également des problèmes mélangeant plusieurs échelles de temps : un cas frappant est celui de la cinétique chimique entrant dans les modèles de pollution de l'air ou en génie chimique. Certaines transformations dans le systèmes sont très rapides et/ou oscillantes (constantes de réaction très grandes ou concentrations importantes), alors que d'autres sont très lentes. Une bonne méthode numérique devrait pouvoir traiter toutes ces échelles de temps simultanément, et ne pas caler le pas de temps sur les événements les plus rapides, sans quoi les événements les plus lents ne pourront pas être résolus.

Enfin, les méthodes d'intégration en temps des EDOs sont une brique fondamentale pour l'intégration en temps d'équations plus compliquées : (i) les équations aux dérivées partielles, où il faut considérer à la fois une discrétisation en espace et en temps. Citons par exemple les problèmes de la dynamique des fluides (météorologie, océanographie, etc), de la mécanique des solides (problèmes de rupture et ruine, chargements divers, etc) et de la mécanique quantique (effet tunnel) ; (ii) les équations différentielles stochastiques, rencontrées notamment dans le cadre de la finance quantitative ou en physique statistique numérique.

3.1.1 Un exemple précis : la dynamique céleste

On décrit ici un système qui correspond à la partie « extérieure » du système solaire : on représente Jupiter, Saturne, Uranus, Neptune et Pluton (positions respectives q_i pour $i = 1, \dots, 5$), et le Soleil auquel on agrège en fait les quatre planètes intérieures que sont Mercure, Vénus, la Terre et Mars (position q_0). L'énergie potentielle du sytème est

$$V(q) = \sum_{0 \leqslant i < j \leqslant 5} v_{ij}(|q_i - q_j|), \qquad v_{ij}(r) = -G\frac{m_i m_j}{r}.$$

On utilise des unités réduites pour l'implémentation informatique (afin de manipuler des quantités qui sont toutes d'ordre 1). L'unité de masse est donnée par la masse du soleil $1,9891 \times 10^{30}$ kg ; l'unité de longueur est la distance Terre-Soleil, à savoir 149 597 870 km ; et l'unité de temps est un jour sur Terre, soit $8,64 \times 10^3$ s. Dans ces unités, la constante de gravitation $G = 6,67384 \times 10^{-11}$ $\mathrm{m^3 kg^{-1} s^{-2}}$ vaut $2,95995 \times 10^{-4}$. Les masses des planètes sont notées m_i, et $p_i = m_i v_i$ est leur quantité de mouvement.

L'énergie totale du système dans la configuration $(q, p) \in \mathbb{R}^{6N}$ est donnée par le Hamiltonien

$$H(q,p) = \sum_{i=1}^{N} \frac{p_i^2}{2m_i} + V(q).$$

L'évolution en temps est régie par la dynamique Hamiltonienne

$$\begin{cases} \dot{q}_i(t) = \dfrac{\partial H}{\partial p_i} = \dfrac{p_i(t)}{m_i}, \\ \dot{p}_i(t) = -\dfrac{\partial H}{\partial q_i} = -\nabla_{q_i} V(q(t)). \end{cases} \tag{3.3}$$

Noter que, quitte à introduire l'inconnue $y = (q, p)$, on peut récrire cette dynamique sous la forme générale

$$\dot{y}(t) = f(y), \qquad f(y) = \begin{pmatrix} \nabla_p H \\ -\nabla_q H \end{pmatrix}. \tag{3.4}$$

Un calcul simple (le faire en exercice) montre que l'énergie du système est constante au cours du temps : $H(q(t), p(t)) = H(q_0, p_0)$. Une notion de stabilité pertinente est par exemple la conservation de l'énergie totale en temps long.

3.2 Etude du problème continu

Pour déterminer le caractère bien posé du problème (3.1), on va successivement étudier l'existence et l'unicité des solutions (Section 3.2.1), puis leur stabilité (Section 3.2.2).

3.2.1 Existence et unicité des solutions

Existence et unicité locales

Pour discuter l'existence et l'unicité des solutions du problème (3.1), on utilise le théorème de Cauchy–Lipschitz, qui donne l'existence locale et l'unicité si le champ de force f est localement Lipschitzien : pour tout $(t_0, y_0) \in \mathbb{R}_+ \times \mathbb{R}^d$, il existe $r, \tau, L > 0$ (dépendant de t_0, y_0 *a priori*) tels que

$$\forall (t, y_1, y_2) \in]t_0 - \tau, t_0 + \tau[\times B(y_0, r)^2, \qquad |f(t, y_1) - f(t, y_2)| \leqslant L|y_1 - y_2|,$$

où $B(y_0, r)$ est la boule (ouverte) de centre y_0 et de rayon r. Cette condition est vérifiée par exemple si les dérivées partielles $\partial_t f$ et $\partial_{y_j} f$ (pour tout $j = 1, \dots, d$) sont continues. Cela définit une unique solution sur un intervalle de temps maximal $[0, t_{\max}[$. Si on n'a pas de solution globale (au sens où $t_{\max} < +\infty$), alors on a explosion de la solution en temps fini : $|y(t)| \to +\infty$ lorsque $t \to t_{\max}$.

Existence et unicité globales

L'existence et l'unicité de la solution globale est assurée dans certains cas.

(i) Le premier cas est celui où $f(t, \cdot)$ est uniformément Lipschitzienne en y, avec une constante de Lipschitz $L(t)$ continue.

(ii) Le second cas est celui où la croissance de f est au plus affine :

$$|f(t, y)| \leqslant c(t) + L(t)|y|,$$

avec des fonctions c, L localement intégrables. On a en effet, en partant de la formulation intégrale (3.2),

$$|y(t)| \leqslant |y_0| + \int_0^t \Big(c(s) + L(s)|y(s)| \Big) ds. \tag{3.5}$$

En introduisant

$$M(t) = \int_0^t L(s)|y(s)|\, ds \geqslant 0, \qquad a(t) = L(t)\Big(|y_0| + \int_0^t c(s)\, ds \Big),$$

on peut reformuler (3.5) sous la forme $\dot{M}(t) \leqslant a(t) + L(t)M(t)$, d'où, par le lemme de Gronwall, [1]

$$0 \leqslant M(t) \leqslant \int_0^t a(s) \exp\left(\int_s^t L \right) ds.$$

En reportant dans (3.5), on obtient bien une borne supérieure finie pour $|y(t)|$.

1. Rappelons rapidement le calcul : on note que

(iii) Un dernier cas est celui où il existe une fonction de Lyapounov, c'est-à-dire une fonction $W \in C^1(\mathbb{R}^d)$ telle que $W(x) \to +\infty$ lorsque $|x| \to +\infty$. Comme W est alors uniformément minorée, on peut lui ajouter une constante de telle manière à ce que $W \geqslant 1$. Enfin, on demande que

$$f(x) \cdot \nabla W(x) \leqslant c < +\infty.$$

Dans ce cas, on a alors

$$\frac{d}{dt}\Big[W\Big(y(t)\Big)\Big] = \Big(f \cdot \nabla W\Big)(y(t)) \leqslant c,$$

ce qui montre que $W(y(t)) \leqslant W(y_0)\,\mathrm{e}^{ct}$, et assure ainsi que la norme de la solution ne peut pas exploser en temps fini. Cette technique est très utile pour établir l'existence et l'unicité dans le cas où f n'est pas globalement Lipschitzienne et croît plus que linéairement à l'infini. On peut appliquer cela au système très simple $\dot{y}(t) = -y(t)^3$ pour lequel $W(x) = 1 + x^4$ est une fonction de Lyapounov.

3.2.2 Stabilité

On suppose à présent que (3.1) admet une unique solution sur un intervalle de temps $[0, t_{\max}[$. Nous allons nous pencher sur la dépendance de la solution en les données, plus précisément vérifier que cette dépendance est continue pour commencer : c'est la stabilité au sens de Lyapounov. Nous verrons ensuite une version plus forte, la stabilité asymptotique.

Mettons en garde le lecteur contre le fait qu'il existe beaucoup d'autres notions de stabilité pour des classes de systèmes particuliers – notamment la stabilité autour d'une position d'équilibre ou d'une orbite périodique, l'existence d'une mesure empirique limite reliée à une notion d'ergodicité, etc. Nous n'évoquerons pas ces notions ici.

Stabilité au sens de Lyapounov

On se place sur un intervalle de temps borné $I = [t_0, t_1] \subset [0, t_{\max}[$, et on considère une perturbation de (3.1) de la forme

$$\dot{z}(t) = f(t, z(t)) + \delta(t, z(t)), \qquad z(t_0) = y(t_0) + \delta_0, \tag{3.6}$$

avec δ_0 et $\delta(t, z)$ "petits" (nous allons préciser cela tout de suite). L'idée est que la trajectoire du système perturbée reste assez proche de la trajectoire de référence si la perturbation n'est pas trop grande.

Définition 3.1 (Stabilité au sens de Lyapounov). *Supposons que δ soit continue et bornée :*

$$|\delta_0| \leqslant \varepsilon, \qquad \sup_{t \in I} \|\delta(t, \cdot)\|_{L^\infty} = \sup_{(t,x) \in I \times \mathbb{R}^d} |\delta(t, x)| \leqslant \varepsilon. \tag{3.7}$$

On dit que (3.1) *est stable au sens de Lyapounov sur I s'il existe $C > 0$ tel que la solution de* (3.6) *satisfasse*

$$\forall t \in I, \qquad |y(t) - z(t)| \leqslant C\varepsilon.$$

$$\frac{d}{dt}\left[M(t)\,\exp\left(-\int_0^t L(s)\,ds\right)\right] = \Big(\dot{M}(t) - L(t)M(t)\Big)\exp\left(-\int_0^t L(s)\,ds\right) \leqslant a(t)\,\exp\left(-\int_0^t L(s)\,ds\right),$$

d'où, par intégration et en tenant compte du fait que $M(0) = 0$,

$$M(t)\,\exp\left(-\int_0^t M(s)\,ds\right) \leqslant \int_0^t a(s)\,\exp\left(-\int_0^s M(r)\,dr\right)\,ds,$$

ce qui permet de conclure en multipliant les deux membres de l'inégalité par $\exp\left(\int_0^t M(s)\,ds\right)$.

Insistons sur le fait que la constante C dépend de f bien sûr, mais surtout de I, la dépendance en fonction du temps étant typiquement exponentielle. La propriété de stabilité est vérifiée par exemple si f uniformément Lipschitzienne de constante L. La preuve est la suivante : partant de

$$y(t) - z(t) = \delta_0 + \int_{t_0}^{t} \Big(f(y(s)) - f(z(s)) \Big) ds + \int_{t_0}^{t} \delta(s, z(s))\, ds,$$

on peut écrire que

$$|y(t) - z(t)| \leqslant (1 + t - t_0)\varepsilon + L \int_{t_0}^{t} |y(s) - z(s)|\, ds \leqslant (1 + t_1 - t_0)\varepsilon + L \int_{t_0}^{t} |y(s) - z(s)|\, ds,$$

et on conclut par une inégalité de Gronwall, avec le choix $C = (1 + t_1 - t_0)\mathrm{e}^{L|t_1 - t_0|}$ (on voit là la dépendance exponentielle en fonction de la taille de l'intervalle d'intégration).

Si f n'est pas uniformément Lipschitzienne, l'étude de la stabilité demande plus d'attention. Si le champ de force est localement Lipschitzien, on peut commencer par montrer que la trajectoire reste dans un compact (en utilisant une fonction de Lyapunov) pour se ramener au cas précédent.

Stabilité asymptotique

La stabilité asymptotique considère des perturbations du type (3.6) sur des intervalles de temps infiniment longs $[t_0, +\infty[$ (pour un certain $t_0 \geqslant 0$, typiquement $t_0 = 0$), sous la condition que la perturbation tende vers 0 en temps long. Cette notion est pertinente pour des systèmes dissipatifs.

Définition 3.2 (Stabilité asymptotique). *Supposons que* (3.7) *soit satisfaite avec* $I = [t_0, +\infty[$, *et que*

$$\sup_{x \in \mathbb{R}^d} |\delta(t, x)| \xrightarrow[t \to +\infty]{} 0.$$

On dit que (3.1) *est asymptotiquement stable si la solution de* (3.6) *est telle que*

$$|y(t) - z(t)| \xrightarrow[t \to +\infty]{} 0.$$

Un exemple de système asymptotiquement stable est la classe des systèmes linéaires dissipatifs étudiés à la Section 3.4.1.

3.3 Approximation par les méthodes à un pas

On va à présent décrire des méthodes numériques pour approcher la solution de (3.1) sur un intervalle de temps fini $[0, T]$. Plus précisément, on va considérer des temps $t_0 = 0 < t_1 < \cdots < t_N = T$, et on va noter y^n l'approximation numérique de la solution exacte $y(t_n)$. Par la suite, on notera $\Delta t_n = t_{n+1} - t_n$ les incréments de temps strictement positifs. Souvent, on choisira un pas de temps uniforme $\Delta t > 0$, auquel cas $t_n = n\Delta t$, le nombre total de pas d'intégration étant [2] $N = T/\Delta t$.

Pour rester le plus simple possible, nous évoquerons seulement les méthodes à un pas, et non les méthodes multi-pas, bien qu'elles soient plus précises à coût de calcul fixé (cependant, leur stabilité demande une attention particulière, voir l'Exercice 3.3 pour une illustration).

2. On suppose que ce nombre est entier ; sinon on peut toujours changer un peu le pas de temps pour que ce soit le cas, en prenant par exemple la partie entière de $T/\Delta t$.

3.3.1 Principe de l'approximation

La construction des méthodes à un pas repose sur une discrétisation de la formulation intégrale (3.2) sur l'intervalle de temps $[t_n, t_{n+1}]$ par une règle de quadrature. De manière abstraite, on peut ainsi écrire une relation de récurrence permettant de calculer itérativement la trajectoire numérique

$$y^{n+1} = y^n + \Delta t_n \Phi_{\Delta t_n}(t_n, y^n), \tag{3.8}$$

où $\Phi_{\Delta t_n}(t_n, y^n)$ est une approximation de

$$\frac{1}{t_{n+1} - t_n} \int_{t_n}^{t_{n+1}} f\Big(s, y(s)\Big) ds.$$

Les méthodes ainsi obtenues sont appelées méthodes de Runge–Kutta. Elles sont décomposées en deux catégories selon qu'elles sont *explicites* (la nouvelle configuration peut être obtenue directement de la précédente) ou *implicites* (pour obtenir la nouvelle configuration, il faut résoudre un problème nonlinéaire). Dans ce second cas, le schéma numérique n'est pas spontanément écrit sous la forme (3.8). Toutefois, on préfère toujours écrire l'incrément $y^{n+1} - y^n$ comme une fonction de y^n seulement pour mettre en exergue le fait qu'un schéma numérique pour les EDOs fonctionne de manière itérative : il suffit de connaitre une approximation de l'état du système y^n au temps t_n, et l'incrément de temps Δt_n, pour en déduire une approximation de l'état au temps $t_n + \Delta t_n$.

Donnons à présent quelques exemples de méthodes numériques pour illustrer notre propos :

(1) Méthodes explicites :

(i) Euler explicite : $y^{n+1} = y^n + \Delta t_n\, f(t_n, y^n)$;

(ii) méthode de Heun : $y^{n+1} = y^n + \dfrac{\Delta t_n}{2}\Big(f(t_n, y^n) + f\big(t_{n+1}, y^n + \Delta t_n f(t_n, y^n)\big)\Big)$;

(iii) schéma de Runge–Kutta d'ordre 4 : on calcule les points intermédiaires

$$\begin{cases} F_1 = f(t_n, y^n) \\ F_2 = f\left(t_n + \dfrac{\Delta t_n}{2}, y^n + \dfrac{\Delta t_n}{2} F_1\right) \\ F_3 = f\left(t_n + \dfrac{\Delta t_n}{2}, y^n + \dfrac{\Delta t_n}{2} F_2\right) \\ F_4 = f(t_n + \Delta t_n, y^n + \Delta t_n F_3), \end{cases}$$

et on pose

$$y^{n+1} = y^n + \Delta t\, \frac{F_1 + 2F_2 + 2F_3 + F_4}{6} \ ;$$

(2) Méthodes implicites :

(i) Euler implicite : $y^{n+1} = y^n + \Delta t_n\, f(t_{n+1}, y^{n+1})$;

(ii) méthode des trapèzes (ou Crank–Nicolson) : $y^{n+1} = y^n + \dfrac{\Delta t_n}{2}\Big(f(t_n, y^n) + f(t_{n+1}, y^{n+1})\Big)$;

(iii) méthode du point milieu : $y^{n+1} = y^n + \Delta t_n f\left(\dfrac{t_n + t_{n+1}}{2}, \dfrac{y^n + y^{n+1}}{2}\right)$.

On peut bien sûr construire des méthodes plus compliquées et plus précises, et nous renvoyons le lecteur à la bibliographique pour des méthodes de Runge–Kutta d'ordre supérieur (explicites ou implicites). Pour les méthodes explicites, on identifie assez simplement la fonction d'incrément $\Phi_{\Delta t}$ dans (3.8). Pour les schémas implicites, cette tâche est moins facile. Prenons l'exemple du schéma d'Euler implicite : l'application $\Phi_{\Delta t}$ est définie de manière implicite par la relation

$$\Phi_{\Delta t_n}(t_n, y^n) = f\Big(t_n + \Delta t_n, y^n + \Delta t_n \Phi_{\Delta t_n}(t_n, y^n)\Big).$$

Comme nous allons le montrer dans le point suivant, on peut s'assurer que $\Phi_{\Delta t_n}(t_n, y^n)$ est bien définie si Δt_n est assez petit.

Un mot sur l'implémentation des schémas implicites.

Dans les schémas implicites, il faut résoudre une équation nonlinéaire pour obtenir l'approximation y^{n+1} partant de y^n. Il y a donc un surcoût de calcul dans l'utilisation de ces schémas, mais qui vaut souvent le coup car on a une stabilité accrue et on peut utiliser des pas de temps plus grands.

Avant de chercher à résoudre numériquement l'équation donnant y^{n+1}, il faut déjà garantir que y^{n+1} existe et est unique ! On peut utiliser pour ce faire un théorème des fonctions implicites, ou un théorème de point-fixe de Banach. Par exemple, le schéma d'Euler implicite $y^{n+1} = y^n + \Delta t_n\, f(t_{n+1}, y^{n+1})$ peut se réécrire

$$y^{n+1} = F(y^{n+1}), \qquad F(y) = y^n + \Delta t_n f(t_{n+1}, y).$$

On a existence et unicité de y^{n+1} lorsque

$$\left\| \frac{\partial F}{\partial y} \right\|_{L^\infty} = \Delta t_n \left\| \frac{\partial f}{\partial y}(t_{n+1}, \cdot) \right\|_{L^\infty} < 1.$$

On pourrait relâcher un peu cette condition en demandant que $\Delta t\, \Lambda_{t_n} < 1$, où Λ_{t_n} est la constante de Lipschitz de $f(t_{n+1}, \cdot)$. Ceci impose dans tous les cas une borne supérieure sur les pas de temps admissibles. Notons qu'une analyse plus fine permettrait de remplacer la norme L^∞ globale par une majoration locale de la dérivée partielle au point y^n, et de considérer ainsi des incréments de temps variables (voir Section 3.3.4 pour des stratégies de pas de temps adaptatifs).

Le calcul pratique des itérations d'un schéma implicite peut se faire par une méthode numérique s'inspirant de la stratégie de point fixe utilisée pour montrer l'existence et l'unicité de y^{n+1} : on part d'un premier essai obtenu par une méthode explicite, que l'on affine ensuite par des itérations de point-fixe, d'où le nom de stratégie « prédicteur/correcteur ». Illustrons cela pour le schéma d'Euler implicite. On peut partir d'un état obtenu par un schéma d'Euler explicite

$$z^{n+1,0} = y^n + \Delta t_n f(t_n, y^n),$$

et le corriger ensuite par des itérations de point-fixe selon

$$z^{n+1,k+1} = y^n + \Delta t_n f(t_{n+1}, z^{n+1,k}).$$

Sous les bonnes hypothèses précédentes, on a $z^{n+1,k} \xrightarrow[k\to+\infty]{} y^{n+1}$. En pratique, on n'effectue qu'un nombre fini d'itérations de point-fixe, en utilisant un critère de convergence tel que

$$|z^{n+1,k+1} - z^{n+1,k}| \leqslant \varepsilon$$

pour une tolérance $\varepsilon > 0$ donnée. On observe souvent une très bonne convergence dans les itérations de point-fixe, avec des erreurs relatives de l'ordre de 10^{-8} en quelques itérations. Une alternative aux itérations de point-fixe est d'utiliser un algorithme de Newton, qui a une convergence plus rapide, mais qui demande que l'on parte suffisamment près de la solution y^{n+1}.

3.3.2 Analyse *a priori* directe

L'objectif de l'analyse d'erreur *a priori* est de donner une estimation de l'erreur commise par la méthode numérique en fonction des paramètres du problème (temps d'intégration, pas de temps, champ de force). L'idée générale est de remarquer qu'à chaque pas de temps on commet une erreur d'intégration locale (erreur de troncature dans la discrétisation de l'intégrale, à laquelle s'ajoutent souvent des erreurs d'arrondi), et que ces erreurs locales s'accumulent. Le contrôle de cette accumulation demande l'introduction d'une notion de stabilité adéquate, alors que les erreurs locales sont liées à une notion de consistance. On peut montrer qu'une méthode numérique stable et consistante est convergente. Insistons sur le fait que ce type résultat, quoique courant en analyse numérique, est un pilier de l'analyse d'erreur *a priori*. Pour mesurer l'erreur, on choisit une norme sur $\mathbb{R}^d$ que l'on note $|\cdot|$; toutes les normes étant équivalentes en dimension finie et la dimension étant fixée, le choix particulier de la norme n'est pas essentiel ici.

Erreur de troncature locale

L'erreur de troncature locale est l'erreur résiduelle que l'on obtiendrait si on appliquait le schéma numérique à la solution exacte. Elle est donc définie à l'itération n comme la différence entre la solution exacte à l'itération suivante, à savoir $y(t_{n+1})$, et l'approximation numérique obtenue en partant de $y(t_n)$, à savoir $y(t_n) + \Delta t_n \Phi_{\Delta t_n}(t_n, y(t_n))$. Ainsi, l'erreur de troncature locale vaut par définition

$$\eta^{n+1} := \frac{y(t_{n+1}) - y(t_n) - \Delta t_n \Phi_{\Delta t_n}(t_n, y(t_n))}{\Delta t_n}. \tag{3.9}$$

Notons que l'erreur de troncature a la même dimension physique que la dérivée en temps de y. La normalisation que nous employons dans ce cours permet d'interpréter η comme une erreur par unité de temps.

Définition 3.3 (Consistance). *On dit qu'un méthode numérique est consistante si* $\eta^{n+1} = \mathrm{o}(1)$*, et consistante d'ordre* p *si* $\eta^{n+1} = \mathrm{O}(\Delta t_n^p)$*.*

Les preuves de consistance reposent sur des développements de Taylor de la solution exacte, et demandent donc de la régularité sur le champ de force f. On supposera toujours que le champ de force est aussi régulier que nécessaire par la suite. Notons que la régularité de la solution y découle de celle du champ de force. En effet, si f est continue, alors la solution de (3.1) est C^1. Si f est C^1, on voit alors que le membre de droite de (3.1) est C^1 (par composition) et donc que la solution y est C^2 (par intégration). On peut itérer cet argument et montrer ainsi que $y \in C^{l+1}$ si $f \in C^l$.

Exemple 3.1. Le schéma d'Euler explicite est consistant d'ordre 1. L'erreur de troncature s'écrit

$$\eta^{n+1} = \frac{y(t_n + \Delta t_n) - \Big(y(t_n) + \Delta t_n\, f(t_n, y(t_n))\Big)}{\Delta t_n}.$$

Or, par application d'une formule de Taylor avec reste exact autour de $y(t_n)$, on voit qu'il existe $\theta^n \in [0,1]$ tel que

$$y(t_n + \Delta t_n) - \Big(y(t_n) + \Delta t f(t_n, y(t_n))\Big) = \frac{\Delta t_n^2}{2} y''(t_n + \theta^n \Delta t_n). \tag{3.10}$$

Par ailleurs, la dérivée seconde $y''(t)$ peut s'exprimer en fonction des dérivées de f en dérivant (3.1) par rapport au temps, ce qui donne

$$y''(\tau) = \partial_t f(\tau, y(\tau)) + \partial_y f(\tau, y(\tau)) \cdot f(\tau, y(\tau)). \tag{3.11}$$

On voit ainsi que y'' est uniformément borné en temps sur tout intervalle de la forme $[0,T]$ ($T < +\infty$) si la fonction f et ses dérivées sont continues (la trajectoire restant bornée dans ce cas). On en déduit finalement que

$$\left|\eta^{n+1}\right| \leqslant C \Delta t_n,$$

avec une constante qui ne dépend que du temps d'intégration et de la condition initiale.

Exercice 3.1 (Calcul d'ordres de schémas numériques). *Montrer que le schéma d'Euler implicite est d'ordre 1, alors que la méthode des trapèzes, le schéma de Heun et le schéma du point milieu sont d'ordre 2.*

Stabilité

La notion de stabilité quantifie la robustesse de l'approximation numérique par rapport à des perturbations. On donne ici la définition pour des schémas à pas fixe, l'extension à des schémas à pas variables ne posant pas de problème. On fixe un intervalle de temps $[0,T]$ et un pas de temps $\Delta t > 0$ constant pour simplifier, et on note $N = T/\Delta t$ le nombre d'itérations correspondantes.

Définition 3.4 (Stabilité). *On dit qu'une méthode numérique* (3.8) *est stable s'il existe une constante $S > 0$ (qui dépend du temps d'intégration $T = N\Delta t$ mais pas de N ou de Δt tout seul) telle que, pour toute suite $z = \{z^n\}_{1\leqslant n\leqslant N}$ partant de la même condition initiale $z^0 = y^0$ et vérifiant*

$$\begin{cases} y^{n+1} = y^n + \Delta t \Phi_{\Delta t}(t_n, y^n), \\ z^{n+1} = z^n + \Delta t \Phi_{\Delta t}(t_n, z^n) + \Delta t\, \delta^{n+1}, \end{cases} \tag{3.12}$$

on ait

$$\max_{1\leqslant n\leqslant N} |y^n - z^n| \leqslant S\, \Delta t \sum_{n=1}^{N} |\delta^n|.$$

Il est utile, pour l'interprétation, de noter que l'on peut réécrire la stabilité sous une forme plus condensée, en introduisant les normes suivantes pour des suites de vecteurs $u = \{u^n\}_{n=0,\dots,N}$: une norme ℓ^∞

$$\|u\|_{\ell^\infty} = \max_{1\leqslant n\leqslant N} |u^n|\,,$$

et une norme ℓ^1 :

$$\|u\|_{\ell^1} = \Delta t \sum_{n=1}^{N} |u^n|\,.$$

La condition de stabilité peut alors se réécrire sous la forme compacte

$$\|\{y-z\}_{n=1,\dots,N}\|_{\ell^\infty} \leqslant S\|\delta\|_{\ell^1}. \tag{3.13}$$

Intéressons nous à présent à l'obtention de conditions suffisantes de stabilité. Un cas simple est celui où $\Phi_{\Delta t}$ est uniformément Lipschitzienne en y, c'est-à-dire qu'il existe $\Lambda > 0$ tel que, pour tout $y_1, y_2 \in \mathbb{R}^d$, on ait

$$|\Phi_{\Delta t}(t, y_1) - \Phi_{\Delta t}(t, y_2)| \leqslant \Lambda |y_1 - y_2|.$$

On peut alors prendre $S = \mathrm{e}^{\Lambda T}$. Cette assertion repose sur l'estimation suivante :

$$|y^{n+1} - z^{n+1}| \leqslant \Delta t|\delta^{n+1}| + (1 + \Delta t\Lambda)|y^n - z^n| \leqslant \Delta t|\delta^{n+1}| + \exp(\Lambda\Delta t)|y^n - z^n|,$$

et l'utilisation d'un lemme de Gronwall discret (voir Exercice 3.2) qui fournit l'estimation

$$|y^n - z^n| \leqslant \mathrm{e}^{\Lambda T}\left(\left|y^0 - z^0\right| + \frac{1}{\Lambda}\max_{i=1,\dots,n}|\delta^i|\right).$$

Exercice 3.2 (Inégalité de Gronwall discrète). *Montrer que si une suite $(r_n)_{n\geqslant 0}$ satisfait la relation de récurrence*

$$0 \leqslant r_{n+1} \leqslant \delta\Delta t + (1 + \Delta t\Lambda) r_n,$$

avec $\delta \geqslant 0$, $\Lambda \geqslant 0$ et $r_0 \geqslant 0$, alors on a la borne supérieure

$$0 \leqslant r_n \leqslant \mathrm{e}^{\Lambda n\Delta t}\Big(r_0 + \Lambda^{-1}\delta\Big).$$

Remarque 3.1. *Noter que les constantes de stabilité qui apparaissent croissent a priori de manière exponentielle avec le temps. En fait, Λ^{-1} peut être interprété comme un temps caractéristique τ de variation du système. La constante de stabilité peut donc être très grande si on intègre la dynamique sur des temps T bien supérieurs à τ. Pour des systèmes particuliers, tels que les systèmes linéaires dissipatifs étudiés à la section 3.4.1, on montre que les constantes de stabilité restent bornées.*

Exercice 3.3 (Analyse d'une méthode multi-pas). *On considère la méthode à 2 pas suivante (dite de Nyström) dans le cas d'un champ de vecteurs autonome et d'un pas de temps Δt fixé :*

$$y^{n+1} = y^{n-1} + 2\Delta t\, f(y^n) = \Psi(y^n, y^{n-1}),$$

avec la condition initiale y^0 fixée et la valeur y^1 obtenue par une méthode à un pas, par exemple la méthode de Heun.

(1) Calculer l'erreur de consistance η^{n+1} et déterminer l'ordre de consistance ;

(2) On va à présent discuter la stabilité sur un cas concret. On considère l'équation différentielle $\dot{y}(t) = -y(t)$ et $y^0 = 1$. Donner l'expression analytique de y^n en fonction de Δt, et montrer que la solution numérique est la somme de deux termes : un qui approche bien la solution analytique, et un terme qui diverge en temps long. Conclure.

Convergence

Une méthode numérique est convergente si l'erreur globale vérifie

$$\max_{1\leqslant n\leqslant N} |y^n - y(t_n)| \to 0$$

lorsque $y^0 = y(t_0)$ et $\Delta t = \max_{0\leqslant n\leqslant N-1} |t_{n+1} - t_n| \to 0$. On néglige dans cette section les erreurs d'arrondis dues à la représentation machine des nombres réels (ceci sera traité plus loin). On a alors le résultat suivant.

Théorème 3.1. *Une méthode stable et consistante est convergente.*

La preuve de ce résultat est très simple : on remplace z^n dans (3.12) par la solution exacte $y(t_n)$, ce qui correspond à choisir $\delta^{n+1} = \eta^{n+1}$, l'erreur de troncature locale définie en (3.9). On part en revanche de la même condition initiale. La stabilité donne donc

$$\max_{1\leqslant n\leqslant N} |y^n - y(t_n)| \leqslant S \sum_{n=1}^{N} \Delta t_n \, |\eta^n| \,. \tag{3.14}$$

Pour simplifier, considérons à présent que le pas de temps est fixe : $\Delta t_n = \Delta t$. Par définition de la consistance, on voit alors que chaque terme de la somme du membre de droite est d'ordre o(1), et qu'il y a $N = T/\Delta t$ tels termes. Ainsi, le membre de droite tend vers 0 lorsque $\Delta t \to 0$. En fait, on peut même préciser les choses : si la méthode numérique est consistante d'ordre p, alors l'erreur de troncature locale est d'ordre $\mathrm{O}(\Delta t^p)$, et on donc a une erreur globale d'ordre Δt^p :

$$\max_{1\leqslant n\leqslant N} |y^n - y(t_n)| \leqslant C\Delta t^p. \tag{3.15}$$

En fait, on peut même vérifier numériquement que l'erreur globale se comporte dans beaucoup de situations vraiment en $C\Delta t^p$ (*i.e.*, la borne supérieure sur l'erreur ne peut pas être améliorée). Des estimations similaires peuvent être obtenues avec des pas de temps variables, quitte à remplacer Δt au membre de droite par $\max_{0\leqslant n\leqslant N-1} \Delta t_n$.

Exercice 3.4 (Etude d'un schéma de Runge–Kutta). *On considère des réels $0 \leqslant \alpha, \beta, \gamma \leqslant 1$ et le schéma*

$$y^{n+1} = y^n + \Delta t_n \Psi_{\alpha,\beta,\gamma}(t_n, y^n; \Delta t_n),$$

avec

$$\Psi_{\alpha,\beta,\gamma}(t, y; \Delta t) = \alpha f(t,y) + \beta f\left(t + \frac{\Delta t}{2}, y + \frac{\Delta t}{2} f(t,y)\right) + \gamma f\big(t + \Delta t, y + \Delta t f(t,y)\big).$$

Pour simplifier, on se placera dans le cas où la dimension spatiale est 1 (i.e., $y \in \mathbb{R}$).

(1) Pour quelles valeurs de (α, β, γ) retrouve-t-on des schémas présentés dans le cours ?

(2) Discuter la consistance de la méthode en fonction des paramètres : quelles sont les relations à satisfaire pour avoir un ordre 1 ou 2 ? Peut-on avoir une méthode d'ordre 3 ou plus ?

(3) Etudier enfin la convergence lorsque f est uniformémement Lipschitzienne.

Influence des erreurs d'arrondi

On a négligé jusqu'à présent l'influence des erreurs d'arrondi liées à la représentation machine des nombres réels (voir Section 1.3). L'erreur d'arrondi globale est définie comme la différence entre la solution numérique "idéale" y^n et celle calculée en pratique, notée $\widetilde{y}^n$ dans la suite. Les écarts ont trois origines : (i) la représentation de la condition initiale $\widetilde{y}^0 = y^0 + \delta y^0$; (ii) les opérations arithmétiques nécessaires à l'évaluation à chaque pas de temps de l'incrément $\Phi(t_n, \widetilde{y}^n; \Delta t_n)$ (termes ρ_n ci-dessous) ; et (iii) les opérations arithmétiques liées au calcul de l'itéré suivant (termes σ_n). Ainsi, on peut écrire

$$\widetilde{y}^{n+1} = \widetilde{y}^n + \Delta t_n \Big(\Phi_{\Delta t_n}(t_n, \widetilde{y}^n) + \rho_n \Big) + \sigma_n.$$

On suppose que $|\rho_n| \leqslant \rho$ et $|\sigma_n| \leqslant \sigma$. Typiquement $\sigma \sim \varepsilon_{\text{machine}}$ et $\rho \sim \kappa \varepsilon_{\text{machine}}$ où κ est le maximum des conditionnements des applications $\Phi_{\Delta t_n}$. Pour une méthode stable et $N = T/\Delta t$ pas de temps (avec un pas Δt fixé), l'erreur d'arrondi globale est ainsi

$$\max_{0 \leqslant n \leqslant N} |\widetilde{y}^n - y^n| \leqslant S \left(|\delta y^0| + \sum_{n=0}^{N-1} |\sigma_n| + \Delta t \, |\rho_n| \right) = S \left(|\delta y^0| + T\rho + \frac{T\sigma}{\Delta t} \right).$$

Notons que l'erreur d'arrondi totale diverge lorsque $\Delta t \to 0$ du fait du nombre croissant d'opérations. Sa prise en compte conduit donc à limiter le nombre d'itérations réalisées, et pose donc une borne inférieure sur le pas de temps.

Avec cette information en main, on peut chercher à minimiser l'erreur numérique totale, somme de l'erreur d'approximation totale $C_T \Delta t^p$ donnée par (3.15) et de l'erreur d'arrondi totale. Pour simplifier, on va ne retenir que le terme principal de l'erreur d'arrondi, à savoir $T\sigma/\Delta t$, auquel cas on cherche à minimiser la quantité

$$C_T \Delta t^p + \frac{T\sigma}{\Delta t}.$$

Un calcul simple montre que le pas de temps optimal pour lequel l'erreur totale soit la plus petite est

$$\Delta t_{\text{opt}} = \left(\frac{T\sigma}{pC_T} \right)^{1/(p+1)}.$$

Cela correspond à un objectif de précision maximale.

3.3.3 Analyse *a priori* rétrograde

Rappelons que la philosophie de l'analyse rétrograde est la suivante : « au lieu de considérer un résultat numérique comme la solution approchée d'un problème exact, considérons-le comme la solution exacte d'un problème approché, et étudions ce problème approché ». Dans le cas présent, au lieu de chercher à estimer l'erreur entre la solution exacte $y(t_n)$ et la solution numérique y^n comme le fait l'analyse *a priori* directe, l'objectif de l'analyse *a priori* rétrograde est de considérer la solution numérique comme la solution exacte d'une EDO avec un champ de force modifié $f_{\Delta t}$ proche de f. Il s'agit donc de construire $f_{\Delta t}$ tel que la solution *exacte* de

$$\dot{z}(t) = f_{\Delta t}\big(z(t)\big), \tag{3.16}$$

soit telle que

$$y^n = z(t_n).$$

Ainsi, la dynamique modifiée coincide avec la trajectoire numérique aux temps $0, \Delta t, 2\Delta t, ...$ (mais pas forcément aux temps intermédiaires). On étudie alors les propriétés de l'EDO modifiée (3.16) pour en déduire des propriétés du schéma numérique.

On va traiter le cas d'une équation autonome (f ne dépend pas explicitement du temps) intégrée numériquement avec un pas de temps constant. Comme on cherche un champ de force modifié $f_{\Delta t}$ proche de f, on écrit

$$\dot{z} = f_{\Delta t}(z) = f(z) + \Delta t F_1(z) + \Delta t^2 F_2(z) + ..., \qquad z(0) = y^0,$$

avec des fonctions F_i (pour $i \geqslant 1$) à déterminer. On définit également le flot numérique du problème (3.1), qui est l'application

$$y^{n+1} = \Psi_{\Delta t}(y^n)$$

(*i.e.*, $\Psi_{\Delta t}(y) = y + \Delta t\, \Phi_{\Delta t}(y)$). Par exemple, pour le schéma d'Euler explicite, $\Psi_{\Delta t}(y) = y + \Delta t f(y)$ et $|\Psi_{\Delta t}(y^0) - y(\Delta t)| = \mathrm{O}(\Delta t^2)$.

Pour assurer la coincidence $y^n = z(t_n)$, il suffit d'assurer la coincidence sur un pas : si $y^0 = z(0)$ alors $y^1 = z(\Delta t)$. Le problème est donc de trouver $F_1, F_2, \ldots$ tels que

$$z(\Delta t) = \Psi_{\Delta t}(y^0).$$

On utilise pour ce faire un développement en puissances de Δt de la solution exacte de la dynamique modifiée selon

$$z(\Delta t) = z(0) + \Delta t\, \dot{z}(0) + \frac{\Delta t^2}{2}\ddot{z}(0) + ...$$

On va commencer par déterminer F_1, et pour ce faire il suffit de considérer les termes d'ordre Δt et Δt^2 dans le développement ci-dessus. On peut écrire

$$\dot{z}(0) = f(z(0)) + \Delta t F_1(z(0)) + \mathrm{O}(\Delta t^2),$$

et

$$\ddot{z}(0) = \partial_z f(z(0)) \cdot f(z(0)) + \mathrm{O}(\Delta t),$$

d'où

$$z(\Delta t) = z^0 + \Delta t\, f(z^0) + \Delta t^2 \left(F_1(z^0) + \frac{1}{2}\partial_z f(z^0) f(z^0) \right) + \mathrm{O}(\Delta t^3). \tag{3.17}$$

En choisissant

$$F_1(z) = -\frac{1}{2}\partial_z f(z) f(z) \tag{3.18}$$

on voit qu'on a donc $|\Psi_{\Delta t}(y^0) - z(\Delta t)| = \mathrm{O}(\Delta t^3)$. L'erreur de consistance pour le schéma numérique $y^{n+1} = \Psi_{\Delta t}(y^n)$, vu comme une discrétisation avec pas Δt de la dynamique continue $\dot{z} = f_{\Delta t}(z)$, est ainsi

$$\widetilde{\eta}^1 = \frac{|z(\Delta t) - \Psi_{\Delta t}(y^0)|}{\Delta t} = \mathrm{O}(\Delta t^2).$$

On peut formellement [3] itérer l'argument jusqu'à un ordre arbitraire, et montrer qu'un bon choix des perturbations au champ de force conduisent à une erreur de consistance en $\mathrm{O}(\Delta t^l)$ entre la solution numérique et la solution de la dynamique approchée, pour l arbitrairement grand ; alors que l'erreur de consistance entre la solution numérique et la solution exacte

$$\eta^1 = \frac{\left| z(\Delta t) - \left(y^0 + \Delta t\, f(y^0) \right) \right|}{\Delta t}$$

est, rappelons-le, d'ordre $\mathrm{O}(\Delta t)$.

On peut ensuite étudier les propriétés de l'équation continue $\dot{z} = f_{\Delta t}(z)$ (ou au moins de $\dot{z} = f(z) + \Delta t F_1(z)$) et en déduire des propriétés du schéma numérique pour l'EDO $\dot{y} = f(y)$, comme l'illustrent les exemples suivants.

Exercice 3.5 (Analyse de temps d'explosion). *L'objectif de cet exercice est d'étudier les temps d'explosion de l'EDO $\dot{y} = f(y) = y^2$, pour une condition initiale $0 < y^0 < 1$. Des simulations numériques sont présentées en Figure 3.1.*

(1) Calculer la solution analytique $y(t)$ lorsque $f(y) = y^2$ et montrer que cette EDO explose en un temps fini noté $T(y^0)$.

3. Ce n'est qu'un calcul formel car la série définissant $f_{\Delta t}$ n'est pas convergente en général.

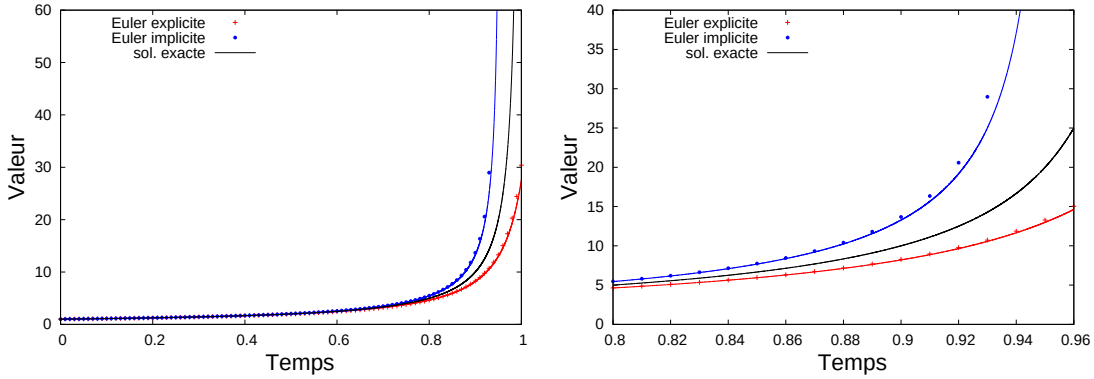

Fig. 3.1. Dynamique $\dot{y} = y^2$, intégrée avec Euler explicite ou implicite, et les dynamiques modifiées correspondantes (au premier ordre en Δt) ; les symboles indiquent le résultat des deux schémas numériques.

(2) Ecrire, dans le cas général, le développement de $z(\Delta t)$ en puissances de Δt (avec un reste d'ordre 3) en fonction de f, g et leurs dérivées évaluées en $z(0) = y^0$.

(3) On considère à présent le schéma d'Euler explicite, pour lequel $\Psi_{\Delta t}(y) = y + \Delta t\, f(y)$.

(a) Quel est l'entier p tel que $y(\Delta t) - \Psi_{\Delta t}(y^0) = \mathrm{O}(\Delta t^p)$?

(b) Déterminer g pour que la solution de l'équation modifiée $\dot{z} = g(z)$ partant de $z(0) = y^0$ soit telle que $z(\Delta t) - \Psi_{\Delta t}(y^0) = \mathrm{O}(\Delta t^{p+1})$.

(c) Montrer que, dans le cas particulier $f(y) = y^2$, on a $z(t) \leqslant y(t)$.
Comme $z(n\Delta t)$ est une bonne approximation de y^n, ceci motive le fait que le temps d'explosion prédit par le schéma d'Euler explicite $T_{\mathrm{EE}}\left(y^0\right)$ sur-estime $T\left(y^0\right)$ (quoiqu'il faut être prudent dans ce genre d'interprétation car on n'a considéré que le premier ordre dans la perturbation).

(4) Reprendre les questions précédentes pour le schéma d'Euler implicite, en montrant que le temps d'explosion est cette fois sous-estimé. On commencera par établir un développement en puissances de Δt de $\Psi_{\Delta t}$.

Exercice 3.6 (Analyse rétrograde pour les systèmes Hamiltoniens). *On considère l'approximation de la dynamique Hamiltonienne* (3.3)*, tout d'abord par des schémas numériques généraux, puis par des schémas numériques spécifiques appelés schémas d'Euler symplectiques*

$$\begin{cases} p^{n+1} = p^n - \Delta t\, \nabla V(q^n), \\ q^{n+1} = q^n + \Delta t\, p^{n+1}, \end{cases} \qquad \begin{cases} q^{n+1} = q^n + \Delta t\, p^n, \\ p^{n+1} = p^n - \Delta t\, \nabla V(q^{n+1}). \end{cases}$$

L'objectif de cet exercice est de montrer que ces schémas préservent bien l'énergie $H(q^n, p^n)$ du système. On se place en dimension 1 pour simplifier (i.e., $(q, p) \in \mathbb{R}$), et on prend une masse égale à l'unité.

(1) Déterminer la modification à l'ordre 1 du champ de force (notée F_1 dans (3.17)*) pour le schéma d'Euler explicite et le schéma d'Euler implicite. Montrer que le champ de force $f + \Delta t\, F_1$ n'est pas de la forme des champs de force apparaissant dans les dynamiques Hamiltoniennes générales* (3.4)*.*

(2) Montrer que le champ de force modifié $f + \Delta t\, F_1$ est au contraire de type Hamiltonien pour les schémas d'Euler symplectiques. En déduire qu'il existe une énergie modifiée préservée à l'ordre 2 en Δt alors que l'énergie exacte n'est préservée qu'à l'ordre 1.

3.3.4 Contrôle du pas d'intégration et analyse *a posteriori*

Nous allons à présent discuter comment obtenir une estimation d'erreur *a posteriori* (utilisant la trajectoire numérique effectivement calculée), et surtout comment s'en servir pour déterminer

de manière adaptative le pas de temps d'intégration. En particulier, on ne fixe pas le nombre de pas d'intégration *a priori*, mais plutôt un niveau d'erreur total. La discussion de cette section est très proche de ce que nous avons vu pour les méthodes de quadrature adaptatives en Section 2.2.3, ce qui n'est pas une surprise puisque les schémas numériques que nous avons décrits sont fondés sur la discrétisation de la formulation intégrale de la solution.

Fixons-nous donc une erreur totale ε. Rappelons que l'erreur d'approximation peut être bornée par (3.14) lorsque l'on part de $y^0 = y(0)$ et que l'on néglige les erreurs d'arrondi :

$$\max_{1 \leqslant n \leqslant N} |y^n - y(t_n)| \leqslant S \sum_{n=1}^{N} \Delta t_n \, |\eta^n| \, .$$

On a parfois une estimation de la constante de stabilité S. Même si ce n'est pas le cas, l'inégalité (3.14) suggère que le pas de temps Δt_n doit être choisi pour que l'erreur par unité de temps soit plus ou moins constante :

$$|\eta^n| \leqslant \frac{\kappa \varepsilon}{T}, \tag{3.19}$$

où $\kappa = 1/S$ quand S est connue, ou $1/S^*$ avec S^* une majoration prudente de S sinon. L'idée est alors de partir d'un pas de temps Δt_0 suffisamment petit, et de l'augmenter ou de le réduire pour que (3.19) soit vraie.

On a donc besoin pour cela d'une estimation *a posteriori* de l'erreur de troncature. Rappelons en effet que l'estimation *a priori* est compliquée car demande le calcul de dérivées de f (voir des expressions telles que (3.10)-(3.11)). Il y a deux approches générales pour ce faire : utiliser une même méthode numérique avec deux pas de temps différents (Δt_n et $\Delta t_n/2$), ou des méthodes de Runge–Kutta d'ordres différents et emboîtées (les calculs de force intermédiaires utilisés pour le schéma d'ordre le plus élevé sont également nécessaires pour le schéma d'ordre le plus petit). L'intérêt des méthodes emboîtées est qu'elles n'engendrent pas de surcoût de calcul.

Traitons un cas particulier simple pour illustrer la méthode : le schéma d'Euler explicite, pour lequel on peut obtenir une estimation de l'erreur locale sans recourir à une méthode avec un pas $\Delta t/2$. Plus précisément, l'estimation d'erreur *a posteriori* est obtenue via la différence des forces (voir (3.10)-(3.11)) :

$$\begin{aligned} f(t_{n+1}, y^{n+1}) - f(t_n, y^n) &= \Delta t_n \Big(\partial_t f(t_n, y^n) + \partial_y f(t_n, y^n) \cdot f(t_n, y^n) \Big) + \mathrm{O}(\Delta t_n^2) \\ &= 2\, \eta^{n+1} + \mathrm{O}(\Delta t_n^2). \end{aligned}$$

Ceci montre que

$$|\eta^{n+1}| \simeq \frac{\big| f(t_{n+1}, y^{n+1}) - f(t_n, y^n) \big|}{2}.$$

Si la condition (3.19) n'est pas satisfaite, on diminue le pas de temps Δt_n jusqu'à ce que ce soit le cas (par exemple, par un facteur 0.8) ; si cette condition est satisfaite avec une bonne marge de sécurité, on peut songer à augmenter un peu le pas de temps pour la prochaine itération (par exemple, en le multipliant par 1.25). En pratique, on fixe un intervalle de valeurs admissibles $[\Delta t_{\min}, \Delta t_{\max}]$ pour le pas de temps (la valeur de $\Delta t_{\min}$ étant fixée par les limitations en temps de calcul et l'accroissement des erreurs d'arrondi, voir la discussion de la Section 3.3.2), et on arrête la simulation si Δt passe sous $\Delta t_{\min}$, ce qui est le signe d'une singularité du champ de force.

3.4 Etude en temps long de systèmes particuliers

3.4.1 Systèmes dissipatifs

On va dans cette section considérer des systèmes linéaires dissipatifs unidimensionnels, de la forme générale : $\lambda \in \mathbb{C}$ étant donné,

$$\dot{y}(t) = \lambda y(t), \qquad \mathrm{Re}(\lambda) < 0. \tag{3.20}$$

Les équations linéaires dissipatives peuvent être vues comme une version linéarisée de problèmes plus intéressants. Elles sont dans tous les cas un bon cadre pour bien comprendre des notions de stabilité de manière analytique, avant d'extrapoler les résultats à des systèmes nonlinéaires plus compliqués.

La notion de stabilité pertinente pour (3.20) est celle de stabilité absolue, qui reproduit le comportement qualitatif asymptotique de la solution continue : $y(t) \to 0$ lorsque $t \to +\infty$. On souhaite donc que la trajectoire numérique $y^n \in \mathbb{R}$ soit telle que $y^n \to 0$ lorsque $n \to +\infty$.

Méthodes numériques générales

Comme le problème (3.20) est linéaire, on peut écrire une itération d'une méthode numérique à un pas sous la forme

$$y^{n+1} = R(\lambda \Delta t) y^n,$$

le nombre $R(\lambda)$ dépendant de la méthode numérique choisie. On a par exemple $R(z) = 1 + z$ pour la méthode d'Euler explicite, $R(z) = (1 - z)^{-1}$ pour la méthode d'Euler implicite, et

$$R(z) = \frac{1 + z/2}{1 - z/2}$$

pour la méthode des trapèzes (aussi appelé schéma de Crank-Nicholson).

On définit la région de stabilité absolue d'une méthode numérique comme l'ensemble suivant :

$$\mathcal{A} = \{z \in \mathbb{C},\ |R(z)| < 1\}.$$

Un schéma est dit absolument stable si $\{z \in \mathbb{C},\ \mathrm{Re}(z) < 0\} \subset \mathcal{A}$; sinon il est conditionnellement absolument stable. On vérifie facilement que le schéma d'Euler explicite est conditionnellement stable (sous la condition $|1 + z| < 1$ *i.e.* $z \in B(-1, 1)$), alors que le schéma d'Euler implicite est inconditionnellement stable.

Exercice 3.7 (Stabilité absolue). *Etudier la stabilité absolue du schéma du point-milieu, du schéma de Heun, et du θ-schéma défini, pour $\theta \in [0, 1]$, par*

$$y^{n+1} = y^n + \Delta t\Big((1 - \theta) f(y^n) + \theta f(y^{n+1})\Big),$$

dans le cas où $f(y) = \lambda y$ avec $\mathrm{Re}(\lambda) < 0$. Conclure quant à la relation entre caractère implicite d'une méthode et stabilité absolue inconditionnelle.

Application à un problème modèle

Présentons à présent une application des méthodes précédente aux problèmes raides, typiquement rencontrés en cinétique chimique par exemple (voir la discussion à ce sujet en Section 3.1). C'est, rappelons-le, un cas où plusieurs échelles de temps cohabitent dans le système. On va considérer un cas très simple, analytiquement soluble, et qui permettra donc de tester la pertinence et la qualité des méthodes numériques (il est toujours bon de valider sa méthode sur des cas simples que l'on peut traiter complètement, avant de se tourner vers un cas plus compliqué pour lequel il peut ne pas exister de calcul de référence auquel se comparer).

On peut penser à l'inconnue y comme à la concentration d'espèces chimiques. Dans le cas simple de deux espèces, et pour un paramètre $\mu \gg 1$ donné, on suppose que l'état du système évolue selon la dynamique suivante :

$$\dot{y} = My, \qquad M = \begin{bmatrix} -1 & 0 \\ 0 & -\mu \end{bmatrix}, \qquad y(0) = \begin{pmatrix} y_1^0 \\ y_2^0 \end{pmatrix}.$$

Ce problème représente par exemple la décroissance de deux isotopes radioactifs au cours du temps. On peut calculer analytiquement la solution exacte du problème dans ce cas : $y(t)^T =$

$(y_1^0\,\mathrm{e}^{-t},\, y_2^0\mathrm{e}^{-\mu t})$. On voit bien l'existence de deux échelles de temps très différentes, une d'ordre 1, et une d'ordre $1/\mu$. Disons que l'on cherche à bien résoudre l'évolution de la première composante, tout en conservant une dynamique stable pour la seconde.

Le schéma d'Euler explicite s'écrit

$$y^{n+1} = \Big(\mathrm{Id} + \Delta t\, M\Big) y^n.$$

L'exigence de précision pour la première composante au moins demande que $(1-\Delta t)^n \simeq \mathrm{e}^{-n\Delta t}$. D'un autre côté la condition de stabilité est $|1-\mu\Delta t| < 1$, soit $\Delta t < 2/\mu$. On voit donc bien que c'est cette seconde condition qui limite sévèrement le pas de temps. Cette conclusion est vraie également dans des problèmes moins simplistes, par exemple l'équation de la chaleur discrétisée par éléments finis ou différences finies.

Une manière de lever cette limitation consiste à utiliser des méthodes implicites, qui sont certes plus coûteuses à mettre en oeuvre, mais qui sont inconditionnellement stables. Dans ce cas, c'est donc l'objectif de précision (sur la première composante ici) qui peut primer et déterminer le pas de temps.

3.4.2 Systèmes Hamiltoniens

Nous discutons dans cette section des méthodes numériques appropriées pour l'intégration en temps long de systèmes Hamiltoniens comme ceux rencontrés en physique statistique numérique ou en dynamique céleste (voir Section 3.1.1). Le lecteur désirant plus de détails se reportera au livre très complet [5], ou plus raisonnablement à l'article introductif [4].

Nous commençons par établir quelques propriétés de la dynamique continue, et nous tournons ensuite vers son approximation numérique. Nous verrons que les méthodes numériques génériques (applicables à toute EDO, même non-Hamiltonienne) donnent de mauvais résultats, et que la conception de méthodes reproduisant de manière qualitative les propriétés de la dynamique Hamiltonienne demande un certain soin.

Etude de la dynamique continue

Rappelons que la dynamique Hamiltonienne s'écrit (voir (3.3))

$$\begin{cases} \dot q(t) = \quad \dfrac{\partial H}{\partial p} = M^{-1}p(t), \\ \dot p(t) = -\dfrac{\partial H}{\partial q} = -\nabla_q V(q(t)). \end{cases} \tag{3.21}$$

Posant $y = (q,p) \in \mathbb{R}^{2dN}$ (pour N particules en dimension d), on peut réécrire cette dynamique sous la forme d'une équation différentielle ordinaire :

$$\frac{dy}{dt} = J \cdot \nabla H(y) = J \cdot \begin{pmatrix} \nabla_q H(q,p) \\ \nabla_p H(q,p) \end{pmatrix}, \tag{3.22}$$

où J est la matrice réelle carrée de taille $2dN$

$$J = \begin{pmatrix} 0 & I_{dN} \\ -I_{dN} & 0 \end{pmatrix}.$$

On remarque que J est antisymétrique et orthogonale (*i.e.* $J^T = -J = J^{-1}$), et donc de déterminant égal à 1.

La technique de choix pour assurer l'existence et l'unicité d'une solution globale de la dynamique Hamiltonienne est d'utiliser $W = H + c$ comme fonction de Lyapounov dans le cas où le potentiel V est borné inférieurement et va à l'infini à l'infini (c étant une constante assurant que $W \geqslant 1$). Notons en effet qu'un calcul simple montre que

$$\frac{dH(q(t),p(t))}{dt} = \partial_q H(q(t),p(t)) \cdot \dot{q}(t) + \partial_p H(q(t),p(t)) \cdot \dot{p}(t) = 0,$$

et on a donc préservation de l'énergie le long des trajectoires : $H(q(t),p(t)) = H(q(0),p(0))$. Nous supposerons par la suite que (3.21) a une unique solution globale pour toute condition initiale. La préservation de l'énergie permet également de définir une notion de stabilité appropriée pour une intégration en temps potentiellement infini par le biais de la conservation de l'énergie : on souhaite que de petites perturbations de la condition initiale (et potentiellement des petites perturbations Hamiltoniennes de la dynamique) conduisent à des niveaux d'énergies proches de ceux de la dynamique non-perturbée.

Une application utile pour la suite est le flot ϕ_t, qui associe à une condition initiale q_0, p_0 l'unique solution au temps $t \geqslant 0$: $\phi_t(q_0,p_0) = (q(t),p(t))$. Notons que l'unicité et l'existence de la trajectoire globale permet de définir ϕ_{-t} comme l'inverse de ϕ_t ($t \geqslant 0$). La propriété

$$\phi_{-t} = (\phi_t)^{-1}$$

est appelée symétrie. Une autre manière de définir le flot pour des temps négatifs est d'utiliser la relation

$$\phi_{-t} = S \circ \phi_t \circ S, \qquad S(q,p) = (q,-p).$$

Cela signifie que l'on peut remonter à l'antécédent en temps $-t \leqslant 0$ en changeant le sens des vitesses, intégrant en temps positif, et changeant à nouveau le sens des vitesses. Cette propriété est connue sous le nom de réversibilité en temps.

Une autre propriété importante de la dynamique Hamiltonienne est la symplecticité. Notant $g'(q,p)$ la matrice jacobienne d'une application $g(q,p) = (g_1(q,p),\dots,g_{2dN}(q,p))^T$:

$$g'(q,p) = \begin{pmatrix} \frac{\partial g_1}{\partial q_1} & \cdots & \frac{\partial g_1}{\partial q_{dN}} & \frac{\partial g_1}{\partial p_1} & \cdots & \frac{\partial g_1}{\partial p_{dN}} \\ & & \ddots & & & \\ & & & \ddots & & \\ \frac{\partial g_{2dN}}{\partial q_1} & \cdots & \frac{\partial g_1}{\partial q_{dN}} & \frac{\partial g_1}{\partial p_1} & \cdots & \frac{\partial g_{2dN}}{\partial p_{dN}} \end{pmatrix}.$$

la propriété de symplecticité s'écrit

$$[\phi_t'(q,p)]^T \cdot J \cdot \phi_t'(q,p) = J, \qquad J = \begin{pmatrix} 0 & I_{dN} \\ -I_{dN} & 0 \end{pmatrix}. \tag{3.23}$$

On a le résultat suivant.

Théorème 3.2 (Symplecticité du flot hamiltonien). *Soit $H(q,p)$ une fonction de $\mathrm{C}^2(U)$ ($U \subset \mathbb{R}^{2dN}$ ouvert). Alors, pour tout t fixé tel que ϕ_t existe, le flot ϕ_t est une transformation symplectique.*

Preuve. Soit le flot ϕ_t de (3.22). On remarque tout d'abord que

$$\frac{d}{dt}\left(\frac{\partial \phi_t(y)}{\partial y}\right) = \frac{\partial}{\partial y}\left(\frac{d\phi_t(y)}{dt}\right) = \frac{\partial}{\partial y}\big(J \cdot \nabla H(\phi_t(y))\big) = J \cdot \nabla^2 H(\phi_t(y)) \cdot \frac{\partial \phi_t(y)}{\partial y}.$$

Ainsi, notant $\psi : t \mapsto \dfrac{\partial \phi_t(y)}{\partial y}$, on a

$$\frac{d}{dt}\left(\psi(t)^T \cdot J \cdot \psi(t)\right) = \psi(t)^T \cdot \nabla^2 H(\phi_t(y)) \cdot J^T J \cdot \psi(t) + \psi(t)^T \cdot \nabla^2 H(\phi_t(y)) \cdot J^2 \cdot \psi(t) = 0,$$

en utilisant $J^T = -J$. Ceci montre que $\psi(y)^T \cdot J \cdot \psi(y) = \psi(0)^T \cdot J \cdot \psi(0) = J$, ce qui permet de conclure à la symplecticité. □

En fait, on peut même montrer qu'une EDO est (localement) Hamiltonienne si et seulement si son flot est symplectique. La symplecticité permet d'obtenir directement la propriété de conservation du volume dans l'espace des phases. En effet, si g est une application symplectique, le déterminant de sa matrice jacobienne vaut ± 1 d'après (3.23).

Intégration numérique : échec des méthodes génériques

Commençons par regarder ce que donnent des méthodes numériques génériques pour l'intégration en temps long de la dynamique Hamiltonienne. On a bien sûr des propriétés de convergence trajectorielle en temps court, qui sont données par la théorie générale, mais les constantes qui apparaissent dans les estimations correspondantes (notamment la constante de stabilité) dépendent typiquement de manière exponentielle du temps, et la question de savoir si l'énergie du système est bien préservée en temps long pour un pas de temps fini n'a pas de rapport *a priori* avec la convergence trajectorielle en temps fini (qui est un résultat valable dans la limite $\Delta t \to 0$).

Un cas analytique simple, l'oscillateur harmonique unidimensionnel de masse $m = 1$ (potentiel $V(q) = \omega^2 q^2/2$) permet déjà de dégager un comportement qui se vérifie numériquement pour des systèmes plus généraux (potentiels nonlinéaires et dimension supérieure). Pour ce système très simple, l'état $y^n \in \mathbb{R}^2$ satisfait une relation de récurrence linéaire de la forme

$$y^{n+1} = A(\Delta t) y^n, \tag{3.24}$$

la matrice $A(\Delta t)$ dépendant de la méthode numérique utilisée. Pour les schémas d'Euler explicite et implicite, on a respectivement

$$A_{\mathrm{EE}}(\Delta t) = \begin{pmatrix} 1 & \Delta t \\ -\omega^2 \Delta t & 1 \end{pmatrix}, \qquad A_{\mathrm{EI}}(\Delta t) = \frac{1}{1+\omega^2 \Delta t^2} \begin{pmatrix} 1 & \Delta t \\ -\omega^2 \Delta t & 1 \end{pmatrix}.$$

Les valeurs propres de ces matrices sont respectivement $|\lambda_\pm^{\mathrm{EE}}| = |1 \pm \mathrm{i}\omega\Delta t| > 1$ et $|\lambda_\pm^{\mathrm{EI}}| = |(1 \pm \mathrm{i}\omega\Delta t)^{-1}| < 1$. On voit alors que l'énergie augmente exponentiellement dans le cas du schéma d'Euler explicite, et décroît exponentiellement pour le schéma d'Euler implicite. La situation n'est pas forcément meilleure pour des schémas d'ordre plus élevé, tel que le fameux schéma de Runge-Kutta d'ordre 4. Dans ce cas, on peut montrer que les valeurs propres sont

$$|\lambda_\pm^{\mathrm{RK4}}| = \sqrt{1 - \frac{(\omega\Delta t)^6}{72} + \frac{(\omega\Delta t)^8}{576}},$$

qui est bien strictement plus petit que 1 pour $\omega\Delta t < \mu^* \simeq 2,828427$. Dans les systèmes simulés avec cet algorithme, on observe donc une lente mais sûre décroissance de l'énergie en temps long pour des pas de temps assez petits.

Méthodes numériques symplectiques

Au vu des résultats de la section précédente, une question naturelle est : quelle sont les propriétés de la dynamique Hamiltonienne qui doivent être préservées par les schémas numériques afin que l'intégration en temps long soit stable (au sens où l'énergie est bien préservée) ?

On montre en fait, en utilisant l'analyse d'erreur rétrograde, que c'est la notion de symplecticité qui est pertinente : en gros, les schémas symplectiques préservent exactement une énergie approchée, ce qui les conduit à préserver approximativement l'énergie du système sur des temps longs.

Définition 3.5 (Symplecticité d'un schéma numérique). *Une méthode numérique à un pas est symplectique si l'application*

$$\Psi_{\Delta t} \ : \ y^0 \mapsto y^1 = \Psi_{\Delta t}(y^0)$$

est symplectique lorsque la méthode est appliquée à un système Hamiltonien régulier.

La notion de symplecticité est liée à la conservation du volume dans l'espace des phases. Ceci explique que pour une méthode symplectique, les trajectoires ne peuvent pas converger vers une trajectoire donnée (ce qui impliquerait une diminution du volume dans l'espace des phases, ce qui est le cas du schéma d'Euler implicite et de Runge-Kutta d'ordre 4 pour Δt assez petit), ou au contraire diverger (ce qui impliquerait une augmentation du volume dans l'espace des phases, ce qui est le cas du schéma d'Euler explicite). Cette préservation du volume explique heuristiquement la très bonne stabilité des schémas symplectiques, précisée par le Théorème 8.1 de [5] :

Théorème 3.3 (Conservation approchée de l'énergie en temps long pour des méthodes symplectiques). *On considère un Hamiltonien $H : \mathbb{R}^{2dN} \to \mathbb{R}$ analytique et une méthode numérique $\Psi_{\Delta t}$ symplectique, d'ordre p. Si la solution numérique reste dans un compact $K \subset \mathbb{R}^{2dN}$, alors il existe $h > 0$ tel que*

$$H(y^n) = H(y^0) + \mathrm{O}(\Delta t^p)$$

pour des temps exponentiellement longs $n\Delta t \leqslant \mathrm{e}^{h/\Delta t}$.

Ce résultat est d'énoncé technique, et pose de fortes contraintes sur le système. On a toutefois des résultats analogues sous des hypothèses un peu plus faibles. Retenons cependant que ce genre de résultat mathématique est peut-être sous-optimal, au sens où on n'a jamais observé en pratique une dérive de l'énergie pour un schéma symplectique !

On peut se demander également si les schémas numériques jouissent d'autres propriétés du flot Hamiltonien.

Définition 3.6 (Réversibilité en temps d'un schéma numérique). *Définissant l'opérateur de réversibilité en temps S par $S(q,p) = (q,-p)$, on dit qu'un schéma numérique est réversible en temps si*

$$S \circ \Psi_{\Delta t} = \Psi_{-\Delta t} \circ S.$$

Cette propriété est satisfaite pour la grande majorité des schémas numériques associés à (3.21), et en tout cas, tous les schémas que nous allons rencontrer satisfont cette propriété, et ne s'avère donc pas être une propriété discriminante. Ceci n'est pas le cas pour la symétrie en temps :

Définition 3.7 (Symétrie d'un schéma numérique). *On dit qu'un schéma numérique est symétrique lorsque*

$$\Psi_{\Delta t} \circ \Psi_{-\Delta t} = \mathrm{Id}.$$

Exercice 3.8 (Propriétés des schémas numériques pour la dynamique Hamiltonienne). *Vérifier (par des calculs explicites) si les schémas d'Euler explicite et symplectiques sont réversibles, symétriques, symplectiques. On considèrera, pour simplifier, un système Hamiltonien avec $q, p \in \mathbb{R}$, de masse 1.*

Construction de schémas symplectiques par méthode de décomposition

Une manière aisée de construire des méthodes symplectiques est d'utiliser une méthode de décomposition. Pour une EDO générale

$$\dot{y} = f(y) = f_1(y) + f_2(y),$$

une méthode de décomposition consiste à proposer une nouvelle configuration y^{n+1} en utilisant d'abord une approximation de $\dot{y} = f_1(y)$ (pour un pas de temps Δt), puis une approximation de $\dot{y} = f_2(y)$ (aussi sur un pas de temps Δt). Le choix de la décomposition $f = f_1 + f_2$ est motivé par le fait que les dynamiques élémentaires soient faciles à intégrer, voire soient analytiquement intégrables.

Dans le cas Hamiltonien, on peut écrire

$$H(q,p) = H_1(q,p) + H_2(q,p), \qquad H_1(q,p) = \frac{1}{2} p^T M^{-1} p, \qquad H_2(q,p) = V(q),$$

et décomposer le champ de force selon

$$\begin{cases} \dot{q} = M^{-1}\, p, \\ \dot{p} = 0, \end{cases} \qquad \begin{cases} \dot{q} = 0, \\ \dot{p} = -\nabla V(q). \end{cases}$$

Chacun des deux systèmes Hamiltoniens ci-dessous a un flot que l'on peut calculer exactement. Ce sont respectivement

$$\phi_t^1(q,p) = (q + t\,M^{-1}p, p), \quad \phi_t^2(q,p) = (q, p - t\nabla V(q)).$$

On utilise alors une formule de Lie-Trotter. [4] Une approximation numérique de (3.21) obtenue par cette méthode de décomposition est par exemple

$$(q^{n+1}, p^{n+1}) = \phi^2_{\Delta t} \circ \phi^1_{\Delta t}(q^n, p^n).$$

Les applications ϕ_t^2, ϕ_t^1 sont symplectiques pour tout temps t car ce sont les flots de dynamiques hamiltoniennes. Par ailleurs, on vérifie facilement par une formule de dérivation des applications composées que la composition de deux applications symplectiques est encore symplectique. Plus explicitement, on obtient le schéma dit "Euler symplectique A"

$$\begin{cases} q^{n+1} = q^n + \Delta t M^{-1}\, p^n, \\ p^{n+1} = p^n - \Delta t \nabla V(q^{n+1}). \end{cases} \tag{3.25}$$

Si on compose les flots dans l'autre sens, on obtient encore un schéma symplectique, dit "Euler symplectique B"

$$\begin{cases} q^{n+1} = q^n + \Delta t M^{-1}\, p^{n+1}, \\ p^{n+1} = p^n - \Delta t \nabla V(q^n). \end{cases} \tag{3.26}$$

Ces deux schémas sont tous deux explicites et d'ordre 1 (on vérifie que $\phi^1_{\Delta t} \circ \phi^2_{\Delta t} = \phi_{\Delta t} + \mathrm{O}(\Delta t^2)$), réversibles mais pas symétriques.

Pour l'étude de la stabilité linéaire, on considère à nouveau l'oscillateur harmonique avec $m = 1$. Dans ce cas particulier, on a la formulation matricielle (3.24) avec respectivement

$$A_{\text{Euler A}}(\Delta t) = \begin{pmatrix} 1 & \Delta t \\ -\omega^2 \Delta t & 1 - (\omega \Delta t)^2 \end{pmatrix}, \quad A_{\text{Euler B}}(\Delta t) = \begin{pmatrix} 1 - (\omega \Delta t)^2 & \Delta t \\ -\omega^2 \Delta t & 1 \end{pmatrix}. \tag{3.27}$$

Les valeurs propres de ces matrices sont

$$\lambda_\pm = 1 - \frac{(\omega \Delta t)^2}{2} \pm \mathrm{i}\omega \Delta t \sqrt{1 - \frac{(\omega \Delta t)^2}{4}}.$$

On a donc bien $|\lambda_\pm| = 1$ à condition que $\omega \Delta t < 2$. Ceci montre que les schémas d'Euler symplectiques sont conditionellement linéairement stables en temps long. Comme les valeurs propres sont de module 1, il n'y a pas de dérive de l'énergie dans un sens ou dans l'autre.

Remarque 3.2. *Insistons sur l'importance de la condition de stabilité linéaire en temps long $\omega \Delta t < 2$. Cette condition montre que s'il existe des fréquences hautes dans le système, le pas de temps d'intégration devra être très petit ! La très bonne stabilité des schémas et la conservation de l'énergie sur des temps longs pose donc une contrainte sur le pas de temps maximal que l'on peut utiliser...*

Schéma de Störmer-Verlet.

Le schéma de Störmer-Verlet est le schéma le plus utilisé en pratique pour l'intégration des équations (3.21). Il a été découvert et redécouvert plusieurs fois au cours du temps, en particulier par Störmer (1907) en astronomie, et par Verlet (1967) en dynamique moléculaire. Mais, comme l'a remarqué Feynman, on en trouve déjà des traces chez Newton (1687) ! Une très bonne introduction sur l'historique de ce schéma est donnée dans [4].

La procédure conduisant à l'obtention des schémas d'Euler symplectiques est quelque peu arbitraire, dans la mesure où il faut décider d'un certain ordre dans la composition. On peut symétriser cette opération par la composition

$$\Psi_{\Delta t} = \phi^2_{\Delta t/2} \circ \phi^1_{\Delta t} \circ \phi^2_{\Delta t/2},$$

4. Avec des mots simples (ou pour ceux qui ont joué au serpent sur leur portable) : pour faire un pas en diagonale, on fait un pas sur le côté et un pas tout droit... C'est l'exemple le plus simple de décomposition.

qui conduit au schéma de Störmer-Verlet :

$$\begin{cases} p^{n+1/2} = p^n - \dfrac{\Delta t}{2}\nabla V(q^n), \\ q^{n+1} = q^n + \Delta t M^{-1} p^{n+1/2}, \\ p^{n+1} = p^{n+1/2} - \dfrac{\Delta t}{2}\nabla V(q^{n+1}). \end{cases}$$

On vérifie que ce schéma est d'ordre 2 (on a $\phi^1_{\Delta t/2}\circ\phi^2_{\Delta t}\circ\phi^1_{\Delta t} = \phi_{\Delta t}+\mathrm{O}(\Delta t^3)$) et qu'il est symétrique, alors que les schémas d'Euler symplectiques A et B ne le sont pas. L'intérêt de la symétrisation est qu'on peut gagner un ordre de précision, en n'augmentant pas le nombre d'évaluations totales de la force (quitte à stocker les forces calculées au pas précédent pour la première mise à jour des vitesses sur un demi-pas de temps).

Concernant la stabilité linéaire en temps long, on a, toujours dans le cas de l'oscillateur harmonique avec des particules de masse $m = 1$, la formulation matricielle (3.24) avec

$$A(\Delta t) = A_{\text{Euler A}}(\Delta t/2)A_{\text{Euler B}}(\Delta t/2) = \begin{pmatrix} 1 - \dfrac{(\omega\Delta t)^2}{2} & \Delta t \\ -\omega^2\Delta t\left(1 - \dfrac{(\omega\Delta t)^2}{4}\right) & 1 - \dfrac{(\omega\Delta t)^2}{2} \end{pmatrix}, \qquad (3.28)$$

où $A_{\text{Euler A}}, A_{\text{Euler B}}$ sont données par (3.27). On montre facilement que les valeurs propres de $A(\Delta t)$ sont toutes deux de module 1 si et seulement si $\omega\Delta t < 2$.

Remarque 3.3 (Formulation sans les impulsions). *Un calcul simple montre que la solution de* (3.28) *vérifie*

$$M\frac{q^{n+1} - 2q^n + q^{n-1}}{\Delta t^2} = -\nabla V(q^n).$$

On reconnaît ainsi une dicrétisation par différences finies centrées de l'équation du mouvement

$$M\ddot{q} = -\nabla V(q).$$

Finalement, le schéma de Störmer-Verlet peut être construit très simplement ! En revanche, cette construction simple ne permet pas de mettre en évidence les propriétés structurelles profondes du schéma numérique, et il serait difficile de montrer que l'énergie est préservée de manière approchée sur des temps longs avec cette manière de faire.

Conservation exacte d'une énergie approchée : un exemple.

Dans le cas de l'oscillateur harmonique, on peut préciser la bonne conservation de l'énergie pour le schéma de Verlet. On cherche une matrice $C(\Delta t) \in \mathbb{R}^{2\times 2}$ diagonale telle que

$$\forall y^0 \in \mathbb{R}^2, \quad (y^n)^T C(\Delta t) y^n = (y^0)^T C(\Delta t) y^0,$$

c'est-à-dire $C(\Delta t)$ diagonale telle que

$$A(\Delta t)^T C(\Delta t) A(\Delta t) = C(\Delta t).$$

Par exemple,

$$C(\Delta t) = \begin{pmatrix} \omega^2\left(1 - \dfrac{(\omega\Delta t)^2}{4}\right) & 0 \\ 0 & 1 \end{pmatrix}$$

convient. Ceci montre que

$$\forall n \geqslant 0, \quad H_{\Delta t}(q^n, p^n) = H_{\Delta t}(q^0, p^0),$$

avec le Hamiltonien modifié $H_{\Delta t}(q,p) = H(q,p) - (\omega\Delta t)^2 q^2/4$. Ainsi, on a bien conservation exacte d'une énergie approchée à $\mathrm{O}(\Delta t^2)$ près, ce qui conduit au final à une conservation de l'énergie exacte à $\mathrm{O}(\Delta t^2)$ pour tout temps.

Conservation exacte d'une énergie approchée : cas général

La bonne conservation de l'énergie peut se montrer en utilisant l'analyse *a priori* rétrograde. Plus précisément, on montre que le champ de force modifié est (localement) Hamiltonien si et seulement si le schéma est symplectique. Ceci explique pourquoi les schémas symplectiques préservent bien l'énergie en temps long : c'est parce qu'ils préservent de manière exacte une énergie approchée

$$H_{\Delta t} = H(q,p) + \Delta t^{p+1}\, H_{p+1}(q,p) + \dots$$

où p est l'ordre de la méthode. En fait, le développement de l'énergie en puissances de Δt n'est pas convergent, et il faut recourir à des constructions techniques pour rendre les choses rigoureuses. Le lecteur (très motivé, car le sujet est technique !) pourra se plonger avec délices dans les chapitres IX et X de [5]. On parle d'*intégration géométrique*, c'est-à-dire d'une intégration numérique qui préserve bien les propriétés géométriques/structurelles fondamentales, ici, la surface d'énergie constante.

A titre d'illustration, vous pouvez calculer H_2 pour le schéma d'Euler symplectique (A ou B), ainsi que suggéré par l'Exercice 3.6.

Bibliographie

[1] R. Caflisch, Monte Carlo and quasi-Monte Carlo methods, *Acta Numerica* **7** (1998) 1–49.

[2] J.-P. Demailly, *Analyse numérique et équations différentielles*, Collection Grenoble Sciences (EDP Sciences, 2006).

[3] D. Goldberg, What every computer scientist should know about floating-point arithmetic, *ACM Computing Surveys* **23**(1) (1991).

[4] E. Hairer, C. Lubich, et G. Wanner, Geometric numerical integration illustrated by the Störmer-Verlet method, *Acta Numerica* **12** (2003) 399–450.

[5] E. Hairer, C. Lubich, et G. Wanner, *Geometric Numerical Integration : Structure-Preserving Algorithms for Ordinary Differential Equations*, volume 31 of *Springer Series in Computational Mathematics* (Springer-Verlag, 2006).

[6] B. D. Mccullough et H. D. Vinod, The numerical reliability of econometric software, *Journal of Economic Literature* **37** (1999) 633–665.

[7] B. Nuseibeh, Ariane 5 : Who dunnit ?, *IEEE Software* **14**(3) (1997) 15–16.

[8] A. Quarteroni, R. Sacco, et F. Saleri, *Méthodes numériques. Algorithmes, analyse et applications* (Springer, 2007).

[9] R. D. Skeel, Roundoff error and the Patriot missile, *SIAM News* **25**(4) (1992).