REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
UNIVERSITE DE JIJEL

Faculté des Sciences Exactes et Informatique
Département de Mathématiques

Mémoire

Pour l'obtention du diplôme de : **Master**

Spécialité : Mathématiques Appliquées

Option : EDP et applications

Thème

Résolution numérique des équations différentielles ordinnaires

Présenté par :

Boulahyoune Nihad

Devant le jury :

Président :	I. Kecis	MC B. Université de Jijel
Encadreur :	Y. Daikh	MC B. Université de Jijel
Examinateur :	I. Touil	MA A. Université de Jijel
Examinateur :	S. Djemai	MA B. Université de Jijel

Promotion 2016/2017

Remerciements

En préambule à ce mémoire nous remercions ALLAH qui nous aide et nous donne la patience et le courage durant ces langues années d'étude.

$\mathcal{J}$e souhaite adresser mes remerciements les plus sincères aux personnes qui m'ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire et particulièrement mon encadreur **Y. Daikh**, d'avoir voulu proposer et assurer la direction de ce mémoire, pour sa confiance et ses conseils judicieux et sa totale disponibilité.

$\mathcal{J}$e tiens à remercier sincèrement les membres de jury qui ont accepté de jurer mon travail. **J**e n'oublie pas mes parents, les membres de ma familles pour leur contribution, leur soutien et leur patience.

Enfin, J'adresse mes plus sincères remerciements à tous mes proches et amis, qui m'ont toujours encouragé au cours de la réalisation de ce mémoire.

Dédicace

$\mathcal{J}$e dédie ce travaille de fin d'études

$\mathcal{A}$ mes chèrs parents $\star$ Saliha et Housin $\star$ pour leurs soutien moraux

et pour leurs encouragements ...

Que Dieu vous protège

$\mathcal{A}$ ma soeur Maryama

$\mathcal{A}$ mon frère, Youcef et Mouhamed

$\grave{\mathcal{A}}$ mes chères Dounia, Razika, Amina et Wahiba

$\mathcal{A}$ mes amis de ma promotion

$\mathcal{A}$ tous mes enseignants depuis le primaire jusqu'à maintenant

Enfin, je dédie ce mémoire à ceux qui m'aiment et surtout ceux que j'aime

$\mathcal{N}$ihad.

Table des matières

Table des figures

Introduction

Les équations différentielles interviennent dans de nombreux domaines comme la mécanique, l'astronomie, la biologie, la médecine, etc. Généralement, on ne peut pas calculer la solution de ces équations de manière exacte. Il est alors nécessaire d'utiliser des méthodes numériques qui donneront des bonnes approximations des solutions pour un coût de calcul qui ne soit pas trop important. L'élaboration de techniques de résolution approchée des équations différentielles ordinaires constitue un vaste domaine d'études et de recherches depuis plus de trois siècle. Il existe un grand nombre de façons de résoudre une équation différentielle et aucune méthode n'est clairement supérieure à toutes les autres. La première méthode numérique fut introduite en 1768 par Leonhard Euler. Depuis, un grand nombre de techniques ont été développées : elles se basent sur la discrétisation de l'intervalle d'étude en un certain nombre de pas, suivant le type de formule utilisé pour approcher les solutions.
Après un bref rappel des notions de base sur les EDO, et des résultats fondammentaux d'existence et d'unicité de la solution d'une EDO, nous introduisons dans les chapitres 2 et 3 deux classes de méthodes les plus couramment utilisées pour l'approximation numérique des EDO qui sont les schémas à un pas et les schémas multi-pas, explicites ou implicites tout en présentant les concepts de consistance, de stabilité, de convergence et d'ordre qui permettent l'analyse théorique de ces méthodes. On présente aussi les méthodes de prédiction correction dans l'objectif de savoir comment régler le carractère implicite des méthodes numériques tout en minimisant le coût de calcul. On termine par donner un exemple d'application dans le domaine des EDP. On a choisi le problème de Dirichlet avec condition initiale de Cauchy pour l'équation de la chaleur, la discrétisation de ce problème a été faite en détail dans [9] par le schéma d'Euler implicite en temps et la méthode spectrale en espace. L'objectif du chapitre 4 est de savoir comment mettre en oeuvre sur MATLAB les schémas vue précédemment, on met aussi en évidence l'ordre de convergence de quelques schémas tout en effectuant des simulations pour différentes valeurs du pas de temps h.

Chapitre 1

Rappels sur le cours d'équations différentielles

1.1 Généralités

Définition 1.1.1 *Soit U un ouvert de $\mathbb{R} \times \mathbb{R}^n$ et $f : U \to \mathbb{R}^n$ une application supposée au moins continue par rapport aux deux variables. On appelle* **solution** *de l'équation différentielle*

$$(E) \quad y'(t) = f(t, y(t)),$$

tout couple (J, y) où $J \subset \mathbb{R}$ est un intervalle de $\mathbb{R}$ et y une fonction dérivable définie sur J telle que

$$\forall t \in J,\ (t, y(t)) \in U \text{ et } y'(t) = f(t, y(t)).$$

Définition 1.1.2 *Toute solution (J, y) de (E) définie sur l'intervalle $J = \mathbb{R}$ tout entier est dite* **globale**.

Il arrive qu'on ne cherche pas toutes les solutions d'une EDO mais seulement celles qui vérifient certaines conditions, dites conditions initiales de Cauchy ou tout simplement conditions de Cauchy.

Définition 1.1.3 *Soit $f : U \to \mathbb{R}^n$, $(t_0, y_0) \in U$. On appelle solution du problème de Cauchy associée à la donnée (t_0, y_0) toute solution (J, y) de l'équation (E) vérifiant de plus*

$$(I) \quad t_0 \in J \text{ et } y(t_0) = y_0.$$

On obtient en intégrant $\{(E), (I)\}$ entre t_0 et t

$$y(t) - y_0 = \int_{t_0}^{t} f(s, y(s))ds. \tag{1.1}$$

La solution y du problème de Cauchy $\{(E),(I)\}$ est de classe $\mathcal{C}^1$ et satisfait l'équation intégrale (1.1). Inversement, si y est définie par (1.1), alors elle est continue sur J et $y(t_0) = y_0$. De plus, en tant que primitive de la fonction continue $f(.,y(.)), y \in \mathcal{C}^1(J)$ et satisfait l'équation différentielle $y'(t) = f(t,y(t))$. Ainsi, si f est continue, le problème de Cauchy $\{(E),(I)\}$ est équivalent à l'équation intégrale (1.1). Nous verrons plus loin comment tirer parti de cette équivalence pour les méthodes numériques.

-**Cylindre de sécurité :** On choisit un compact $K_0 = [t_0 - T_0, t_0 + T_0] \times \overline{B}(y_0, r_0) \subset U$, centré en (t_0, y_0), où $T_0, r_0 > 0$ et $\overline{B}(y_0, r_0)$ est la boule fermée de centre y_0 et de rayon r_0. On note $M = \sup_{(t,y)\in K_0} \|f(t,y)\|$. Ce nombre M est fini par la compacité de K_0 et la continuité de f. le choix de la norme sur $\mathbb{R}^n$ est arbitraire.

Proposition 1.1.4 *Soit* $T := \min(T_0, \frac{r_0}{M})$. *Toute solution* y *au problème de Cauchy pour les conditions initiales* (t_0, y_0) *satisfait*

$$|t - t_0| \leq T \Longrightarrow \|y(t) - y_0\| \leq r_0.$$

Cette dernière implication veut dire aucune solution y *ne peut "s'échapper" de* $[t_0 - T, t_0 + T] \times \overline{B}(y_0, r_0)$. *On dit dans ce cas que* $[t_0 - T, t_0 + T] \times \overline{B}(y_0, r_0)$ *est un* **cylindre de sécurité** *pour l'équation* (E).

Preuve :

Supposons que la solution y s'échappe de $[t_0 - T, t_0 + T] \times \overline{B}(y_0, r_0)$ sur l'intervalle $[t_0 - T, t_0 + T]$. Soit τ le premier instant où cela se produit :

$$\tau = \inf\{t \in [t_0, t_0 + T]; \|y(t) - y_0\| > r_0\}.$$

Par définition de τ on a $\|y(t) - y_0\| \leq r_0$ pour $t \in [t_0, \tau[$, donc par continuité de y on obtient $\|y(\tau) - y_0\| = r_0$. Comme $(t, y(t)) \in [t_0 - T, t_0 + T] \times \overline{B}(y_0, r_0) \subset K_0$ pour $t \in [t_0, \tau]$, il vient

$$\|y'(t)\| = \|f(t, y(t))\| \leq M,$$

et

$$r_0 = \|y(\tau) - y_0\| = \| \int_{t_0}^{\tau} y'(u)du\| \leq M(\tau - t_0),$$

donc $\tau - t_0 \geq \frac{r_0}{M}$. Par conséquent si $T \leq \frac{r_0}{M}$, aucune solution ne peut s'échapper de $[t_0 - T, t_0 + T] \times \overline{B}(y_0, r_0)$ sur l'intervalle $[t_0 - T, t_0 + T]$. ∎

Définition 1.1.5 *Une fonction $f : U \to \mathbb{R}^n$ est dite* **localement lipschitzienne** *par rapport à la variable d'état (ou à la seconde variable) si pour tout $(t_0, y_0) \in U$, il existe une constante $C_{t_0,y_0} > 0$ et un voisinage V de (t_0, y_0) dans U tel que*

$$\forall (t, y_1) \in V \,\text{et}\, (t, y_2) \in V, \|f(t, y_1) - f(t, y_2)\| \leq C_{t_0,y_0} \|y_1 - y_2\|.$$

Théorème 1.1.6 (**Cauchy-Lipschitz**) *On suppose que $f : U \to \mathbb{R}^n$ est continue et localement lipschitzienne par rapport à la variable d'état, alors pour tout cylindre de sécurité $C = [t_0 - T, t_0 + T] \times \overline{B}(y_0, r_0)$, l'équation (E)* ***admet*** *une* ***unique*** *solution de condition initiale (t_0, y_0) définie sur l'intervalle $[t_0 - T, t_0 + T]$.*
De plus, si on pose

$$\Phi(y)(t) = y_0 + \int_{t_0}^{t} f(x, y(x))dx,$$

il existe $p \in \mathbb{N}$ tel que la suite itérée $\Phi^p(z)$ converge uniformément vers la solution exacte du problème de Cauchy.

Preuve :
Soit $C = [t_0+T, t_0-T] \times \overline{B}(y_0, r_0) \subset K_0$ un cylindre de sécurité pour (E) avec $T := \min(T_0, \frac{r_0}{M})$, et $M = \sup_{(t,y)\in K_0} \|f(t, y)\|$. Notons $\mathcal{F} = C([t_0 - T, t_0 + T], \overline{B}(y_0, r_0))$ l'ensemble des applications continues de $[t_0 - T, t_0 + T]$ dans $\overline{B}(y_0, r_0)$, muni de la distance de la convergence uniforme $d = \|.\|_\infty$. A toute fonction $y \in \mathcal{F}$ on associe $\Phi(y)$ définie par

$$\Phi(y)(t) = y_0 + \int_{t_0}^{t} f(u, y(u))du, \;\; t \in [t_0 - T, t_0 + T].$$

On montre d'abord que y est une solution de (E) si et seulement si y est un point fixe de Φ. Supposons que y est un point fixe de Φ. Alors, $\forall y \in \mathcal{F}$ on a $\Phi(y) = y$ d'où

$$y(t) = y_0 + \int_{t_0}^{t} f(u, y(u))du.$$

Or f est continue sur U donc y est continue sur U. De plus, y est dérivable sur $[t_0 - T, t_0 + T]$ est sa dérivée égale à $f(t, y(t))$. On a aussi

$$y(t_0) = y_0 + \int_{t_0}^{t} f(u, y(u))du = y_0.$$

Donc y est solution du problème de Cauchy $\{(E), (I)\}$. Supposons maintenant que y est solution de (E), on intégre y' par rapport à u on obtient

$$\int_{t_0}^{t} y'(u)du = \int_{t_0}^{t} f(u, t(u))du = y(t) - y(t_0) = y(t) - y_0$$

donc on a bien

$$y(t) = y_0 + \int_{t_0}^{t} f(u, t(u))du = \Phi(y)(t)$$

et donc y est un point fixe de Φ. Observons que

$$\begin{aligned}
\|\Phi(y)(t) - y_0\| &= \| \int_{t_0}^{t} f(u, y(u))du\| \\
&\leq \int_{t_0}^{t} \|f(u, y(u))\|du \\
&\leq M \int_{t0}^{t} du \\
&\leq MT \\
&\leq r_0,
\end{aligned}$$

donc $\Phi(y) \in \mathcal{F}$. L'opérateur ϕ envoie donc $\mathcal{F}$ dans $\mathcal{F}$. Soient maintenant $y, z \in \mathcal{F}$ et $y_{(p)} = \Phi^p(y)$, $z_{(p)} = \Phi^p(z)$. On a

$$\begin{aligned}
\|y_{(1)}(t) - z_{(1)}(t)\| &= \| \int_{t_0}^{t} (f(u, y(u)) - f(u, z(u)))du\| \\
&\leq | \int_{t_0}^{t} k\|y(u) - z(u)\|du \,| \\
&\leq k|t - t_0|d(y, z).
\end{aligned}$$

De même

$$\begin{aligned}
\|y_{(2)}(t) - z_{(2)}(t)\| &= | \int_{t_0}^{t} k\|y_{(1)}(t) - z_{(1)}(t)\|du| \\
&\leq | \int_{t_0}^{t} kk|t - t_0|d(y, z)du \,| \\
&= k^2 \frac{|t - t_0|^2}{2} d(y, z).
\end{aligned}$$

Par récurrence sur p, on vérifie aussitôt que

$$\|y_{(p)}(t) - z_{(p)}(t)\| \leq k^p \frac{|t - t_0|^p}{p!} d(y, z),$$

en particulier

$$d(\Phi^p(y), \Phi^p(z)) = d(y_{(p)}, z_{(p)}) \leq \frac{k^p T^p}{p!} d(y, z).$$

On a Φ^p est lipschitzienne de rapport $\dfrac{k^p T^p}{p!}$ sur $\mathcal{F}$. Comme $\lim\limits_{p \to +\infty} \dfrac{k^p T^p}{p!} = 0$, il existe p assez grand tel que $\dfrac{k^p T^p}{p!} < 1$; pour une telle valeur de p, Φ^p est une application contractante de $\mathcal{F}$

dans $\mathcal{F}$. Comme $\mathcal{F}$ est un espace métrique complet, cette application admet un point fixe et un seul, c'est-à-dire qu'il existe $y \in \mathcal{F}$ unique telle que $\Phi^p(y) = y$. On a alors

$$\Phi^p(\Phi(y)) = \Phi(\Phi^p(y)) = \Phi(y),$$

et l'unicité du point fixe de Φ^p implique $\Phi(y) = y$. L'application Φ admet donc un point fixe et un seul (l'unicité provenant de ce que tout point fixe de Φ et aussi un point fixe de Φ^p). Nous avons donc bien redémontré la propriété d'existence et d'unicité des solutions du théorème de Cauchy Lipschitz. ∎

1.2 Sur la des solutions

Rappelons qu'une fonction de plusieurs variables est dite de classe $\mathcal{C}^k$ avec k un entier ≥ 1 si elle admet des dérivées partielles continues jusqu'à l'ordre k.
Si la fonction f est de classe $\mathcal{C}^k$, on peut en déduire un résultat sur l'ordre de dérivabilité des solutions y et calculer les dérivées successives de y à l'aide de fonctions construites récursivement à partir de f. C'est utile pour obtenir des majorations fondées sur la formule de Taylor.

Proposition 1.2.1

i) : Si la fonction $f = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix}$ *est de classe* $\mathcal{C}^k, k \geq 1$, *toute solution* $t \mapsto y(t)$, *de l'équation* (E) *est de classe* $\mathcal{C}^{k+1}$.
ii) : Les fonctions $f^{[i]} : U \to \mathbb{R}$, *définies récurssivement pour* $i = 0, ..., k$, *par les formules*

$$f^{[0]} = f, \ f^{[i+1]}(t, y) = \frac{\partial f^{[i]}}{\partial t}(t, y) + \sum_{\ell=0}^{n} f_\ell(t, y) \frac{\partial f^{[i]}}{\partial y_\ell}(t, y)$$

sont de classes respectives $\mathcal{C}^{k-i}$ *et pour toute solution* y *de l'équation* (E) *ses dérivées successives s'obtiennent par les formules*

$$y^{(i+1)}(t) = f^{[i]}(t, y(t))$$

Preuve :
On démontre par récurrence sur i.
-La classe de différentiabilité de $f^{[i]}$, s'obtient par la formule de récurrence et une application directe de la définition de la classe $\mathcal{C}^k$.
-Pour la différentiabilité de la solution y, on montre par récurrence sur i, pour $0 \leq i \leq k$,

l'existence de $y^{(i+1)}$ et la formule annoncée en $ii)$: pour $i = 0$, il s'agit de l'égalité $y'(t) = f(t, y(t))$ qui signifie que $t \mapsto y(t)$ est une solution. On voit sur cette relation que y' est continûment dérivable et la formule de dérivation des fonctions composées s'écrit :

$$\begin{aligned} y''(t) &= \frac{\partial f}{\partial t}(t, y(t)) + \sum_{\ell=0}^{n} y'_\ell(t) \frac{\partial f}{\partial y_\ell}(t, y(t)) \\ &= \frac{\partial f}{\partial t}(t, y(t)) + \sum_{\ell=0}^{n} f_\ell(t, y(t)) \frac{\partial f}{\partial y_\ell}(t, y(t)) \\ &= f^{[1]}(t, y(t)). \end{aligned}$$

Pour le pas général de la récurrence, on suppose que $y^{(i+1)}(t) = f^{[i]}(t, y(t))$, avec $1 \leq i \leq k-1$, alors puisque $f^{[i]}$ est de classe $\mathcal{C}^{k-i}$, avec $k-i \geq 1$, on en déduit que $y^{(i+1)}$ est encore continûment dérivable et par un calcul semblable au cas de y'', on trouve

$$\begin{aligned} y^{(i+2)}(t) = \big(y^{(i+1)}\big)'(t) &= \frac{\partial f^{[i]}}{\partial t}(t, y(t)) + \sum_{\ell=0}^{n} f_\ell(t, y(t)) \frac{\partial f^{[i]}}{\partial y_\ell}(t, y(t)) \\ &= f^{[i+1]}(t, y(t)). \end{aligned}$$

■

Chapitre 2

Méthodes numériques à un pas

Pour des équations différentielles d'un intérêt pratique, on trouve rarement la solution $y(t)$ exprimée avec une formule exacte et on ne peut exprimer la solution que sous forme implicite. Par exemple, la solution de $y' = (y-t)/(y+t)$ vérifie la relation implicite

$$\frac{1}{2}\ln(t^2+y^2) + \arctan(\frac{y}{t}) = C,$$

où C est une constante. Dans d'autre cas, on ne parvient même pas à représenter la solution sous forme implicite. Par exemple, la solution générale de $y' = e^{-t^2}$ ne peut s'exprimer qu'à l'aide d'un développement en séries.

On a vu dans le chapitre précédent qu'une solution de l'équation

$$y'(t) = f(t, y(t))$$

est obtenue en intégrant de t_0 à t :

$$y(t) = y(t_0) + \int_{t_0}^{t} f(s, y(s))ds.$$

Le problème avec cette solution est que l'inconnue y se trouve sous l'intégrale.

On reprend les notations précédentes concernant un cylindre de sécurité, et on ne s'occupe pour simplifier que des solutions à droite c'est à dire sur l'intervalle $[t_0, t_0 + T]$.

L'objectif de ce chapitre est de décrire un certain nombre de méthodes permettant de résoudre numériquement le problème de Cauchy de condition initiale

$$(I) \qquad y(t_0) = y_0$$

pour une équation différentielle

$$(E) \qquad y'(t) = f(t, y(t)).$$

Dans ce mémoire, on supposera que f satisfait aux conditions du théorème de Cauchy-Lipschitz. Ceci assure que le problème {(E),(I)} admet une unique solution. On notera $t \mapsto y(t)$ la solution unique du problème sur $[t_0, t_0 + T]$ dont le graphe est contenu dans un cylindre de sécurité $[t_0 - T, t_0 + T] \times \overline{B}(y_0, r_0)$.

Nous avons choisi ici d'exposer le cas des équations unidimensionnelles dans le seul but de simplifier les notations ; le cas des systèmes dans $\mathbb{R}^n$ est tout à fait identique, à condition de considérer y et f comme des fonctions vectorielles.

Etant donné une subdivision $t_0 < t_1 < \ldots < t_N = t_0 + T$ de $[t_0, t_0 + T]$, on cherche à déterminer des valeurs approchées y_n des valeurs $y(t_n), 0 \leq n \leq N$, prises par la solution exacte y. On notera les pas successifs

$$h_n = t_{n+1} - t_n, \quad 0 \leq n \leq N - 1,$$

et

$$h_{\max} = \max_n(h_n),$$

le maximum du pas.

Définition 2.0.2 *Une méthode (ou schéma) à un pas* **explicite** *est une équation de récurrence de la forme*

$$\begin{cases} y_{n+1} = y_n + h_n \Phi(t_n, y_n, h_n), & 0 \leq n \leq N - 1, \\ t_{n+1} = t_n + h_n \end{cases}$$

Le domaine de définition de Φ contient au moins $U \times [0, \delta], \delta > 0$.

Définition 2.0.3 *Un schéma à un pas est dit* **implicite** *s'il est de la forme*

$$\begin{cases} y_{n+1} = y_n + h_n \Phi(t_n, y_n, y_{n+1}, h_n), & 0 \leq n \leq N - 1, \\ t_{n+1} = t_n + h_n \end{cases}$$

C'est-à-dire si Φ dépend non linéairement de y_{n+1}.

Pour ce type de méthodes il s'agira le plus souvent de s'assurer que l'équation

$$y = y_n + h\Phi(t_n, y_n, y, h)$$

a une unique solution du moins pour tout h assez petit. Dans les cas les plus courants cela résultera du théorème du point fixe.

2.1 Quelques exemples de méthodes à un pas

• **Méthodes d'Euler**

Une façon d'obtenir une multitude de schémas, est d'intégrer l'EDO (E) sur $[t_n, t_{n+1}]$:

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} f(t, y(t))dt,$$

et ensuite d'approcher l'intégrale.

Par exemple

– Intégration par la méthode des rectangles à gauche

$$\int_{t_n}^{t_{n+1}} f(t, y(t))dt \approx h_n f(t_n, y(t_n)),$$

ce qui donne le schéma d'**Euler explicite**

$$y_{n+1} = y_n + h_n f(t_n, y_n). \tag{2.1}$$

– Intégration par la méthode des rectangles à droite

$$\int_{t_n}^{t_{n+1}} f(t, y(t))dt \approx h_n f(t_{n+1}, y(t_{n+1})),$$

ce qui donne le schéma d'**Euler implicite**

$$y_{n+1} = y_n + h_n f(t_{n+1}, y_{n+1}). \tag{2.2}$$

– Intégration par la méthode du point milieu

$$\int_{t_n}^{t_{n+1}} f(t, y(t))dt \approx h_n f(t_n + \frac{h_n}{2}, y(t_n + \frac{h_n}{2})), \tag{2.3}$$

ici, on connaît uniquement la valeur de y_n, et pour donner une approximation de la solution au point $t_n + \frac{h_n}{2}$, on utilise le schéma d'Euler explicite

$$y(t_n + \frac{h_n}{2}) \approx y(t_n) + \frac{h_n}{2} f(t_n, y(t_n)),$$

ce qui donne le schéma d'**Euler modifié**

$$y_{n+1} = y_n + h_n \Big(f(t_n + \frac{h_n}{2}, y_n + \frac{h_n}{2} f(t_n, y_n) \Big),$$

c'est une méthode qui nécessite deux évaluations du second membre f.

• **Méthode de Taylor d'ordre** p

Supposons maintenant que f soit de classe $\mathcal{C}^p$, on a vu alors que la solution exacte y de (E) est de classe $\mathcal{C}^{p+1}$ et on a défini des fonctions $f^{[k]}$ construites par récurrence à partir de f et de ses dérivées partielles telles que

$$y^{(k)}(t) = f^{[k-1]}(t, y(t)),$$

pour $k = 1, ..., p+1$. La formule de Taylor d'ordre p donne

$$y(t_n + h_n) = y(t_n) + \sum_{k=1}^{p} \frac{1}{k!} h_n^k f^{[k-1]}(t_n, y(t_n)) + o(h_n^p),$$

ou avec la formule de Taylor avec reste de Lagrange :

$$\begin{aligned} y(t_n + h_n) &= y(t_n) + \sum_{k=1}^{p} \frac{1}{k!} h_n^k f^{[k-1]}(t_n, y(t_n)) \\ &+ \frac{1}{(p+1)!} h_n^{p+1} f^{[p]}(t_n + \theta h_n, y(t_n + \theta h_n)),\ \theta \in]0, 1[. \end{aligned} \tag{2.4}$$

On est donc amené à considérer l'algorithme suivant, appelé **méthode de Taylor** d'ordre p :

$$(\mathcal{T}_p) \quad \begin{cases} y_{n+1} = y_n + \sum_{k=1}^{p} \frac{1}{k!} h_n^k f^{[k-1]}(t_n, y_n), \\ t_{n+1} = t_n + h_n. \end{cases}$$

D'après la définition 2.0.2, cet algorithme correspond au choix

$$\Phi(t, y, h) = \sum_{k=1}^{p} \frac{1}{k!} h^{k-1} f^{[k-1]}(t, y(t)).$$

On peut remarquer facilement que la méthode d'Euler n'est autre que la méthode de Taylor $(\mathcal{T}_1)$.

Remarque 2.1.1 *La méthode de Taylor n'est en général pas utilisée en pratique car le calcul des valeurs $f^{[k]}$ est trop coûteux.*

2.2 Etude des méthodes à un pas

Dans la section précédente, plusieurs schémas numériques ont été présentés et afin de les comparer plusieurs critères seront étudiés. Les trois principaux sont la convergence, la stabilité et la consistance, permettant de relier la solution exacte à la solution approchée. L'étude concernera les méthodes explicites à un pas .

Définition 2.2.1 *Si on remplace la solution approchée par la solution exacte dans le schéma numérique, quelle est l'erreur commise en fonction de h_n ?. Pour ceci, on définit l'erreur locale de consistance*

$$e_n = y(t_{n+1}) - \overline{y}_{n+1}, \tag{2.5}$$

avec $\overline{y}_{n+1}$ est la solution du schéma issue de $y(t_n)$, i.e.

$$\overline{y}_{n+1} = y(t_n) + h_n \Phi(t_n, y(t_n), h_n).$$

Définition 2.2.2 (*Consistance*) *On dit que la méthode est consistante si pour toute solution exacte y la somme des erreurs de consistance relatives à y, soit $\sum_{n=0}^{N-1} |e_n|$, tend vers 0 quand $h_{\max}$ tend vers 0.*

Il existe une méthode simple pour vérifier la consistance des méthodes numériques. Dans l'énoncé suivant, on suppose que Φ remplit la condition suivante presque toujours réalisée dans les exemples usuels : Φ est continue sur un ouvert contenant $U \times [0, \delta], \delta > 0$.

Théorème 2.2.3 (*Condition nécessaire et suffisante de consistance*)
La méthode à un pas définie par la fonction Φ est consistante si et seulement si

$$\forall (t, y) \in U, \;\; \Phi(t, y, 0) = f(t, y).$$

Preuve : Soit y une solution exacte de l'équation (E) et soient

$$e_n = y(t_{n+1}) - y(t_n) - h_n \Phi(t_n, y(t_n), h_n)$$

les erreurs de consistance correspondantes. D'après le théorème des accroissements finis, il existe $c_n \in]t_n, t_{n+1}[$ tel que

$$y(t_{n+1}) - y(t_n) = h_n y'(c_n) = h_n f(c_n, y(c_n)),$$

d'où

$$e_n = h_n (f(c_n, y(c_n)) - \Phi(t_n, y(t_n), h_n)) = h_n(\alpha_n + \beta_n),$$

avec

$$\alpha_n = f(c_n, y(c_n)) - \Phi(c_n, y(c_n), 0),$$

$$\beta_n = \Phi(c_n, y(c_n), 0) - \Phi(t_n, y(t_n), h_n).$$

D'après l'uniforme continuité de Φ sur $C \times [0, \delta]$, où C est le cylindre de sécurité sur lequel on travaille : on a
$\forall \varepsilon > 0, \exists \eta > 0, \exists \gamma > 0$, tels que $|h| < \eta, |t_1 - t_2| < \eta, |y_1 - y_2| < \gamma \Rightarrow |\Phi(t_1, y_1, 0) - \Phi(t_2, y_2, h)| <$

ε.

Par ailleurs, quitte à diminuer η, on peut s'assurer en utilisant l'uniforme continuité de y sur $[t_0, t_0+T]$ que

$$h_{\max} < \eta \Rightarrow |y(c_n) - y(t_n)| < \gamma.$$

En enchaînant les deux implications précédentes

$$h_{\max} < \eta \Rightarrow \Phi(c_n, y(c_n), 0) - \Phi(t_n, y(t_n), h_n) < \varepsilon.$$

Par conséquent

$$h_{\max} < \eta \Rightarrow \sum_{n=0}^{N-1} h_n |\Phi(c_n, y(c_n), 0) - \Phi(t_n, y(t_n), h_n)| < \varepsilon \sum_{n=0}^{N-1} h_n = \varepsilon T.$$

avec les notations abrégées, on obtient

$$\Big| \sum_{n=0}^{N-1} |e_n| - \sum_{n=0}^{N-1} h_n |\alpha_n| \Big| \leq \sum_{n=0}^{N-1} h_n |\beta_n| \leq \varepsilon \sum_{n=0}^{N-1} h_n = \varepsilon T.$$

On en déduit

$$\begin{aligned} \lim_{h_{\max}\to 0} \sum_{n=0}^{N-1} |e_n| &= \lim_{h_{\max}\to 0} \sum_{n=0}^{N-1} h_n |\alpha_n| \\ &= \int_{t_0}^{t_0+T} |f(t, y(t)) - \Phi(t, y(t), 0)| dt, \end{aligned}$$

car $\sum_{n=0}^{N-1} h_n |\alpha_n|$ est une somme de Riemann de l'intégrale précédente.
La condition de consistance est équivalente au fait que cette limite est nulle donc à

$$\int_{t_0}^{t_0+T} |f(t, y(t)) - \Phi(t, y(t), 0)| dt = 0.$$

La nullité de cette intégrale de fonction positive continue impose pour tout t, $f(t, y(t)) = \Phi(t, y(t), 0)$. Dans tout ce raisonnement la condition initiale (t_0, y_0) est arbitraire, et donc l'égalité $f(t_0, y_0) = \Phi(t_0, y_0, 0)$ est valable pour tout $(t_0, y_0) \in U$. ∎

Définition 2.2.4 *On dit qu'une méthode à un pas est* **consistante d'ordre** $\geq p$ *si pour toute solution exacte y de (E), il existe une constante $C \geq 0$ telle que l'erreur de consistance vérifie*

$$|e_n| \leq C h_n^{p+1}, \quad 0 \leq n \leq N.$$

Elle est dite d'ordre p (exactement) si elle est d'ordre $\geq p$ mais pas d'ordre $\geq p+1$.

Remarque 2.2.5 *Des définitions 2.2.2 et 2.2.4, on déduit immédiatement qu'une méthode à un pas est consistante si elle est au moins d'ordre 1.*

Exemple 2.2.6 La consistance de la méthode d'Euler explicite
Puisque pour la méthode d'Euler explicite

$$\overline{y}_{n+1} = y(t_n) + h_n f(t_n, y(t_n)),$$

et d'après le développement de Taylor d'ordre 2 *on a*

$$y(t_{n+1}) = y(t_n) + h_n f(t_n, y(t_n) + \frac{h_n^2}{2} y''(\xi),\ \xi \in]t_n, t_{n+1}[,$$

l'erreur de consistance est

$$e_n = \frac{h_n^2}{2} y''(\xi).$$

Ceci implique, si la dérivée de f *est bornée que* $\sum_{n=0}^{N-1} |e_n|$, *tend vers* 0 *quand* $h_{\max}$ *tend vers* 0. *La méthode d'Euler explicite est donc consistante d'ordre* 1

Exemple 2.2.7 *La méthode de Taylor* $(\mathcal{T}_p)$ *est du point de vue de l'erreur de consistance d'ordre p. Plus précisément si on considère un cylindre de sécurité* $C = [t_0 - T, t_0 + T] \times \overline{B}(y_0, r)$, *on a la majoration*

$$e_n \leq \frac{1}{(p+1)!} \sup_{(t,y)\in C} \|f^{[p]}(t,y)\| h_n^{p+1}.$$

En effet, selon la formule de Taylor qui a servi de base à la méthode, on obtient directement en prenant la formule avec reste de Lagrange (2.4)

$$\begin{aligned} e_n &= y(t_n + h_n) - y(t_n) - \sum_{k=1}^{p} h_n^k \frac{1}{k!} f^{[k-1]}(t_n, y(t_n)) \\ &= \frac{1}{(p+1)!} f^{[p]}(t_n + \theta h_n, y(t_n + \theta h_n)) h_n^{p+1} \leq \frac{1}{(p+1)!} \sup_{(t,y)\in C} \|f^{[p]}(t,y)\| h_n^{p+1}. \end{aligned}$$

Décrivons maintenant une méthode générale permettant de calculer l'ordre de consistance :

Proposition 2.2.8 *Sous l'hypothèse que* f *et* Φ *sont de classe* $\mathcal{C}^p$,*une méthode à un pas est d'ordre p si et seulement si les conditions suivantes sont remplies :*

$$\frac{\partial^\ell \Phi}{\partial h^\ell}(t_n, y(t_n), 0) = \frac{1}{(\ell+1)!} f^{[\ell]}(t_n, y(t_n)),\ \textit{pour}\ \ 0 \leq \ell \leq p-1,\ \ 0 \leq n \leq N.$$

Preuve :

L'erreur de consistance est donnée par

$$e_n = y(t_{n+1}) - y(t_n) - h_n\Phi(t_n, y(t_n), h_n).$$

Si on suppose que Φ est de classe $\mathcal{C}^p$, et puisque f est de classe $\mathcal{C}^p$, la solution y est de classe $\mathcal{C}^{p+1}$, par conséquent, en appliquant la formule de Taylor avec reste de Lagrange (2.4), on a l'existence de $c_n, d_n \in]t_n, t_{n+1}[$ tels que :

$$\begin{aligned} y(t_{n+1}) - y(t_n) &= y(t_n + h_n) - y(t_n) \\ &= \sum_{k=1}^{p} \frac{1}{k!} h_n^k y^{(k)}(t_n) + \frac{1}{(p+1)!} h_n^{p+1} y^{(p+1)}(c_n) \\ &= \sum_{\ell=0}^{p-1} \frac{1}{(\ell+1)!} h_n^{\ell+1} f^{[\ell]}(t_n, y(t_n)) + \frac{1}{(p+1)!} h_n^{p+1} f^{[p]}(c_n, y(c_n)), \end{aligned}$$

et

$$\Phi(t_n, y(t_n), h_n) = \sum_{\ell=0}^{p-1} \frac{1}{\ell!} h_n^{\ell} \frac{\partial^\ell \Phi}{\partial h^\ell}(t_n, y(t_n), 0) + \frac{h_n^p}{p!} \frac{\partial^p \Phi}{\partial h^p}(t_n, y(t_n), d_n).$$

On en tire

$$\begin{aligned} e_n &= h_n\Big(\sum_{\ell=0}^{p-1} \frac{h_n^\ell}{\ell!} \big(\frac{f^{[\ell]}(t_n, y(t_n))}{\ell+1} - \frac{\partial^\ell \Phi(t_n, y(t_n), 0)}{\partial h^\ell} \big) \Big) \\ &+ \frac{h_n^{p+1}}{p!} \Big(\frac{f^{[p]}(c_n, y(c_n))}{p+1} - \frac{\partial^p \Phi}{\partial h^p}(t_n, y(t_n), d_n) \Big). \end{aligned}$$

Pour que cette expression soit un développement limité en h_n de la forme $o(h_n^{p+1})$, il faut et il suffit que tous les termes $\dfrac{f^{[\ell]}(t_n, y(t_n))}{\ell+1} - \dfrac{\partial^\ell \Phi(t_n, y(t_n), 0)}{\partial h^\ell}$ soient nuls. Le reste fournit la majoration de l'énoncé avec la constante

$$C = \frac{1}{(p+1)!} \|f^{[p]}\|_\infty + \frac{1}{p!} \|\frac{\partial^p \Phi}{\partial h^p}\|_\infty,$$

où les normes utilisées sont les normes du sup $\|\cdot\|_\infty$ et les solutions sont supposées confinées dans le compact $[t_0, t_0 + T] \times J \times [0, \delta]$, où J est un compact. ∎

Dans la pratique les valeurs de y_n sont perturbées par des valeurs voisines $\tilde{y}_n$ pour deux raisons :

1) *Erreurs d'arrondis* : on représente en machine la valeur y_n issue du calcul par un nombre décimal à q chiffres. $|y_n - \tilde{y}_n|$ est alors l'erreur d'arrondi majorée en valeur relative par $10^{-q} y_n$.

2) *Incertitude expérimentale* : dans la plupart des problèmes concrets, la "vraie" valeur de y_0 est remplacée par une valeur $\tilde{y_0}$ tirée d'une expérience, d'une hypothèse, etc, $|y_0 - \tilde{y}_0|$ est donc majorée par un nombre qui dépend de la précision expérimentale.

La méthode ne peut donc être utile que si la perturbation sur $y_n - \tilde{y}_n$ provoquée par une faible perturbation $|y_0 - \tilde{y}_0|$ des données initiales et par les erreurs d'arrondi est faible. On est donc amené à la définition suivante.

Définition 2.2.9 *Un schéma numérique est dit stable s'il permet de contôler la solution quand on perturbe les données. On dit que la méthode est stable s'il existe une constante $S \geq 0$, appelée constante de stabilité, telle que pour toutes suites $(y_n), (\tilde{y}_n)$ définies par*

$$y_{n+1} = y_n + h_n \Phi(t_n, y_n, h_n), \quad 0 \leq n < N,$$

$$\tilde{y}_{n+1} = \tilde{y}_n + h_n \Phi(t_n, \tilde{y}_n, h_n) + \varepsilon_n, \quad 0 \leq n < N,$$

on a

$$\max_{0 \leq n \leq N} |\tilde{y}_n - y_n| \leq S\Big(|\tilde{y}_0 - y_0| + \sum_{0 \leq n < N} |\varepsilon_n|\Big).$$

où $\varepsilon_n \in \mathbb{R}$ (appelé la perturbation).

Autrement dit, une petite erreur initiale $|\tilde{y}_0 - y_0|$ et de petites erreurs d'arrondi ε_n dans le calcul récurrent des $\tilde{y}_n$ provoquent une erreur finale $\max\limits_n |\tilde{y}_n - y_n|$ contrôlable.

Dans la nature on trouve des phénomènes qui sont modélisés par des équations différentielles. Si on pend comme exemple les problèmes de météorologie, il sont modélisés par le système de Lorenz (voir [6]) : le phénomène de l'orage amène tout un lot de perturbations et peut beaucoup modifier la température du jour suivant. L'orage est très sensible à sa température initiale. Par exemple, il est possible qu'un orage qui arrive alors qu'il fait 10 degrés amène une température de 20 degrés le jour suivant alors qu'un orage qui arrive alors qu'il fait 9.9 degrés amène une température de 5 degrés. Une petite différence de 0.1 degré au début de l'orage a amené une grande différence de 15 degrés à la fin. même la plus petite perturbation comme les battements d'ailes d'un papillon pourront finalement avoir un grand effet, ont dit dans ce cas que ces problèmes ont un comportement chaotique, c'est pourquoi nous ne pouvons prédire le temps à long terme.

Théorème 2.2.10 *(Une condition suffisante de stabilité)*
Si Φ est lipschitzienne par rapport à la variable y, la méthode est stable. De plus si L est la constante de Lipschitz pour Φ, la constante de stabilité est $S = e^{LT}$.

La démonstration de ce théorème repose sur le lemme suivant :

Lemme 2.2.11 *(de Gronwall discret) - Soient les suites $h_n, \theta_n \geq 0$ et $\varepsilon_n \in \mathbb{R}$ telles que*

$$\theta_{n+1} \leq (1 + Lh_n)\theta_n + |\varepsilon_n|,$$

alors

$$\theta_n \leq e^{L(t_n - t_0)}\theta_0 + \sum_{i=0}^{n-1} e^{L(t_n - t_{i+1})}|\varepsilon_i|.$$

Preuve : Le lemme se vérifie par récurrence sur n. Pour $n = 0$, l'inégalité se réduit à $\theta_0 \leq \theta_0$. Supposons maintenant l'inégalité vraie à l'ordre n. On sait que

$$1 + Lh_n \leq e^{Lh_n} = e^{L(t_{n+1} - t_n)}.$$

Par hypothèse on a

$$\begin{aligned} \theta_{n+1} &\leq (1 + Lh_n)\theta_n + |\varepsilon_n| \\ &\leq e^{L(t_{n+1} - t_n)}\theta_n + |\varepsilon_n| \\ &\leq e^{L(t_{n+1} - t_0)}\theta_0 + \sum_{i=0}^{n-1} e^{L(t_{n+1} - t_{i+1})}|\varepsilon_i| + |\varepsilon_n|. \end{aligned}$$

L'inégalité s'ensuit à l'ordre $n + 1$. ∎

Preuve du théorème 2.2.10 : Considérons deux suites (y_n) et $(\tilde{y}_n)$ telles que

$$y_{n+1} = y_n + h_n\Phi(t_n, y_n, h_n),$$

$$\tilde{y}_{n+1} = \tilde{y}_n + h_n\Phi(t_n, \tilde{y}_n, h_n) + \varepsilon_n.$$

Par différence, on obtient

$$|\tilde{y}_{n+1} - y_{n+1}| \leq |\tilde{y}_n - y_n| + h_n|\Phi(t_n, y_n, h_n) - \Phi(t_n, \tilde{y}_n, h_n)| + |\varepsilon_n|.$$

On pose $\theta_n = |y_n - \tilde{y}_n|$. Par défnition de la constante de Lipschitz pour Φ,

$$\Phi(t, y_1, h) - \Phi(t, y_2, h) \leq L|y_1 - y_2|,$$

quel que soient $(t, y_1, h), (t, y_2, h)$ dans le domaine de définition de Φ. La majoration suivante découle aussitôt de la défnition de la suite (θ_n) et de la condition de Lipschitz

$$\theta_{n+1} \leq (1 + h_n L)\theta_n + \varepsilon_n.$$

Comme $t_n - t_0 \leq T$ et $t_n - t_{i+1} \leq T$, le lemme de Gronwall implique

$$\max_{0 \leq n \leq N} \theta_n \leq e^{LT}\Big(\theta_0 + \sum_{i=0}^{N-1} |\varepsilon_i|\Big),$$

et le théorème est démontré. ∎

Remarque 2.2.12 *Ces calculs ne sont corrects que tant que les (t_n, y_n, h_n) et $(t_n, \tilde{y}_n, h_n)$ restent dans le domaine où Φ est Lipschitzienne de constante L.*

Une autre notion très importante en pratique est la suivante :

Définition 2.2.13 *La convergence est une propriéte de la solution numérique. Une méthode numérique est dite convergente si pour toute solution exacte y définie sur un intervalle $[t_0, t_0+T]$ et toute suite (y_n) construite selon le schéma numérique considéré, à partir de y_0 et d'une subdivision de $[t_0, t_0 + T]$ on a la relation de convergence uniforme :*

$$\lim_{\substack{h_{\max}\to 0 \\ y_0 \to y(t_0)}} \max_{0\le n\le N} |y_n - y(t_n)| \to 0.$$

La quantité $\max_{0\le n\le N} |y_n - y(t_n)|$ notée e s'appelle l'erreur globale. C'est évidemment cette erreur qui importe dans la pratique.
Si $\exists c > 0$ tel que

$$\max_{0\le n\le N} |y_n - y(t_n)| \le ch^p.$$

La méthode est dite convergente d'ordre p.

La consistance et la stabilité sont en général relativement facile à démontrer. La convergnence demande souvent des démonstrations longues et ardues. Le théorème suivant permet d'obvier à cette difficulté.

Théorème 2.2.14 *Une méthode numérique à un pas qui est stable et consistante est convergente.*

Preuve : Posons $\tilde{y}_n = y(t_n)$. Dans ce cas l'erreur de consistance est par défnition le réel e_n qui complète la formule

$$y(t_{n+1}) = \tilde{y}_{n+1} = \tilde{y}_n + h_n\Phi(t_n, \tilde{y}_n, h_n) + e_n.$$

Donc e_n joue pour la suite des $y(t_n)$ le rôle de la perturbation ε_n introduite dans la définition 2.2.9. D'après la définition de la stabilité on a donc

$$\max_{0\le n\le N} |y_n - y(t_n)| \le S\big(|y(t_0) - y_0| + \sum_{n=0}^{N-1} e_n\big).$$

L'hypothèse de consistance donne alors immédiatement le résultat annoncé. ∎

Exemple 2.2.15 *Si f est lipschitzienne en y, les méthodes d'Euler explicite et du point milieu sont convergentes.*

En effet : d'après le Théorème 2.2.14, il suffit pour cela d'établir la consistance et la stabilité.
- La consistance est une conséquence directe du Théorème 2.2.3 et de l'égalité $\Phi_{|_{h=0}} = f$. C'est aussi valable pour la méthode de Taylor d'ordre p.
-La stabilité se déduit du Théorème 2.2.10 et du fait que Φ est lipschitzienne : pour la méthode d'Euler explicite c'est immédiat car $\Phi = f$. Pour la méthode du point milieu cela résulte des calculs suivants : on a

$$\Phi(t, y, h) = f(t + \frac{h}{2}, y + \frac{h}{2} f(t, y)),$$

donc si f est lipshitzienne en y de constante de Lipschitz L on obtient :

$$\begin{aligned} |\Phi(t, y_1, h) - \Phi(t, y_2, h)| &\leq L|y_1 - y_2 + \frac{h}{2}\big(f(t, y_1) - f(t, y_2)\big)| \\ &\leq L|y_1 - y_2| + \frac{hL}{2}|f(t, y_1) - f(t, y_2)| \\ &\leq (L + \frac{h_{\max}}{2} L^2)|y_1 - y_2|, \end{aligned}$$

d'où le caractère lipschitzien de Φ avec la constante de Lipschitz $\Lambda = L + \frac{h_{\max}}{2} L^2$.

2.2.1 Influence de l'ordre de consistance de la méthode sur l'erreur globale

On considère une méthode consistante et stable de constante de stabilité S qui est d'ordre p. On a les majoration d'erreurs suivantes. D'abord le cumul des erreurs de consistance (qui n'est pas l'erreur globale ! !) est :

$$\sum_{0 \leq n < N} e_n \leq C \sum_{0 \leq n < N} h_n^{p+1} \leq C h_{\max}^p \sum_{0 \leq n \leq N} h_n = C T h_{\max}^p.$$

Par définition de la stabilité, on trouve alors :

$$\max_{0 \leq n < N} |y_n - y(t_n)| \leq S(|y_0 - y(t_0)| + C T h_{\max}^p) \tag{2.6}$$

L'erreur globale est donc majorée dans le cas d'absence (ou de négligence) d'erreur sur la condition initiale par

$$SCT h_{\max}^p.$$

Si la constante SCT n'est pas trop grande (disons $\leq 10^2$), une méthode d'ordre 3 avec un pas maximum $h_{\max} = 10^{-2}$ permet d'atteindre une précision globale de l'ordre de 10^{-4}.

2.2.2 Visualisation de l'ordre de convergence

Si l'erreur globale $e = ch^p$ alors $log(e) = log(c) + plog(h)$. Donc la pente de la droite $log(e)$ en représentation logarithmique corespond à p.
Traçons un graphe de l'erreur globale e en fonction de h en échelle logarithmique,

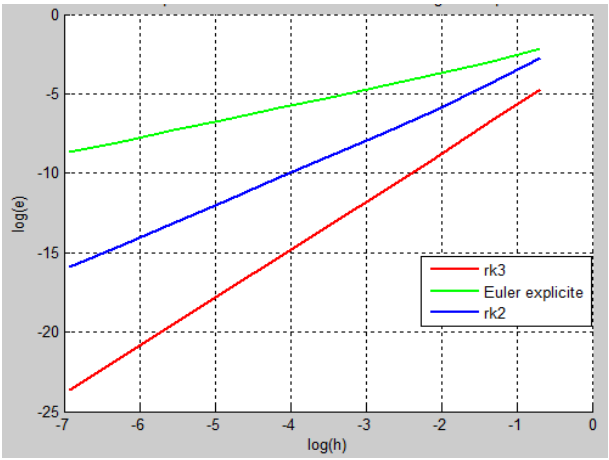

FIG. 2.1 – Graphe des mêmes données en échelle logarithmique

ceci signifie que le graphe porte en abscisse les valeurs de $log(h)$ et en ordonnées les valeurs de $log(e)$. On voit sur le graphe que : l'ordre de la méthode d'Euler explicite est 1, et de Runge Kutta 2 est 2 et de Runge Kutta 3 est 3. Il y a une manière non graphique d'établir l'ordre de convergence d'une méthode quand on connaît les erreurs globales pour quelques valeurs du paramètre de discrétisation $h_i, i = 1, \cdots, N$: elle consiste à supposer que e_i est égale à ch_i^p, où c ne dépend pas de i, donc

$$\frac{e_i}{e_{i-1}} = \left(\frac{h_i}{h_{i-1}}\right)^p \Leftrightarrow \log\left(\frac{e_i}{e_{i-1}}\right) = p\log\left(\frac{h_i}{h_{i-1}}\right)$$

Il s'en suit que p peut être estimé par la moyenne des valeurs

$$p_i = \frac{\log\left(\frac{e_i}{e_{i-1}}\right)}{\log\left(\frac{h_i}{h_{i-1}}\right)}, i = 2, \cdots, N. \tag{2.7}$$

(Voir le chapitre 4 pour la mise en oeuvre)

2.3 Méthodes de type Runge-Kutta

Les méthodes de type Runge-Kutta permettent d'obtenir une plus grande précision que les méthodes d'Euler (dans le sens où elles donnent en général des solutions numériques plus proches

des solutions analytiques que les méthodes d'Euler). Cette précision est obtenue par l'utilisation d'un pas de calcul intermédiare. Les deux méthodes de Runge-Kutta les plus employées sont l'algorithme dit $(RK2)$ à deux pas de calcul et l'algorithme dit $(RK4)$ à quatre pas de calcul.

2.3.1 Description de la méthode

On considère comme d'habitude le problème de Cauchy $\{(E),(I)\}$ avec une solution exacte $y(t)$ sur $[t_0, t_0+T]$ et une subdivision $t_0 < t_1 < ... < t_N = t_0 + T$.
L'idée est de calculer par récurrence les points (t_n, y_n) en utilisant des points intermédiares $(t_{n,i}, y_{n,i})$ avec

$$t_{n,i} = t_n + c_i h_n, \quad 1 \leq i \leq q, \quad c_i \in [0,1].$$

A chacun de ces points on associe la pente correspondante

$$p_{n,i} = f(t_{n,i}, y_{n,i}).$$

Soit y la solution exacte de l'équation (E). On a

$$\begin{aligned} y(t_{n,i}) &= y(t_n) + \int_{t_n}^{t_{n,i}} f(t, y(t))dt \\ &= y(t_n) + h_n \int_0^{c_i} f(t_n + uh_n, y(t_n + uh_n))du, \end{aligned}$$

grâce au changement de variable $t = t_n + uh_n$. De même

$$y(t_{n+1}) = y(t_n) + h_n \int_0^1 f(t_n + uh_n, y(t_n + uh_n))du$$

On se donne alors pour chaque $i = 1, 2, ..., q$ une méthode d'intégration approchée

$$\int_0^{c_i} g(t)dt \simeq \sum_{j=1}^{i-1} a_{ij} g(c_j), \qquad (M_i)$$

ces méthodes pouvant être différentes. On se donne également une méthode d'intégration approchée sur $[0,1]$

$$\int_0^1 g(t)dt \simeq \sum_{j=1}^{q} b_j g(c_j). \qquad (M)$$

En appliquant ces méthodes d'intégration à $g(u) = f(t_n + uh_n, y(t_n + uh_n))$, il vient

$$y(t_{n,i}) \simeq y(t_n) + h_n \sum_{j=1}^{i-1} a_{ij} f(t_{n,j}, y(t_{n,j})),$$

$$y(t_{n+1}) \simeq y(t_n) + h_n \sum_{j=1}^{q} b_j f(t_{n,j}, y(t_{n,j})).$$

La méthode de Runge-Kutta correspondante est définie par l'algorithme

$$\begin{cases} t_{n,i} = t_n + c_i h_n \\ y_{n,i} = y_n + h_n \sum_{j=1}^{i-1} a_{ij} p_{n,j} \\ p_{n,i} = f(t_{n,i}, y_{n,i}), \quad 1 \leq i \leq q \\ t_{n+1} = t_n + h_n \\ y_{n+1} = y_n + h_n \sum_{j=1}^{q} b_j p_{n,j} \end{cases}$$

Conventionnellement elle est représentée par le tableau

(M_1)	c_1	0	0	$\cdots$	0	0
(M_2)	c_2	a_{21}	0	$\cdots$	0	0
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
	$\vdots$	$\vdots$	$\vdots$		0	0
(M_q)	c_q	a_{q1}	a_{q2}	$\cdots$	a_{qq-1}	0
(M)		b_1	b_2	$\cdots$	b_{q-1}	b_q

où les méthodes d'intégration approchées correspondent aux lignes. On pose par convention $a_{ij} = 0$ pour $j \geq i$.

Hypothèse : On supposera que les méthodes d'intégration (M_i) et (M) sont d'ordre 0, *i.e.*

$$\sum_{j=1}^{i-1} a_{i,j} = c_i, \quad \sum_{j=1}^{q} b_j = 1. \tag{2.8}$$

En particulier, on aura toujours

$$c_1 = 0,\ t_{n,1} = t_n,\ y_{n,1} = y_n,\ p_{n,1} = f(t_n, y_n).$$

2.3.2 Exemples

Exemple 1 : Pour $q = 1$, le seul choix possible est

0	0
	1

On a ici $c_1 = 0,\ a_{11} = 0,\ b_1 = 1$. L'algorithme est donné par

$$\begin{cases} p_{n,1} = f(t_n, y_n) \\ t_{n+1} = t_n + h_n \\ y_{n+1} = y_n + h_n p_{n,1} \end{cases}$$

Il s'agit de la méthode d'Euler explicite.

Exemple 2 : Pour $q = 2$, on considère les tableaux de la forme

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \alpha & \alpha & 0 \\ \hline & 1-\frac{1}{2\alpha} & \frac{1}{2\alpha} \end{array}$$

où $\alpha \in]0, 1]$.
L'algorithme s'écrit ici

$$\begin{cases} p_{n,1} = f(t_n, y_n) \\ t_{n,2} = t_n + \alpha h_n \\ y_{n,2} = y_n + \alpha h_n p_{n,1} \\ p_{n,2} = f(t_{n,2}, y_{n,2}) \\ t_{n+1} = t_n + h_n \\ y_{n+1} = y_n + h_n\big((1 - \dfrac{1}{2\alpha})p_{n,1} + \dfrac{1}{2\alpha}p_{n,2}\big), \end{cases}$$

ou encore sous forme condensée :

$$y_{n+1} = y_n + h_n\big((1 - \frac{1}{2\alpha})f(t_n, y_n) + \frac{1}{2\alpha}f(t_n + \alpha h_n, y_n + \alpha h_n f(t_n, y_n))\big).$$

Pour $\alpha = \frac{1}{2}$ par exemple, on retrouve la méthode du point milieu

$$y_{n+1} = y_n + h_n f\big(t_n + \frac{h_n}{2}, y_n + \frac{h_n}{2}f(t_n, y_n)\big).$$

Exemple 3 : Méthode de Runge-Kutta classique *i.e.* pour $q = 4$:
il s'agit de la méthode définie par le tableau

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array}$$

L'algorithme correspondant s'écrit

$$\begin{cases} p_{n,1} = f(t_n, y_n) \\ t_{n,2} = t_n + \frac{1}{2}h_n \\ y_{n,2} = y_n + \frac{1}{2}h_n p_{n,1} \\ p_{n,2} = f(t_{n,2}, y_{n,2}) \\ y_{n,3} = y_n + \frac{1}{2}h_n p_{n,2} \\ p_{n,3} = f(t_{n,2}, y_{n,3}) \\ t_{n+1} = t_n + h_n \\ y_{n,4} = y_n + h_n p_{n,3} \\ p_{n,4} = f(t_{n+1}, y_{n,4}) \\ y_{n+1} = y_n + h_n\big(\frac{1}{6}p_{n,1} + \frac{2}{6}p_{n,2} + \frac{2}{6}p_{n,3} + \frac{1}{6}p_{n,4}\big) \end{cases}$$

On verra plus loin que cette méthode est d'ordre 4. Dans ce cas les méthodes d'intégration (M_i) et (M) utilisées sont respectivement :

(M_2) $\displaystyle\int_0^{\frac{1}{2}} g(t)dt \simeq \frac{1}{2}g(0)$ méthodes des rectangles à gauche,

(M_3) $\displaystyle\int_0^{\frac{1}{2}} g(t)dt \simeq \frac{1}{2}g(\frac{1}{2})$ méthodes des rectangles à droite,

(M_4) $\displaystyle\int_0^{1} g(t)dt \simeq g(\frac{1}{2})$ méthodes du point milieu,

(M) $\displaystyle\int_0^{1} g(t)dt \simeq \frac{1}{6}g(0) + \frac{2}{6}g(\frac{1}{2}) + \frac{2}{6}g(\frac{1}{2}) + \frac{1}{6}g(1)$ méthode de Simpson.

2.3.3 Stabilité des méthodes de Runge-Kutta

Les méthode de Runge-Kutta sont des méthodes à un pas

$$y_{n+1} = y_n + h_n\Phi(t_n, y_n, h_n),$$

avec $\Phi(t_n, y_n, h_n) = \sum_{1\le j\le q} b_j p_{n,j}$. La fonction Φ est définie de manière explicite par

$$\begin{cases} \Phi(t, y, h) = \sum_{1\le j\le q} b_j f(t + c_j h, y_j) \text{ avec} \\ y_i = y + h\sum_{1\le j<i} a_{ij} f(t + c_j h, y_j), \ \ 1 \le i \le q. \end{cases} \tag{2.9}$$

Supposons que f soit k-lipschitzienne en y (uniformément par rapport à $t \in [t_0, t_0 + T]$ et $h \in [0, h_{\max}]$). On va montrer que Φ est alors également lipschitzienne. Soit $z \in I\!R$ et supposons $\Phi(t, z, h)$ et z_i définis à partir de z comme dans la formule (2.9).

Lemme 2.3.1 *Soit* $\alpha = \max_i \big(\sum_{1\le j\le i} |a_{ij}| \big)$. *Alors*

$$|y_i - z_i| \le \big(1 + (\alpha k h) + (\alpha k h)^2 + ... + (\alpha k h)^{i-1}\big)|y - z|.$$

Preuve : Le lemme se démontre par récurrence sur i. Pour $i = 1$, on a $y_1 = y, z_1 = z$ et le résultat est évident. Supposons le résultat vrai pour $j < i$, alors

$$|y_i - z_i| \le |y - z| + h \sum_{j<i} |a_{ij}| k \max_{j<i} |y_j - z_j|,$$

$$|y_i - z_i| \le |y - z| + \alpha k h \max_{j<i} |y_j - z_j|.$$

Par hypothèse de récurrence il vient

$$\max_{j<i} |y_j - z_j| \le \big(1 + \alpha k h + ... + (\alpha k h)^{i-2}\big)|y - z|,$$

et l'inégalité s'ensuit à l'ordre i. ∎

La formule (2.9) entraîne maintenant

$$|\Phi(t, y, h) - \Phi(t, z, h)| \le \sum_{1\le j\le q} |b_j| k |y_j - z_j| \le \Lambda |y - z|,$$

avec

$$\Lambda = k \sum_{1\le j\le q} |b_j| \big(1 + (\alpha k h_{\max}) + ... + (\alpha k h_{\max})^{j-1}\big),$$

d'où le corollaire suivant

Corollaire 2.3.2 *Les méthodes de Runge-Kutta sont stables, avec constante de stabilité* $S = e^{\Lambda T}$.

2.3.4 Ordre des méthodes de Runge-Kutta

Pour déterminer l'ordre, on applique la Propositions 2.2.8 : l'ordre est au moins égal à p si et seulement si

$$\frac{\partial^\ell \Phi}{\partial h^\ell}(t, y, 0) = \frac{1}{\ell + 1} f^{[\ell]}(t, y), \quad \ell \le p - 1.$$

Grâce à l'expression (2.9), on a

$$\Phi(t, y, 0) = \sum_{1\le j\le q} b_j f(t, y) = f(t, y).$$

D'après le Théorème 2.2.3, les méthodes de Runge-Kutta sont donc toujours d'ordre ≥ 1, i.e. consistantes.

Lemme 2.3.3

$$\frac{\partial \Phi}{\partial h}(t,y,0) = \Big(\sum_{i=1}^{q} b_i c_i\Big) f^{[1]}(t,y)$$

Preuve : On a

$$\Phi(t,y,h) = \sum_{i=1}^{q} b_i f(t + c_i h, y_i(t,y,h))),$$

d'où

$$\frac{\partial \Phi}{\partial h}(t,y,h) = \sum_{i=1}^{q} b_i c_i \frac{\partial f}{\partial t}(t + c_i h, y_i(t,y,h)) + \sum_{i=1}^{q} b_i \frac{\partial y_i}{\partial h}\frac{\partial f}{\partial y}(t + c_i h, y_i(t,y,h)),$$

Par ailleurs

$$y_i(t,y,h) = y + h\sum_{j=1}^{i-1} a_{ij} f(t + c_j h, y_j(t,y,h)),$$

donc

$$\frac{\partial y_i}{\partial h}(t,y,h) = \sum_{j=1}^{i-1} a_{ij} f(t + c_j h, y_j(t,y,h)) + h\sum_{j=1}^{i-1} a_{ij}\frac{\partial}{\partial h}\Big(f(t + c_j h, y_j(t,y,h))\Big).$$

Puisque $y_j(t,y,0) = 0$, et grâce à (2.8) on obtient

$$\frac{\partial y_i}{\partial h}(t,y,0) = \Big(\sum_{j=1}^{i-1} a_{ij}\Big) f(t,y) = c_i f(t,y),$$

par conséquent

$$\begin{aligned}\frac{\partial \Phi}{\partial h}(t,y,0) &= \Big(\sum_{i=1}^{q} b_i c_i\Big)\Big(\frac{\partial f}{\partial t}(t,y) + f(t,y)\frac{\partial f}{\partial y}(t,y)\Big) \\ &= \Big(\sum_{i=1}^{q} b_i c_i\Big) f^{[1]}(t,y).\end{aligned}$$

■

Corollaire 2.3.4 *Les méthodes de Runge-Kutta sont d'ordre ≥ 2 si et seulement si* $\sum_{i=1}^{q} b_i c_i = \frac{1}{2}$.

Exemple 2.3.5

• On reprend le tableau de la méthode $(RK2)$: la méthode est d'ordre au moins 2 puisque $0 \times (1 - \frac{1}{2\alpha}) + \alpha \times \frac{1}{2\alpha} = \frac{1}{2}$.

• La méthode classique $(RK4)$ est d'ordre au moins 2 également, puisque $0\times\frac{1}{6}+\frac{1}{2}\times\frac{2}{6}\times 1\times\frac{1}{6} = \frac{1}{2}$.

$$\frac{\partial^2 \Phi}{\partial h^2}(t,y,h) = \sum_j b_j (c_j^2 \frac{\partial^2 f}{\partial t^2} + 2c_j \frac{\partial^2 f}{\partial t \partial y}\frac{\partial y_j}{\partial h} + \frac{\partial^2 f}{\partial y^2}(\frac{\partial y_j}{\partial h})^2 + \frac{\partial f}{\partial y}\frac{\partial^2 y_j}{\partial h^2}),$$

$$\frac{\partial^2 y_i}{\partial h^2} = 2\sum_{j<i} a_{ij}(c_j \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y_j}{\partial h}) + h\sum_{j<i} a_{ij}(c_j^2 \frac{\partial^2 f}{\partial t^2} + \ldots).$$

Pour $h = 0$, il vient

$$\frac{\partial^2 y_i}{\partial h^2}(t, y, 0) = 2\sum_{j<i} a_{ij} c_j (\frac{\partial f}{\partial t} + \frac{\partial f}{\partial y} f)(t, y),$$

$$\begin{aligned}\frac{\partial^2 \Phi}{\partial h^2}(t, y, 0) &= \sum_j b_j c_j^2 (\frac{\partial^2 f}{\partial t^2} + 2f\frac{\partial^2 f}{\partial t \partial y} + \frac{\partial^2 f}{\partial y^2} f^2)(t, y) \\ &+ 2\sum_{i,j} b_i a_{ij} c_j \frac{\partial f}{\partial y}(\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y})(t, y).\end{aligned}$$

Or $f^{[2]}$ est donné par

$$\begin{aligned}f^{[2]}(t, y) &= \frac{\partial f^{[1]}}{\partial t} + f\frac{f^{[1]}}{\partial y} \\ &= \frac{\partial}{\partial t}(\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}) + f\frac{\partial}{\partial y}(\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}) \\ &= \frac{\partial^2 f}{\partial t^2} + f\frac{\partial^2 f}{\partial t \partial y} + \frac{\partial f}{\partial y}\frac{\partial f}{\partial t} + f\frac{\partial^2 f}{\partial t \partial y} + f^2\frac{\partial^2 f}{\partial y^2} + f(\frac{\partial f}{\partial y})^2. \\ &= (\frac{\partial^2 f}{\partial t^2} + 2f\frac{\partial^2 f}{\partial t \partial y} + f^2\frac{\partial^2 f}{\partial y^2}) + \frac{\partial f}{\partial y}(\frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}).\end{aligned}$$

La condition $\frac{\partial^2 \Phi}{\partial h^2}(t, y, 0) = \frac{1}{3} f^{[2]}(t, y)$ se traduit en général par les conditions $\sum_j b_j c_j^2 = \frac{1}{3}$, et $\sum_{i,j} b_i a_{ij} c_j = \frac{1}{6}$. Un calcul analogue (pénible !) de $\dfrac{\partial^3 \Phi}{\partial h^3}$ conduirait au résultat suivant :

Théorème 2.3.6 *Les méthodes de Runge-Kutta définies par les tableaux de coefficients c_i, a_{ij}, b_j sont*

- *d'ordre ≥ 2 ssi*

$$\sum_{i=1}^{q} b_i c_i = \frac{1}{2}.$$

- *d'ordre ≥ 3 ssi*

$$\sum_{i=1}^{q} b_i c_i = \frac{1}{2};\ \sum_{i=1}^{q} b_i c_i^2 = \frac{1}{3};\ \sum_{i,j} b_i a_{ij} c_j = \frac{1}{6}.$$

- *d'ordre ≥ 4 ssi*

$$\sum_{i=1}^{q} b_i c_i = \frac{1}{2};\ \sum_{i=1}^{q} b_i c_i^2 = \frac{1}{3};\ \sum_{i=1}^{q} b_i c_i^3 = \frac{1}{3};\ \sum_{i,j} b_i a_{ij} c_j = \frac{1}{6};$$

$$\sum_{i,j} b_i a_{ij} c_j^2 = \frac{1}{12} \ \sum_{i,j} b_i c_i a_{ij} c_j = \frac{1}{8}; \sum_{i,j} b_i a_{ij} a_{jk} c_k = \frac{1}{12}$$

2.4 Méthodes adaptatives

Les schémas de Runge Kutta étant des méthodes à un pas, ils se prêtent bien à des techniques d'adaptation du pas de temps h, à condition que l'on dispose d'un estimateur efficace de l'erreur locale. L'estimateur d'erreur peut être construit de deux méthodes : la première consiste à utiliser la même méthode RK, mais avec deux pas de discrétisation différents, la deuxième utilise deux méthodes RK d'ordre de convergence différent. L'idéal est d'avoir deux méthodes d'ordres différents mais qui évaluent la fonction f aux mêmes endroits. Ainsi le nombre d'évaluations reste petit même si l'on emploie deux méthodes simultanément.

Principe :

Le principe de la deuxième méthode est d'ajuster le pas localement pour obtenir une précision imposée. Estimer la quantité $\eta_{n+1} := \frac{y(t_{n+1})-y_{n+1}}{h}$ (appelée l'erreur de trancature locale) à l'aide de l'écart entre deux solutions numériques : y_n d'une méthode d'ordre p et y_n^* d'une méthode d'ordre p^* tel que $p^* > p$. On a

$$\eta_{n+1} = \frac{y(t_{n+1}) - y_{n+1}}{h},$$

et

$$\eta_{n+1}^* = \frac{y(t_{n+1}) - y_{n+1}^*}{h}.$$

Par soustraction on obtient

$$\eta_{n+1} = \eta_{n+1}^* + \frac{y_{n+1}^* - y_{n+1}}{h}.$$

Comme $\eta_{n+1} = O(h^p)$ et $\eta_{n+1}^* = O(h^{p^*})$, nous pouvons négliger η_{n+1}^* et obtenir l'estimation d'erreur suivante

$$\eta_{n+1} \approx \frac{y_{n+1}^* - y_{n+1}}{h},$$

avec modification du pas d'un facteur δ on obtient

$$\eta_{n+1}(h\delta) \approx \delta^p \eta_{n+1}(h) \approx \frac{\delta^p}{h}(y_{n+1}^* - y_{n+1}).$$

Pour obtenir une précision ε donnée on prend $\eta_{n+1}(h\delta) \leq \varepsilon$. D'où le facteur à appliquer au pas est

$$\delta \leq \Big(\frac{\varepsilon h}{y_{n+1}^* - y_{n+1}}\Big)^{\frac{1}{p}}.$$

2.4.1 Méthode de Runge Kutta Fehlberg

Une méthode très efficace a été développée en utilisant une paire de méthodes de Runge Kutta d'ordres 4 et 5 (notée $rk45$) et requérant 6 évaluations de la fonction f au lieu de 12 évaluations (si on utilise la première méthode). Si l'estimation d'erreur de troncature locale $|\eta|$ est inférieure à une tolérance fixée ε, on passe au pas de temps suivant. Sinon, l'estimation est répétée avec un pas de temps plus petit. Si l'estimation d'erreur est beaucoup plus petite que ε, alors on passe à l'étape suivante, en prenant un pas de temps plus grand. L'algorithme pour cette méthode est donné par

$$
\begin{aligned}
&y_0 = \alpha \\
&k_1 = hf(t_i, y_i) \\
&k_2 = hf(t_i + \tfrac{h}{4}, y_i + \tfrac{k_1}{4}) \\
&k_3 = hf(t_i + \tfrac{3h}{8}, y_i + \tfrac{3}{32}k_1 + \tfrac{9}{32}k_2) \\
&k_4 = hf(t_i + \tfrac{12h}{13}, y_i + \tfrac{1932}{2197}k_1 - \tfrac{7200}{2197}k_2 + \tfrac{7296}{2197}k_3) \\
&k_5 = hf(t_i + h, y_i + \tfrac{439}{216}k_1 - 8k_2 + \tfrac{3680}{513}k_3 - \tfrac{845}{4104}k_4) \\
&k_6 = hf(t_i + \tfrac{h}{2}, y_i - \tfrac{8}{27}k_1 + 2k_2 - \tfrac{3544}{2565}k_3 + \tfrac{1859}{4104}k_4 - \tfrac{11}{40}k_5) \\
&y_{i+1} = y_i + \tfrac{25}{216}k_1 + \tfrac{1408}{2565}k_3 + \tfrac{2197}{4104}k_4 - \tfrac{1}{5}k_5 \\
&y^*_{i+1} = y_i + \tfrac{16}{135}k_1 + \tfrac{6656}{12825}k_3 + \tfrac{28561}{56430}k_4 - \tfrac{9}{50}k_5 + \tfrac{2}{55}k_6 \\
&\eta = \tfrac{1}{h}|y^*_{i+1} - y_{i+1}| \\
&\delta = 0.84(\tfrac{\varepsilon}{\eta})^{\frac{1}{4}}
\end{aligned}
$$

-Si $\eta \leq \varepsilon$ on passe à l'étape suivante en prenant $h = \delta h$. Pour ce cas le δ peut être strictement inférieur à 1 comme il peut être strictement supérieur à 1. En effet :

- Si $\eta < \dfrac{\varepsilon}{2}$ (on note $\eta \ll \varepsilon$) alors $\eta < (0.84)^4\varepsilon$, i.e $1 < \delta$.
- Si $\dfrac{\varepsilon}{2} < \eta \leq \varepsilon$ alors $\delta < 1$.

-Si $\eta > \varepsilon$ recalculons les étapes actuelles avec $h = \delta h$, et dans ce cas, il est clair que $\delta < 1$.

Exemple 2.4.1 *Soit le problème de Cauchy suivant*

$$
\begin{cases}
y' = y - t^2 + 1, \;\; t \in [0, 2] \\
y(0) = 0.5
\end{cases}
\tag{2.10}
$$

La solution exacte du problème (2.10) est donnée par

$$
y(t) = t^2 + 2t + 1 - \frac{1}{2}e^t.
$$

On résout ce problème par la méthode de Runge Kutta Fehlberg avec une précision $\varepsilon = 0.00001$, *les résultats sont donnés dans le tableau suivant*

k	t_k	y_k *la solution approchée par* $rk45$	$y(t_k)$ *la solution exacte*
0	0.0000	0.5000	0.50000
1	0.2000	0.8293	0.8293
2	0.4353	1.2874	1.2874
3	0.6766	1.8274	1.8274
4	0.9264	2.4483	2.4483
5	1.1902	3.1530	3.1531
6	1.4806	3.9556	3.9556
7	1.8537	4.9520	2.2620
8	2.0000	5.3055	5.3055

Puis on résout le problème (2.10) par la méthode de Runge Kutta d'ordre 4, *les résultats sont donnés dans le tableau suivant*

k	t_k	y_k *la solution approchée par* $rk4$	$y(t_k)$ *la solution exacte*
0	0.0000	0.5000	0.5000
1	0.2000	0.8293	0.8293
2	0.4000	1.2141	1.2141
3	0.6000	1.6489	1.6489
4	0.8000	2.1272	2.1272
5	1.0000	2.6408	2.6409
6	1.2000	3.1799	3.1799
7	1.4000	3.7323	3.7324
8	1.6000	4.2834	4.2835
9	1.8000	4.8151	4.8152
10	2.0000	5.3054	5.3055

D'après les résultats précédents on voit bien que la méthode de Runge Kutta Fehlberg est plus précise et rapide que la méthode de Runge Kutta d'ordre 4.

Chapitre 3

Méthodes à pas multiples

Comme dans le chapitre précédent, on s'intéresse à la résolution numérique du problème de Cauchy suivant

$$\begin{cases} y' = f(t,y) \\ y(0) = y_c. \end{cases} \tag{3.1}$$

avec $(t,y) \in [t_0, t_0+T] \times I\!R$ et $y_c \in I\!R$. On supposera f continue et Lipschitzienne par rapport à y de rapport L. Etant donné un maillage régulier $t_n = t_0 + nh$, $0 \leq n \leq N$, $h = \frac{T}{N}$ du segment $[t_0, t_0+T]$, et un entier $r \geq 1$. Une méthode à r pas est définie par le schéma

$$y_{n+1} = \sum_{i=0}^{r-1} a_i y_{n-i} + h \sum_{i=0}^{r} b_i f(t_{n-i+1}, y_{n-i+1}), \;\; n \geq r-1. \tag{3.2}$$

Si on pose $a_i = \frac{\alpha_{r-1-i}}{\alpha_r}$, $i = 0, ..., r-1$, et $b_i = \frac{\beta_{r-i}}{\alpha_r}$, $i = 0, ..., r$, $\alpha_r \neq 0$, on obtient la forme générale des méthodes à r pas suivante :

$$\begin{cases} \sum\limits_{i=0}^{r} \alpha_i y_{n+i} = h \sum\limits_{i=0}^{r} \beta_i f_{n+i}; \;\; n = 0, ..., N-r; \\ y_i = y_{ic}, \;\; 0 \leq i \leq r-1. \end{cases} \tag{3.3}$$

où les constantes α_i et β_i, $0 \leq i \leq r$ sont indépendantes de n. On a posé $f_{n+i} = f(t_{n+i}, y_{n+i})$ et on supposera $\alpha_r = 1$.

- Si $\beta_r = 0$, nous pouvons obtenir directement de l'algorithme (3.3) la valeur de y_{n+r} en fonction de $y_{n+r-1}, ..., y_n$; et la méthode est dite **explicite.**
- Si $\beta_r \neq 0$, la méthode est dite **implicite**, car elle définie y_{n+r} par une équation implicite donnée par

$$y_{n+r} = h\beta_r f(t_{n+r}, y_{n+r}) + \sum_{i=0}^{r-1} (h\beta_i f(t_{n+i}, y_{n+i}) - \alpha_i y_{n+i}). \tag{3.4}$$

Les valeurs initiales $y_{ic}, 0 \leq i \leq r-1$ doivent être calculées séparément par une procédure de démarrage adéquate, à partir de la donnée de Cauchy y_c, on dit dans ce cas que les méthodes à pas multiples ne sont pas **auto-démarrante**.
Pour régler le carractère implicite de (3.4) on peut utiliser l'algorithme du point fixe (la méthode des approximations successives) comme l'annonce le théorème suivant :

Théorème 3.0.2 *Quand $\beta_r \neq 0$, l'équation (3.4) admet une solution unique si $h < \dfrac{1}{L|\beta_r|}$. La suite $(y_{n+r}^{(k)})_{k\in\mathbb{N}}$ définie par la méthode itérative*

$$y_{n+r}^{(k+1)} = h\beta_r f(t_{n+r}, y_{n+r}^{(k)}) + \sum_{i=0}^{r-1}(h\beta_i f_{n+i} - \alpha_i y_{n+i}). \tag{3.5}$$

converge vers y_{n+r} lorsque k tend vers l'infini.

Preuve :
Si $\beta_r \neq 0$, on a

$$y_{n+r} = h\beta_r f(t_{n+r}, y_{n+r}) + \sum_{i=0}^{r-1}(h\beta_i f(t_{n+i}, y_{n+i}) - \alpha_i y_{n+i}).$$

Si nous supposons connue les valeurs $y_{n+r-1}, ..., y_n$, et les $f_{n+r-1}, ..., f_n$, le deuxième terme de cette somme est une constante que nous noterons C. Donc y_{n+r} est solution de l'équation

$$y = h\beta_r f(t_{n+r}, y) + C.$$

Soit la fonction

$$\varphi : y \to \varphi(y) = h\beta_r f(t_{n+r}, y) + C,$$

comme f est lipschitzienne, $\forall y, y^* \in \mathbb{R}$:

$$\begin{aligned} |\varphi(y) - \varphi(y^*)| &\leq h|\beta_r||f(t_{n+r}, y) - f(t_{n+r}, y^*)| \\ &\leq hL|\beta_r||y - y^*|. \end{aligned}$$

Donc φ contactante dés que $h < \dfrac{1}{L|\beta_r|}$, par conséquent, y_{n+r} est l'unique point fixe de φ d'après le théorème du point fixe. De plus la suite $(y_{n+r}^{(k)})_{k\in\mathbb{N}}$ définie par

$$y_{n+r}^{(k+1)} = h\beta_r f(t_{n+r}, y_{n+r}^{(k)}) + \sum_{i=0}^{r-1}(h\beta_i f_{n+i} - \alpha_i y_{n+i})$$

converge vers l'unique point fixe y_{n+r} lorsque k tend vers l'infini. ∎

Les méthodes à pas multiples implicites sont donc bien plus coûteuses en temps de calcul et plus complexes à mettre en oeuvre que leurs analogues explicites. Il est cependant possible de diminuer le nombre d'itérations dans (3.5) en choisissant judicieusement la valeur $y_{n+r}^{(0)}$. On peut par exemple, effectuer une étape d'une méthode à pas multiples explicite de même ordre et poursuivre les itérations de la méthode implicite à partir de la valeur obtenue ; c'est le type de stratégie retenue par les méthodes de prédiction correction que nous présenterons dans la section 3.3.

3.1 Notions d'ordre, de consistance, de stabilité et de convergence des méthodes à pas multiples

Pour une méthode à pas multiples de la forme (3.3), l'erreur de consistance prend la forme

$$\begin{aligned} e_{n+r} &= y(t_{n+r}) - \overline{y}_{n+r} \\ &= \sum_{i=0}^{r} (\alpha_i y(t_{n+i}) - h\beta_i f(t_{n+i}, y(t_{n+i}))) \\ &= \sum_{i=0}^{r} (\alpha_i y(t_{n+i}) - h\beta_i y'(t_{n+i})),\ n = 0, .., N-r, \end{aligned}$$

où $\overline{y}_{n+r}$ est l'approximation fournie par la méthode en supposant que $y_{n+i} = y(t_{n+i})$, $i = 0, .., r-1$.

Définition 3.1.1 *Le schéma à r pas* (3.3) *est consistant avec l'équation* (3.1) *si*

$$\lim_{h \to 0} |e_{n+r}| = 0,\ \ \forall\ 0 \leq n \leq N - r.$$

Il est dans ce cas commode d'introduire l'opérateur $\mathcal{L}$ que l'on définit, pour toute fonction arbitraire z de classe $\mathcal{C}^1$ sur l'intervalle $[t_0, t_0 + T]$, par

$$\mathcal{L}(z(t), h) = \sum_{i=0}^{r} (\alpha_i z(t + ih) - h\beta_i z'(t + ih)), \tag{3.6}$$

l'erreur de consistance s'écrit alors

$$e_{n+r} = \mathcal{L}(y(t_n), h),\ \ n = 0, ..., N - r. \tag{3.7}$$

On peut voir cet opérateur comme un opérateur linéaire agissant sur toute fonction différentiable.

Définition 3.1.2 *Une méthode à r pas est dite d'ordre p s'il existe une constante $C > 0$ indépendante de h telle que*

$$|e_{n+r}| \leq Ch^{p+1},\ 0 \leq n \leq N - r$$

Supposons à présent la fonction z infiniment différentiable. En effectuant des développements de Taylor au point t de $z(t+ih)$ et $z'(t+ih)$, $i = 0, ..., r$, dans (3.6) et en regroupant les termes, on obtient

$$\mathcal{L}(z(t), h) = C_0 z(t) + C_1 h z'(t) + ... + C_k h^k z^{(k)}(t) + ... \tag{3.8}$$

où

$$\begin{aligned} C_0 &= \sum_{i=0}^{r} \alpha_i, \\ C_1 &= \sum_{i=0}^{r} (i\alpha_i - \beta_i), \\ C_k &= \sum_{i=0}^{r} (\frac{i^k}{k!}\alpha_i - \frac{i^{k-1}}{(k-1)!}\beta_i),\ k \geq 2. \end{aligned}$$

Ceci conduit à la définition suivante :

Définition 3.1.3 caractérisation de l'ordre d'une méthode à pas multiples
Une méthode à pas multiples est d'ordre p, avec p un entier naturel, si l'opérateur $\mathcal{L}$, défini par (3.6), *qui lui est associé est tel que l'on a $C_0 = C_1 = ... = C_p = 0$ et $C_{p+1} \neq 0$.*

Définition 3.1.4 *On dit que La méthode* (3.3) *est stable s'il existe une constante S, indépendante de h, telle que pour $y_{id}, z_{id} \in \mathbb{R}, 0 \leq i \leq r-1$, pour $\varepsilon_n \in \mathbb{R}$, $\forall\ 0 \leq n \leq N - r$, les suites y_n et z_n définies par :*

$$\sum_{i=0}^{r} \alpha_i y_{n+i} = h \sum_{i=0}^{r} \beta_i f_{n+i},\ y_i = y_{id},\ 0 \leq i \leq r-1,$$

$$\sum_{i=0}^{r} \alpha_i z_{n+i} = h \sum_{i=0}^{r} \beta_i \widetilde{f}_{n+i} + \varepsilon_k,\ z_i = z_{id},\ 0 \leq i \leq r-1.$$

Avec $f_i = f(t_i, y_i)$ et $\widetilde{f}_i = f(t_i, z_i)$, vérifient :

$$\max_{0 \leq n \leq N-r} |y_n - z_n| \leq S(\max_{0 \leq i \leq r-1} |y_{id} - z_{id}| + \sum_{n=0}^{N-r} |\varepsilon_n|).$$

Définition 3.1.5 *Une méthode à pas multiples est dite convergente si*

$$\lim_{h\to 0} |y_n - y(t_n)| = 0, \forall\ 0 \leq n \leq N$$

dès que les valeurs d'initialisation $y_i, i = 0, ..., r-1,$ *satisfont*

$$\lim_{h\to 0} |y_{ic} - y_c| = 0,\ \ i = 0, ..., r-1.$$

Théorème 3.1.6 *Si la méthode à* r *pas est consistante et stable, alors elle est convergente.*

La preuve est analogue à celle des méthodes à un pas.

3.2 Schémas d'Adams

Si y est une solution exacte du problème (3.1) alors

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(t, y(t))dt.$$

Les schémas d'Adams approchent l'intégrale $\displaystyle\int_{t_n}^{t_{n+1}} f(t, y(t))dt$ par l'intégrale d'un polynôme P interpolant f en des points donnés qui peuvent être à l'extérieur de l'intervalle $[t_n; t_{n+1}]$. On peut construire différentes schémas selon les points d'interpolation choisis. Ils se divisent en deux familles : les méthodes d'Adams Bashforth qui sont explicites et les méthodes d'Adams-Moulton qui sont implicites. Voici quelques exemples :

3.2.1 Méthodes d'Adams-Bashforth

On ne suppose plus ici que le pas h_n soit nécessairement constant.

Principe :

Il s'agit d'interpoler f aux points $t_n, t_{n-1}, \ldots, t_{n-r}$, où $r \geq 0$ est fixé, par un polynôme noté $P_{n,r}$, c'est-à-dire l'unique polynôme de degré r vérifiant

$$P_{n,r}(t_{n-i}) = f(t_{n-i}, y(t_{n-i})),\ \ 0 \leq i \leq r.$$

Si l'on écrit ce polynôme dans la base de Lagrange, on a

$$P_{n,r}(t) = \sum_{i=0}^{r} f(t_{n-i}, y(t_{n-i}))L_{n,i,r}(t),$$

où $L_{n,i,r}(t) = \prod_{\substack{0\leq j\leq r\\ j\neq i}} \frac{t-t_{n-j}}{t_{n-i}-t_{n-j}}$.
Par conséquent

$$\begin{aligned} y(t_{n+1}) &= y(t_n) + \int_{t_n}^{t_{n+1}} f(t, y(t))dt \\ &\simeq y(t_n) + \int_{t_n}^{t_{n+1}} P_{n,r}(t)dt \\ &= y(t_n) + h_n \sum_{i=0}^{r} b_{n,i,r} f(t_{n-i}, y(t_{n-i})), \end{aligned}$$

avec

$$b_{n,i,r} = \frac{1}{h_n} \int_{t_n}^{t_{n+1}} L_{n,i,r}(t)dt.$$

On obtient donc le schéma suivant, appelé schéma d'Adams-Bashforth à $r+1$ pas, noté AB_{r+1} :

$$y_{n+1} = y_n + h_n \sum_{i=0}^{r} b_{n,i,r} f(t_{n-i}, y_{n-i}).$$

Voici quelques schémas d'Adams-Bashforth correspondant à quelques valeurs de r.
-Pour $r = 0$: on a

$$P_{n,0}(t) = f(t_n, y(t_n)),$$
$$\int_{t_n}^{t_{n+1}} P_{n,0}(t)dt = h_n f(t_n, y(t_n)).$$

On obtient le schéma

$$\begin{cases} y_{n+1} = y_n + h_n f(t_n, y_n), \ \ n \geq 0 \\ y_0 \text{ donné.} \end{cases}$$

La méthode AB_1 coïncide donc avec la méthode d'Euler explicite.
-Pour $r = 1$: l'algorithme d'Adams-Bashforth à 2 pas s'écrit

$$y_{n+1} = y_n + h_n\Big(f_n + \frac{h_n}{2h_{n-1}}(f_n - f_{n-1})\Big), \ \ n \geq 1$$

où $f_n := f(t_n, y_n)$.
-Pour $r = 2$: dans le cas où le pas de temps $h_n = h$ est constant, l'algorithme d'Adams Bashforth à 3 pas s'écrit

$$y_{n+1} = y_n + \frac{h}{12}\Big(23f_n - 16f_{n-1} + 5f_{n-2}\Big), \ \ n \geq 2$$

On donne le tableau suivant pour les petites valeurs de r et les coefficients $b_{n,i,r}$ ($h_n = h$) correspondants sont aussi donnés :

r	$b_{0,r}$	$b_{1,r}$	$b_{2,r}$	$b_{3,r}$	$\cdots$	$\lambda_r = \sum_{i=0}^{r} \lvert b_{i,r} \rvert$
0	1					1
1	$\frac{3}{2}$	$\frac{-1}{2}$				2
2	$\frac{23}{12}$	$\frac{-16}{12}$	$\frac{5}{12}$			$3,66\cdots$
3	$\frac{55}{24}$	$\frac{-59}{24}$	$\frac{37}{24}$	$\frac{-9}{24}$		$6,6\cdots$

3.2.2 Erreur de consistance et ordre des méthodes AB_{r+1}

Soit y une solution exacte du problème de Cauchy (3.1). L'erreur de consistance à l'étape n est donnée par

$$e_n = y(t_{n+1}) - \overline{y}_{n+1} = y(t_{n+1}) - (y(t_n) + \int_{t_n}^{t_{n+1}} P_{n,r}(t)dt) = \int_{t_n}^{t_{n+1}} (y'(t) - P_{n,r}(t))dt.$$

Où $P_{n,r}$ est précisément le polynôme d'intérpolation de la fonction $y' = f$ au points t_{n-i}, $0 \leq i \leq r$. D'après le théorème de la moyenne (voir annexe), il existe un point $\theta \in]t_n, t_{n+1}[$ tel que $e_n = h_n(y'(\theta) - P_{n,r}(\theta))$. La formule donnant l'erreur d'interpolation donne :

$$y'(\theta) - P_{n,r}(\theta) = \frac{1}{(r+1)!} y^{(r+2)}(\xi)\pi_{n,r}(\xi),$$

où $\xi \in]t_{n-r}, t_{n+1}[$ est un point intermédiaire entre θ et les points t_{n-i}, et

$$\pi_{n,r}(t) = \prod_{0 \leq i \leq r} (t - t_{n-i}),$$

où $\xi \in]t_{n-j}, t_{n-j+1}[$, $0 \leq j \leq r$. On a l'inégalité

$$\mid \xi - t_{n-i} \mid \leq (1 + \mid j - i \mid) h_{max}, \forall 0 \leq i \leq r, \forall 0 \leq j \leq r,$$

D'où

$$\begin{aligned} \mid \pi_{n,r}(\xi) \mid &= \prod_{i=0}^{j} (\xi - t_{n-i}) \prod_{i=j+1}^{r} (\xi - t_{n-i}) \\ &\leq h_{max}^{r+1}(1+j)\cdots(1+1)1(1+1)\cdots(1+r-j) \\ &= h_{max}^{r+1}(j+1)!(r-j+1)! \\ &\leq h_{max}^{r+1}(r+1)! \end{aligned} \tag{3.9}$$

En majorant 2 pas $j+2, ..., (r-j+1)$ par $(r+1)$.
On en déduit par conséquent

$$|y'(\theta) - P_{n,r}(\theta)| \leq |y^{r+2}(\xi)| h_{max}^{r+1},$$

ce qui donne la majoration cherchée de l'erreur de consistance :

$$|e_n| \leq |y^{r+2}(\xi)| h_n h_{max}^{r+1} \leq C h_n h_{max}^{r+1}$$

La méthode d'Adams Bashforth à $r+1$ pas est donc d'ordre $r+1$.

3.2.3 Stabilité de la méthode d'AB$_{r+1}$:

Théorème 3.2.1 *On suppose que f est uniformément k lipschitzienne par apport à y, et que les sommes $\sum_{i=0}^{r} |b_{n,i,r}|$ sont majorées indépendamment de n par une constante λ_r, alors la méthode d'AB_{r+1} est stable avec une constante de stabilité $S = e^{\lambda_r k T}$.*

Preuve :
Soit $\widetilde{y}_n$ la suite récurrente perturbée telle que :

$$\begin{cases} \widetilde{y}_{n+1} = \widetilde{y}_n + h_n \sum_{i=0}^{r} b_{n,i,r} \widetilde{f}_{n-i} + \varepsilon_n. \\ \widetilde{f}_{n-i} = f(t_{n-i}, \widetilde{y}_{n-i}). \end{cases}$$

Avec $\theta_n := \max_{0 \leq i \leq n} \mid y_i - \widetilde{y}_i \mid$. On a $|f_{n-i} - \widetilde{f}_{n-i}| \leq k|y_i - \widetilde{y}_i| \leq k\theta_n$. Si on suppose que $h_n = h$ constant on trouve

$$\begin{aligned} |y_{n+1} - \widetilde{y}_{n+1}| &\leq |y_n - \widetilde{y}_n| + kh \sum_{i=0}^{r} |b_{n,i,r}| \theta_n + |\varepsilon_n|. \\ &\leq \theta_n (1 + kh\lambda_r) + |\varepsilon_n|. \end{aligned}$$

Comme $\theta_{n+1} = \max(\theta_n, |y_{n+1} - \widetilde{y}_{n+1}|)$, on en déduit

$$\theta_{n+1} \leq \theta_n (1 + kh\lambda_r) + |\varepsilon_n|.$$

Le lemme de Gronwall implique alors

$$\theta_n = \max_{0 \leq i \leq n} |y_i - \widetilde{y}_i| \leq e^{Tk\lambda_r} |y_0 - \widetilde{y}_0| + e^{Tk\lambda_r} \sum_{i=0}^{N-1} |\varepsilon_i|.$$

Donc la méthode est stable avec constante de stabilité $S = e^{Tk\lambda_r}$.

Remarque 3.2.2 *D'après le tableau on voit que la constante λ_r (quand h est constant) croit vite avec r. La stabilité devient donc de moins en moins bonne et c'est l'un des inconvénients les plus sérieux des méthodes d'Adams Bashforth lorsque r est grand. On pratique, on se limite le plus souvent au cas $r = 1$ ou 2.*

3.2.4 Schémas d'Adams-Moulton

Nous allons procéder de la même manière que pour les schémas d'Adams-Bashforth. Cette fois-ci cependant, nous allons approcher f par l'unique polynôme noté ici $P^*_{n,r}$ de degré $r+1$ vérifiant

$$P^*_{n,r}(t_{n-i}) = f(t_{n-i}, y(t_{n-i})),\ -1 \leq i \leq r.$$

Dans la base de Lagrange, $P^*_{n,r}$ s'écrit

$$P^*_{n,r}(t) = \sum_{i=-1}^{r} f(t_{n-i}, y(t_{n-i}))L^*_{n,i,r}(t),$$

où $L^*_{n,i,r}$ sont les polynômes de Lagrange associés aux noeuds $t_{n-i},\ -1 \leq i \leq n$, *i.e.*

$$L^*_{n,i,r}(t) = \prod_{\substack{-1\leq j\leq r\\ j\neq i}} \frac{t - t_{n-j}}{t_{n-i} - t_{n-j}}.$$

On remarque donc que l'on inclut $y(t_{n+1})$ (quand $i = -1$) dans les valeurs "connues", c'est pourquoi nous obtenons ainsi une famille de schémas multi-pas **implicites** appelés schémas d'Adams-Moulton à $r+1$ pas, notés AM_{r+1}

$$y_{n+1} = y_n + h_n \sum_{i=-1}^{r} b^*_{n,i,r} f_{n-i},\ n \geq r$$

avec

$$b^*_{n,i,r} = \frac{1}{h_n} \int_{t_n}^{t_{n+1}} L^*_{n,i,r}(t)dt.$$

Reste à savoir comment régler le caractère implicite de cette méthode.
Notons u_n la quantité explicite

$$u_n = y_n + h_n \sum_{i=0}^{r} b^*_{n,i,r} f_{n-i}.$$

La valeur y_{n+1} cherchée est donc la solution de l'équation

$$x = u_n + h_n b^*_{n,-1,r} f(t_{n+1}, x).$$

On va donc calculer la suite itérée $x_{p+1} = \varphi(x_p)$ avec

$$\varphi(x) = u_n + h_n b^*_{n,-1,r} f(t_{n+1}, x).$$

Comme $\varphi'(x) = h_n b^*_{n,-1,r} f'_y(t_{n+1}, x)$, et si f est k lipschitzienne en y, il suffit que $h_n < \dfrac{1}{k|b^*_{n,-1,r}|}$ pour que $\varphi'(x) < 1$ et par conséquent d'après le théorème des accroissements finis φ contractante. La solution y_{n+1} est alors unique d'après le théorème du point fixe et l'algorithme itératif

$$x_{p+1} = u_n + h_n b_{n,-1,r} f(t_{n+1}, x_p).$$

converge vers l'unique point fixe *i.e.* y_{n+1}. On choisira une valeur initiale x_0 qui soit une approximation de y_{n+1}, par exemple la valeur donnée par la méthode d'Adams-Bashforth :

$$x_0 = y_n + h_n \sum_{i=0}^{r} b^*_{n,i,r} f_{n-i}.$$

On arrête l'itération pour $|x_{p+1} - x_p| \leq \varepsilon$, où ε est la précision donnée, et on prend y_{n+1} la dernière valeur x_p calculée.

Exemple 3.2.3

• Pour $r = 0$: le polynôme $P^*_{n,0}$ est le polynôme de degré 1 qui interplole f en t_n et t_{n+1}, soit

$$P^*_{n,0}(t) = f_n + \frac{f_{n+1} - f_n}{h_n}(t - t_n),$$

$$\int_{t_n}^{t_{n+1}} P^*_{n,0}(t)dt = h_n\big(\frac{1}{2}f_{n+1} + \frac{1}{2}f_n\big).$$

On obtient ainsi la méthode dite des trapèzes (ou méthode de Crank-Nicolson) :

$$y_{n+1} = y_n + h_n\big(\frac{1}{2}f_{n+1} + \frac{1}{2}f_n\big)$$

• Pour $r = 1$: le polynôme $P^*_{n,1}$ interplole f en t_{n-1}, t_n et t_{n+1}, soit

$$P^*_{n,1}(t) = f_{n+1}\frac{(t-t_n)(t-t_{n-1})}{h_n(h_n+h_{n-1})} - f_n\frac{(t-t_{n-1})(t-t_{n-1})}{h_n h_{n-1}} + f_{n-1}\frac{(t-t_n)(t-t_{n+1})}{h_{n-1}(h_n+h_{n-1})},$$

et

$$\begin{aligned} y_{n+1} &= y_n + \int_{t_n}^{t_{n+1}} P^*_{n,1}(t)dt \\ &= y_n + h_n\Big(\frac{2h_n+3h_{n-1}}{6(h_n+h_{n-1})}f_{n+1} + \frac{h_n+3h_{n-1}}{6h_{n-1}}f_n - \frac{h_n^2}{6h_{n-1}(h_n+h_{n-1})}f_{n-1}\Big). \end{aligned}$$

Dans le cas où la pas $h_n = h$ est constant, cette formule se réduit à

$$y_{n+1} = y_n + h\big(\frac{5}{12}f_{n+1} + \frac{8}{12}f_n - \frac{1}{12}f_{n-1}\big).$$

On donne le tableau suivant pour les petites valeurs de r et les coeficients $b^*_{n,i,r}$ ($h_n = h$) correspondants sont aussi donnés :

r	$b^*_{-1,r}$	$b^*_{0,r}$	$b^*_{1,r}$	$b^*_{2,r}$	$b^{3,r*}$	$\cdots$	$\lambda^*_r = \sum_{i=-1}^{r} \lvert b^*_{i,r}\rvert$
0	$\frac{1}{2}$	$\frac{1}{2}$					1
1	$\frac{5}{12}$	$\frac{8}{12}$	$\frac{-1}{12}$				$1,16\cdots$
2	$\frac{9}{24}$	$\frac{19}{24}$	$\frac{-5}{24}$	$\frac{1}{24}$			$1,41\cdots$
3	$\frac{251}{720}$	$\frac{646}{720}$	$\frac{-264}{720}$	$\frac{106}{720}$	$\frac{-19}{720}$		$1,78\cdots$

3.2.5 Erreur de consistance et ordre de la méthode AM_{r+1}

Soit y une solution exacte du problème de Cauchy (3.1). Sous l'hypothèse $y_{n-i} = y(t_{n-i})$, $0 \leq i \leq r$, l'erreur de consistance est donnée par

$$\begin{aligned} e_n &= y(t_{n+1}) - \overline{y}_{n+1} \\ &= y(t_{n+1}) - (y(t_n) + h_n \sum_{0 \leq i \leq r} b^*_{n,i,r} f(t_{n-i}, y(t_{n-i})) + h_n b^*_{n,-1,r} f(t_{n+1}, y_{n+1})) \\ &= y(t_{n+1}) - (y(t_n) + h_n \sum_{-1 \leq i \leq r} b^*_{n,i,r} f(t_{n-i}, y(t_{n-i}))) \\ &+ h_n b^*_{n,-1,r}(f(t_{n+1}, y(t_{n+1})) - f(t_{n+1}, y_{n+1})). \end{aligned}$$

$$e_n = \int_{t_n}^{t_{n+1}} (y'(t) - P^*_{n,r}(t))dt + h_n b^*_{n,-1,r}(f(t_{n+1}, y(t_{n+1})) - f(t_{n+1}, y_{n+1})).$$

Supposons que $f(t, y)$ soit lipschitzienne de rapport k en y. Alors il vient

$$\begin{aligned} |e_n| &\leq |\int_{t_n}^{t_{n+1}} (y'(t) - P^*_{n,r}(t))dt| + h_n b^*_{n,-1,r} k |e_n| \\ &\leq \frac{1}{1 - h_n b^*_{n,-1,r} k} |\int_{t_n}^{t_{n+1}} (y'(t) - P^*_{n,r}(t))dt|. \end{aligned}$$

Quand le pas h_n est suffisamment petit, on a donc

$$|e_n| \leq |\int_{t_n}^{t_{n+1}} (y'(t) - P^*_{n,r}(t))dt|(1 + O(h_n)).$$

Par ailleurs la formule de la moyenne donne

$$\int_{t_n}^{t_{n+1}} (y'(t) - P^*_{n,r}(t))dt = h_n(y'(\theta) - P^*_{n,r}(\theta)), \theta \in]t_n, t_{n+1}[,$$

$$y'(\theta) - P^*_{n,r}(\theta) = \frac{1}{(r+2)!} y^{(r+3)}(\xi) \pi^*_{n,r}(\xi), \xi \in]t_{n,r}, t_{n+1}[,$$

où

$$\pi^*_{n,r}(t) = \prod_{-1 \leq i \leq r} (t - t_{n-i}).$$

Il résulte de (3.9) que

$$\begin{aligned} |\pi^*_{n,r}(\xi)| &= |\xi - t_{n+1}||\pi_{n,r}(\xi)| \\ &\leq (r+1) h_{max} (r+1)! h^{r+1}_{max} \\ &\leq (r+2)! h^{r+2}_{max}, \end{aligned}$$

$$|\int_{t_n}^{t_{n+1}}(y'(t)-P_{n,r}^*(t))dt| \leq |y^{r+3}(\xi)|h_n h_{max}^{r+2}.$$

Par conséquent l'erreur de consistance admet la majoration

$$|e_n| \leq Ch_n h_{max}^{r+2}(1+O(h_n)),$$

Avec $C = \max_{t\in[t_0,t_0+T]} |y^{(r+3)}|$. Donc la méthode AM_{r+1} est d'ordre $r+2$.
On initialisera les r premières valeurs $y_1, \cdots, y_r$ à l'aide d'une méthode de Runge Kutta d'ordre $r+2$ (ou à la rigueur $r+1$).

3.2.6 Stabilité de la méthode AM_{r+1}

On suppose que les rapports $\dfrac{h_n}{h_{n-1}}$ restent bornés, de sorte que les quantités

$$\lambda_r^* = \max_{1\leq n\leq N} \sum_{-1\leq i\leq r} |b_{n,i,r}^*|$$

et

$$\gamma_r^* = \max_{1\leq n\leq N} |b_{n,-1,r}^*|$$

sont controlées.

Supposons également que $f(t,y)$ soit k lipschitzienne en y. D'après le théorème du point fixe, la méthode de résolution itérative pour y_{n+1} fonctionne dès que $h_n < \dfrac{1}{|b_{n,-1,r}^*|k}$ en particulier dès que $h_{max} < \dfrac{1}{\gamma_r^* k}$, Ce qui nous supposons désormais.
Soit $\widetilde{y}_n$ une suite perturbée telle que

$$\begin{cases} \widetilde{y}_{n+1} = \widetilde{y}_n + h_n(b_{n,-1,r}^* \widetilde{f}_{n+1} + \sum\limits_{0\leq i\leq r} b_{n,i,r}^* \widetilde{f}_{n-i}) + \varepsilon_n, \\ \widetilde{f}_{n-i} = f(t_{n-i}, \widetilde{y}_{n-i}), \qquad r \leq n \leq N, \end{cases}$$

Posons $\theta_n = \max\limits_{0\leq i\leq n} |\widetilde{y}_i - y_i|$. Comme on a $\theta_{n+1} = \max(|\widetilde{y}_{n+1} - y_{n+1}|, \theta_n)$, il vient

$$\theta_{n+1} \leq \theta_n + kh_n(|b_{n,-1,r}^*|\theta_{n+1} + \sum_{0\leq i\leq r} |b_{n,i,r}^*|\theta_n) + |\varepsilon_n|,$$

$$\begin{aligned} \theta_{n+1}(1-|b_{n,-1,r}^*|kh_n) &\leq \theta_n(1+\sum_{0\leq i\leq r} |b_{n,i,r}^*|kh_n) + |\varepsilon_n| \\ &\leq (1+\sum_{0\leq i\leq r} |b_{n,i,r}^*|kh_n)(\theta_n + |\varepsilon_n|). \end{aligned}$$

Or $1 - |b^*_{n,-1,r}|kh_n \geq 1 - \gamma^*_r kh_{max} > 0$, par suite

$$\begin{aligned}
\theta_{n+1} &\leq \frac{1 + \sum\limits_{0\leq i\leq r} |b^*_{n,i,r}|kh_n}{1 - |b^*_{n,-1,r}|kh_n}(\theta_n + |\varepsilon_n|), \\
\theta_{n+1} &\leq (1 + \frac{\sum\limits_{-1\leq i\leq r} |b^*_{n,i,r}|kh_n}{1 - |b^*_{n,-1,r}|kh_n})(\theta_n + |\varepsilon_n|), \\
\theta_{n+1} &\leq (1 + \Lambda h_n)(\theta_n + |\varepsilon_n|),
\end{aligned}$$

avec $\Lambda = \dfrac{\lambda^*_r k}{1 - \gamma^*_r kh_{max}}$.
D'où d'après le lemme de Gronwall :

$$\theta_n \leq e^{\Lambda T}(\theta_r + \sum_{i=r}^{n} |\varepsilon_i|).$$

D'où la constante de stabilité $S = e^{\Lambda T}$. Lorsque h_{max} est assez petit devant $\dfrac{1}{\gamma^*_r k}$ on obtient : $S \simeq e^{\lambda^*_r kT}$.

Remarque 3.2.4 *Le tableau précédent montre que la méthode AM_{r+1} est plus stable que la méthode d'AB_{r+2}.*

3.3 Méthodes de prédiction-correction

3.3.1 Principe

Nous avons déjà évoqué les difficultés pratiques rencontrées lors de l'utilisation d'une méthode à pas multiples implicite, liées à la résolution numérique à chaque étape de l'équation (3.4) quand $\beta_r \neq 0$, généralement non linéaire, par une méthode des approximations successives. Bien que l'on puisse garantir, en prenant une grille de discrétisation suffisamment fine, que la suite définie par la relation de récurrence (3.5) sera convergente pour toute initialisation arbitraire, on ne sait cependant pas prédire combien d'itérations et donc combien d'évaluations de la fonction f seront nécessaires pour atteindre une précision voulue. Cette incertitude sur le coût de calcul a priori des méthodes implicites rend l'emploi de ces dernières délicat dans certains domaines d'applications. On peut évidemment chercher à rendre cette étape moins coûteuse en fournissant une initialisation raisonnable, mais cela ne permettra pas de contrôler le nombre d'itérations de point fixe réellement effectuées. L'idée des méthodes de prédiction

correction repose sur la prise en compte de ces deux dernières remarques, en tirant parti d'une méthode explicite (le schéma (3.4) quand $\beta_r = 0$, qualifiée de prédicteur, pour obtenir une approximation $y_{n+r}^{(0)}$ de la valeur y_{n+r}, solution de l'équation (3.4), recherchée, dont on se sert pour effectuer un nombre fixé à l'avance d'itérations de point fixe associées à une méthode implicite, alors appelée le correcteur, La méthode vise à combiner la stabilité de la méthodes implicite et la rapidité computationnelle de la méthode explicite. Dans toute la suite, nous distinguerons le correcteur du prédicteur en attachant des astérisques à tout paramètre s'y rapportant, comme ses coefficients α_i^* et β_i^*, $\forall i = 0, ..., r$. Posons p l'ordre de prédiction et p^* l'ordre de correction. L'implémentation d'une méthode de prédiction correction comporte plusieurs phases, que nous allons maintenant décrire. En supposant que les approximations $y_{n+i}, i = 0, ..., r$ ont été calculées aux étapes précédentes (ou font partie des valeurs de démarrage de la méthode) et que les quantités $f_{n+i} = f(t_{n+i}, y_{n+i}), i = 0, ..., r$ sont également connues. La prédiction consiste en l'obtention de la valeur $y_{n+r}^{(0)}$, donnée par

$$y_{n+r}^{(0)} = \sum_{i=0}^{r-1} (h\beta_i f_{n+i} - \alpha_i y_{n+i}) \tag{3.11}$$

Suit une évaluation de la fonction f utilisant cette approximation,

$$f(t_{n+r}, y_{n+r}^{(0)}) \tag{3.12}$$

qui permet alors une correction

$$y_{n+r}^{(1)} = h\beta_r^* f(t_{n+r}, y_{n+r}^{(0)}) + \sum_{i=0}^{r-1} (h\beta_i^* f_{n+i} - \alpha_i^* y_{n+i}). \tag{3.13}$$

Un moyen mnémotechnique pour décrire les diverses mises en oeuvre possibles à partir de ces trois phases est de désigner ces dernières respectivement par les lettres P, E et C, l'ordre des lettres indiquant leur enchaînement dans la méthode. Par exemple, à chaque étape de la résolution, une méthode de type PEC effectue, dans cet ordre, les calculs (3.11), (3.12) et (3.13), suivis des affectations $y_{n+r} = y_{n+r}^{(1)}$ et $f_{n+r} = f(t_{n+r}, y_{n+r}^{(0)})$. Remarquons qu'on aurait pu choisir d'utiliser la valeur $y_{n+r}^{(1)}$ pour mettre à jour f_{n+r} en effectuant une nouvelle évaluation de la fonction f,

$$f(t_{n+r}, y_{n+r}^{(1)})$$

ou même d'utiliser cette évaluation pour faire une seconde itération de correction,

$$y_{n+r}^{(2)} = h\beta_r^* f(t_{n+r}, y_{n+r}^{(1)}) + \sum_{i=0}^{r-1} (h\beta_i^* f_{n+i} - \alpha_i^* y_{n+i}).$$

et alors poser $y_{n+r} = y^{(2)}_{n+r}$, donnant ainsi respectivement lieu aux modes PECE et PECEC de la méthode de prédiction-correction. On regroupe l'ensemble des modes ainsi formés par des combinaisons de ces deux procédés sous la notation condensée $P(EC)^\mu E^{1-\tau}$, avec μ un entier naturel et $\tau = 0$ ou 1.

3.3.2 Exemples de méthodes de prédiction correction en mode PECE

Exemple 3.3.1 *On prend comme prédicteur la méthode d'Euler explicite et comme correcteur la méthode AM_1.*

$$\begin{cases} P : y^{(0)}_{n+1} = y_n + h_n f_n \\ E : f^{(0)}_{n+1} = f(t_{n+1}, y^{(0)}_{n+1}) \\ C : y^{(1)}_{n+1} = y_n + h_n(\frac{1}{2} f^{(0)}_{n+1} + \frac{1}{2} f_n) \\ E : f^{(1)}_{n+1} = f(t_{n+1}, y^{(1)}_{n+1}). \end{cases}$$

Cet algorithme coïncide avec la méthode de Heun, qui n'est autre que la méthode de Runge Kutta 2 définie par

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 0 & 1 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

Exemple 3.3.2 *Le prédicteur est AB_{r+1} (d'ordre $r+1$) et le correcteur est AM_{r+1} (d'ordre $r+2$).*

$$\begin{cases} P : y^{(0)}_{n+1} = y_n + h_n \sum\limits_{0 \le i \le r} b_{n,i,r} f_{n-i} \\ E : f^{(0)}_{n+1} = f(t_{n+1}, y^{(0)}_{n+1}) \\ C : y^{(1)}_{n+1} = y_n + h_n(b^*_{n,-1,r} f^{(0)}_{n+1} + \sum\limits_{0 \le i \le r} b^*_{n,i,r} f_{n-i}) \\ E : f^{(1)}_{n+1} = f(t_{n+1}, y^{(1)}_{n+1}). \end{cases}$$

3.3.3 Etude d'ordre de la méthode de prédiction correction

Compte tenu de leur fonctionnement, on conçoit facilement que l'erreur de consistance des schémas obtenus combine l'erreur de consistance du prédicteur avec celle du correcteur de manière plus ou moins évidente selon le mode considéré. Nous considérerons ici que le mode en question est $P(EC)^\mu$, avec $\mu \ge 1$.

Supposons que la solution y du problème de Cauchy est de classe $\mathcal{C}^{\max(p,p^*)+1}$. Pour le prédicteur nous avons d'après (3.7) et (3.8)

$$e_{n+r} = \sum_{i=0}^{r} \alpha_i y(t_{n+i}) - h \sum_{i=0}^{r-1} \beta_i f(t_{n+i}, y(t_{n+i})) = C_{p+1} h^{p+1} y^{p+1}(t_n) + o(h^{p+2}).$$

En additionnant la seconde égalité à (3.11) sous l'hypothèse

$$y_{n+i} = y(t_{n+i}),\ i = 0, ..., r-1, \tag{3.14}$$

il vient

$$y(t_{n+r}) - \overline{y}_{n+r}^{(0)} = C_{p+1} h^{p+1} y^{p+1}(t_n) + o(h^{p+2}), \tag{3.15}$$

où $\overline{y}_{n+r}$ est l'approximation fournie par la méthode sous l'hypothèse (3.14). Pour le correcteur, nous avons

$$e_{n+r}^* = \sum_{i=0}^{r} \alpha_i^* y(t_{n+i}) - h \sum_{i=0}^{r} \beta_i^* f(t_{n+i}, y(t_{n+i})) = C_{p^*+1}^* h^{p^*+1} y^{p^*+1}(t_n) + o(h^{p^*+2}),$$

et, en additionnant la seconde égalité à (3.13) pour $0 \leq k \leq \mu - 1$ sous l'hypothèse (3.14) et en utilisant le théorème des accroissements finis (voir annexe), on trouve

$$\begin{aligned} y(t_{n+r}) - \overline{y}_{n+r}^{(k+1)} &= h\beta_r^* \frac{\partial f}{\partial y}(t_{n+r}, \eta_k)(y(t_{n+r}) - \overline{y}_{n+r}^{(k)}) \\ &+ C_{p^*+1}^* h^{p^*+1} y^{p^*+1}(t_n) + o(h^{p^*+2}), \end{aligned} \tag{3.16}$$

où η_k est un point intérieur du segment joingnant $\mathrm{y}(\mathrm{t}_{n+r})$ à $\overline{y}_{n+r}^{(k)}$. Pour pouvoir poursuivre, il nous faut discuter en fonction des valeurs relatives des entiers p et p^*.

Si $p \geq p^*$, on obtient en reportant (3.15) dans (3.16) pour $k = 0$

$$y(t_{n+r}) - \overline{y}_{n+r}^{(1)} = C_{p^*+1}^* h^{p^*+1} y^{(p^*+1)}(t_n) + o(h^{p^*+2}).$$

En reportant cette nouvelle égalité dans (3.16) pour $k = 1$, il vient

$$y(t_{n+r}) - \overline{y}_{n+r}^{(2)} = C_{p^*+1}^* h^{p^*+1} y^{(p^*+1)}(t_n) + o(h^{p^*+2}).$$

En réitérant ce procédé, on trouve finalement que

$$y(t_{n+r}) - \overline{y}_{n+r}^{(\mu)} = C_{p^*+1}^* h^{p^*+1} y^{(p^*+1)}(t_n) + o(h^{p^*+2}).$$

Par conséquent, la méthode de prédiction correction a le même ordre que le correcteur pour toute valeur de l'entier μ.

Si $p = p^* - 1$, on obtient cette fois pour $k = 0$

$$y(t_{n+r}) - \overline{y}_{n+r}^{(1)} = h^{p^*+1} \Big(\beta_r^* \frac{\partial f}{\partial y}(t_{n+r}, \eta_k) C_{p^*} y^{(p^*)}(t_n) + C_{p^*+1} y^{(p^*+1)}(t_n) \Big) + o(h^{p^*+2}).$$

En effectuant des substitutions successives, on trouve que l'ordre de la méthode est bien celui du correcteur.

Si $p = p^* - 2$, on a

$$y(t_{n+r}) - \overline{y}_{n+r}^{(1)} = \beta_r^* \frac{\partial f}{\partial y}(t_{n+r}, \eta_k) C_{p^*-1} h^{p^*} y^{(p^*-1)}(t_n) + o(h^{p^*+1}),$$

et l'ordre de la méthode est inférieur à celui du correcteur si $\mu = 1$. Pour $\mu = 2$, on retrouve l'ordre du correcteur

$$\begin{aligned} y(t_{n+r}) - \overline{y}_{n+r}^{(2)} &= h^{p^*+1}\Big(\big(\beta_r^* \frac{\partial f}{\partial y}(t_{n+r}, \eta_k)\big)^2 C_{p^*-1}^* y^{(p^*-1)}(t_n) + C_{p^*+1} y^{(p^*+1)}(t_n)\Big) \\ &+ o(h^{p^*+2}), \end{aligned}$$

alors que l'ordre est le même du schéma correcteur dès que $\mu \geq 3$. La tendance est donc claire : l'ordre d'une méthode de prédiction correction dépend à la fois de l'écart entre les ordres du prédicteur et du correcteur et du nombre d'étapes de correction. On peut résumer ces constatations dans la proposition suivante :

Proposition 3.3.3 *Soit une méthode de prédiction correction en mode $P(EC)^\mu$, avec $\mu \geq 1$, basée sur une paire de méthodes à pas multiples d'ordre p pour le prédicteur et p^* pour le correcteur.*

- *Si $p \geq p^*$ (ou si $p < p^*$ et $\mu \geq p^* - p$), la méthode de prédiction correction a le même ordre que le correcteur.*
- *Si $p < p^*$ et $\mu < p^* - p$, la méthode de prédiction correction est d'ordre $p + \mu < p^*$.*

3.4 Application aux EDP

Pour résoudre numériquement une EDP il faut discrétiser toutes les variables. Supposons qu'on ait une variable temporelle t et une variable spatiale $\boldsymbol{x}$. On peut commencer la discrétisation en espace comme on peut commencer en temps. Pour la discrétisation en espace plusieurs possibilités s'offrent à nous : différences finis, éléments finis, méthodes spectrales,...
Dans ce travail on a choisi de donner comme exemple la discrétisation de l'équation de la chaleur par le schéma d'Euler implicite en temps et par la méthode spectrale en espace-la méthode spectrale est une méthode de Galerkin avec intégration numérique-[1]. Cette démarche a été faite en détail dans [9]. On considère donc le problème de Dirichlet à condition initiale de Cauchy pour l'équation de la chaleur suivant :

$$\begin{cases} \dfrac{\partial u}{\partial t} - \Delta u(t,x) = f(t,x) \ \ dans \ \ ]0,\mathrm{T}[\times\Omega, \\ u = 0 \ \ sur \ \ ]0,\mathrm{T}[\times\partial\Omega, \\ u(0,x) = u_0 \ \ dans \ \ \Omega, \end{cases} \tag{3.17}$$

tels que : Ω est un domaine borné de $\mathbb{R}^2$ assez régulier de frontière $\partial\Omega$, f est une donnée dans $L^2(0,T;L^2(\Omega))$ et u_0 dans $L^2(\Omega)$ (Voir [9] pour la définition de ces espaces fonctionnels).

On commence donc par discrétiser le problème (3.17) en temps. Cette première étape est appelée semi-discrétisation en temps : On considère un maillage de l'intervalle $[0,T]$: $0=t_0 \leq t_1 \leq ... \leq t_K = T$. On note par $h_k := t_k - t_{k-1}$, $1 \leq k \leq K$ le pas de temps.
Le problème semi-discret issu du schéma d'Euler implicite est le suivant :

$$\begin{cases} u^k = u^{k-1} + h_k(\Delta u^k + f^k) \ \ dans\ \Omega,\ 1 \leq k \leq K, \\ u^k = 0 \ \ sur\ \partial\Omega,\ 1 \leq k \leq K, \\ u^0 = u_0 \ \ dans \ \ \Omega, \end{cases} \tag{3.18}$$

où u^k est la solution approchée par le schéma d'Euler implicite de u à l'instant t_k i.e :
$u^k(x) \approx u(t_k, x)$, pour presque tout $x \in \Omega$. La donnée $f^k := f(t_k, x)$, et $u^0 = u(0, x)$. Autrement dit, nos inconnues dans (3.18) sont les k fonctions : $u^1(x), u^2(x), ..., u^K(x)$.
La semi-discrétisation aboutit donc à un système de formulations variationnelles équivalentes suivant :
Trouver $(u^k)_{0 \leq k \leq K}$ dans $L^2(\Omega) \times H_0^1(\Omega)^K$ telles que $u^0 = u_0$ dans Ω, et

$$\forall v \in H_0^1(\Omega),\ a^k(u^k, v) = L^k(v), \tag{3.19}$$

les formes bilinéaires $a^k,\ 1 \leq k \leq K$, sont définies par

$$a^k(u^k, v)(x) = \int_\Omega u^k(x)v(x)dx + h_k \int_\Omega \nabla u(x)\nabla v(x)dx,$$

et les formes linéaires L^k sont définies par

$$L^k(v)(x) = \int_\Omega u^{k-1}(x).v(x)dx + h_k \int_\Omega f^k(x)v(x)dx.$$

L'existence et l'unicité de la solution $(u^k)_{0 \leq k \leq K}$ du problème variationnelle a été établie dans [9], pour toute donnée f dans $\mathcal{C}^0(0, T; L^2(\Omega))$ et $u_0 \in L^2(\Omega)$ en utilisant le théorème de Lax-Milgram.
On définit maintenant pour $0 \leq k \leq K$ la norme

$$[[u^k]]_K = \left(\|u^K(t)\|_{0,\Omega}^2 + \sum_{k=0}^{K} h_k \|\nabla u^k\|_{0,\Omega}^2 \right)^{\frac{1}{2}}, \tag{3.20}$$

où $\|.\|_{0,\Omega}$ désigne la norme de $L^2(\Omega)$.
L'étape suivante est d'estimer l'erreur entre la solution exacte du problème variationnel (3.19) et la solution approchée donnée par le schéma d'Euler implicite, en norme (3.20) : elle est donnée par

$$[[u(t_k) - u^k]]_K \leq \frac{2}{3}(\max_{1 \leq k \leq K} h_k) \| \frac{\partial^2 u}{\partial t^2} \|_{L^2(0,T;L^2(\Omega))}^2,\ 1 \leq k \leq K$$

(Voir toujours [9] pour les détails).
On passe ensuite à la discrétisation en espace par la méthode spectrale, cette méthode consiste à approcher les $(u^k)_{1\leq k\leq K}$ dans un espace de dimension finie, ici $\mathbb{P}_N(]-1,1[^2)$ ($\mathbb{P}_N$ est l'espace vectoriel de polynômes de degré $\leq N$) est d'utiliser ensuite une intégration numérique : La formule de quadrature de Gausse Lobatto qui est exacte dans $\mathbb{P}_{2N-1}(]-1,1[)$ i.e :

$$\forall \varphi_N \in \mathbb{P}_{2N-1}(]-1,1[),\ \int_{-1}^{1} \varphi(x)dx = \sum_{j=0}^{N} \varphi(\xi_j)\rho_j.$$

tels que $\xi_0 = -1$, $\xi_N = 1$, $\xi_j,\ 1 \leq j \leq N-1$, sont les zéros du polynôme L'_N (L_N est le polynôme de Legendre de degré N), $\rho_j,\ 0 \leq j \leq N$, sont les poids associés.
Le problème discret associé à (3.19) est le suivant :
Trouver $(u_N^k)_{0\leq k\leq K}$ dans $\mathbb{P}_N(\Omega) \times (\mathbb{P}_N(\Omega))^N \cap H_0^1(\Omega)$ tels que $u_N^0 = \mathcal{I}_N u_0$, *dans* Ω, ($\mathcal{I}_N$ est le polynôme d'interpolation aux points de Gauss-Lobatto) et pour $1 \leq n \leq N$,

$$\forall v_N \in \mathbb{P}_N(\Omega) \cap H_0^1(\Omega),\ a_N^k(u_N^k, v_N) = L_N^k(v_N),$$

et les formes bilinéaires discrètes $a_N^k,\ 1 \leq k \leq K$, sont définies par :

$$a_N^k(u_N^k, v_N) = \sum_{i=0}^{N}\sum_{j=0}^{N} u_N^k(\xi_i)v_N(\xi_j)\rho_i\rho_j + h_k \sum_{i=0}^{N}\sum_{j=0}^{N} \nabla u_N(\xi_i)\nabla v_N(\xi_j)\rho_i\rho_j,$$

et la forme linéaire L_N^k est définie par

$$L_N^k(v_N) = \sum_{i=0}^{N}\sum_{j=0}^{N} u_N^{k-1}(\xi_i)v_N(\xi_j)\rho_i\rho_j + h_k \sum_{i=0}^{N}\sum_{j=0}^{N} f^k(\xi_i)v_N(\xi_j)\rho_i\rho_j.$$

Une fois l'existence et l'unicité est établie, l'étape finale consiste à estimer l'erreur spatiale entre $(u^k)_{0\leq k\leq K}$ et $(u_N^k)_{0\leq k\leq K}$ i.e : $[[u^k - u_N^k]]_K$ (voir [9]).
L'erreur globale $[[u(t_k) - u_N^k]]_K$ est donc déduite des deux discrétisation temporelle et spatiale i.e :

$$[[u(t_k) - u_N^k]]_K \leq [[u(t_k) - u^k]]_K + [[u^k - u_N^k]]_K.$$

Chapitre 4

Mise en oeuvre

Le but de ce chapitre est de montrer comment mettre en oeuvre les schémas classiques de résolution numérique des équations différentielles vues précédemment à l'aide de matlab. On estime aussi leurs ordres de convergence respectifs.
Commençons par considérer le problème suivant

$$\begin{cases} y' = t - ty \\ y(0) = 2 \end{cases} \tag{4.1}$$

On utilise cet exemple comme référence pour analyser la précision des différentes solutions numériques.
La fonction matlab f.m correspondante est la suivante

```
function yp=f(t,y)
yp=t-t*y;
```

Et la fonction matlab fex.m correspondante à la solution exacte est :

```
function y=fex(t)
y=1+exp(-t.^2/2);
```

4.1 Mise en oeuvre de quelques méthodes de résolution numérique d'une EDO

4.1.1 La méthode d'Euler explicite

Soit la fonction matlab eulerexplicite.m ; elle renvoie les temps successifs où sont effectuées les solutions approchées.

```
function [liste_t,liste_y]=eulerexplicite(y0,N,T)
y=y0;liste_y=[y0];
t=0;liste_t=[0];
h=T/N;
for i=1:N
    y=y+h*f(t,y);
    t=t+h;
    liste_y=[liste_y,y];
    liste_t=[liste_t,t];
end
```

Si l'on souhaite avoir accès aux deux arguments de sortie c.à.d. les temps et les valeurs des approximations, il faut effectuer l'appel comme suit :

```
>> [tps,sol]=eulerexplicite(2,5,2)
tps =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
sol =
    2.0000    2.0000    1.8400    1.5712    1.2970    1.1069
```

4.1.2 La méthode d'Euler implicite

La fonction correspondante est la suivante

```
function[t,y]=eulerimplicite(f,T,y0,h)
N=T/h; t(1)=0; y(1)=y0;
fori=1:N
    t(i+1)=t(i)+h;
    ynew=y(i)+h*feval(f,t(i),y(i));
    y(i+1)=y(i)+h*feval(f,t(i+1),ynew);
end
```

Le programme pour calculer les solutions approchées du problème (4.1) à partir de la méthode d'Euler explicite et d'Euler implicite est le suivant :

```
y0=2; N=5; T=2;
[t,yexpl]=eulerexplicite(2,5,2)
[t,yimpl]=eulerimplicite('f',2,2,0.4)
```

```
t=[0:0.4:2]
yexact=[fex(t)]
%graphique de comparaison entre la solution exacte et la
%la solution approchée par les méthodes
%d'Euler explicite et d'Euler implicite
plot(t,yexpl,'r:',t,yimpl,'b',t,fex(t),'b--','Linewidth',2)
xlabel('temps t','fontsize',14) ylabel('y(t)','fontsize',14)
legend('Euler explicite','Euler implicite','solution exacte')
```

L'exécution de ce programme donne le résultat suivant :

```
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yexpl =
    2.0000    2.0000    1.8400    1.5712    1.2970    1.1069
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yimpl =
    2.0000    1.8400    1.6142    1.4137    1.2760    1.1965
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yexact =
    2.0000    1.9231    1.7261    1.4868    1.2780    1.1353
```

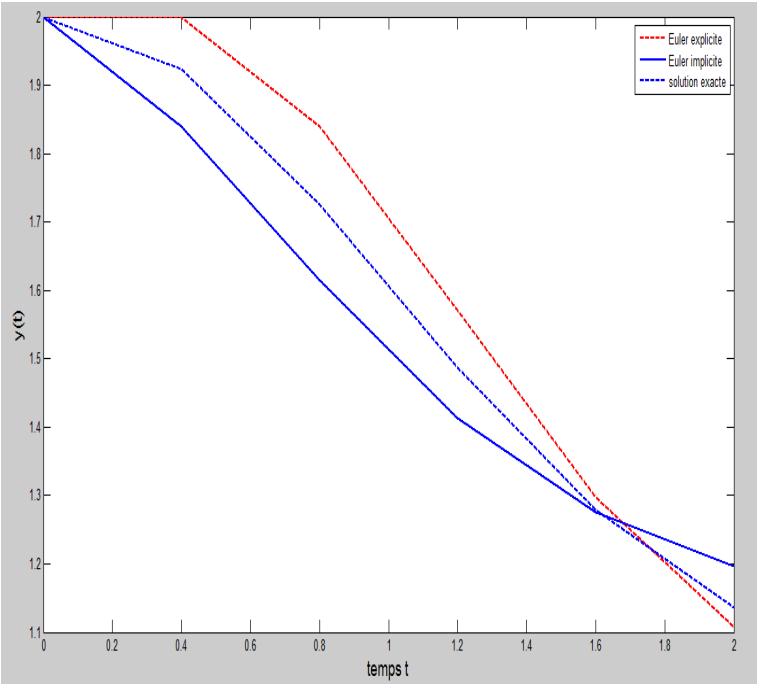

FIG. 4.1 – Graphique de comparaison entre la solution exacte et la la solution approchée par les méthodes d'Euler explicite et d'Euler implicite

4.1.3 La méthode de Runge Kutta 4

```
function[t,y]=rk4(f,T,y0,h)
N=T/h;t(1)=0; y(1)=y0;
for i=1:N
    f1=h*feval(f,t(i),y(i));
    f2=h*feval(f,t(i)+h/2,y(i)+f1/2);
    f3=h*feval(f,t(i)+h/2,y(i)+f2/2);
    f4=h*feval(f,t(i)+h,y(i)+f3);
    y(i+1)=y(i)+(f1+2*f2+2*f3+f4)/6;
    t(i+1)=t(i)+h;
end
```

Et le programme principal pour calculer les solutions approchées du problème (4.1) à partir de la méthode d'Euler explicite et de Runge Kutta d'ordre 4 est le suivant :

```
y0=2;
N=5;
```

```
 T=2;
[t,yexpl]=eulerexplicite(2,5,2)
[t,yrk4]=rk4('f',2,2,0.4)
t=[0:0.4:2]
yexact=[fex(t)]
%graphique de comparaison entre la solution exacte la solution
%approchée par les méthodes
d'Euler explicite et du Runge Kutta 4
plot(t,yexpl,'g',t,yrk4,'b',t,fex(t),'r--','Linewidth',2)
xlabel('temps t','fontsize',14) ylabel('y(t)','fontsize',14)
legend('Euler explicite','Runge Kutta 4','solution exacte')
```

L'exécution de ce programme donne le résultat suivant :

```
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yexpl =
    2.0000    2.0000    1.8400    1.5712    1.2970    1.1069
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yrk4 =
    2.0000    1.9231    1.7261    1.4868    1.2782    1.1358
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yexact =
    2.0000    1.9231    1.7261    1.4868    1.2780    1.1353
```

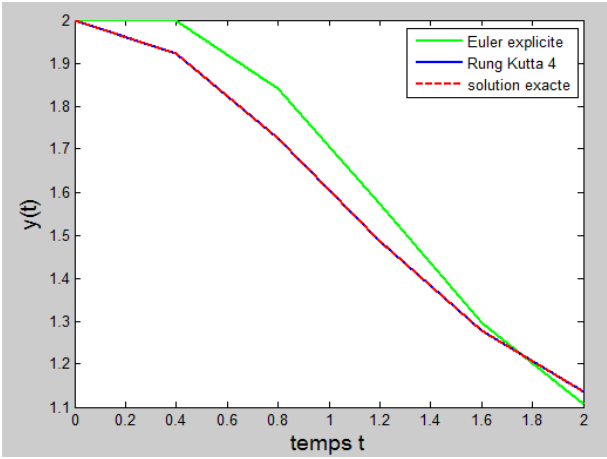

FIG. 4.2 – Graphique de comparaison entre la solution exacte et la solution approchée par les méthodes d'Euler explicite et de Runge Kutta 4

4.1.4 La méthode d'Adams Bashforth d'ordre 3

Le programme de cette méthode est le suivant :

```
t(1)=0;
y(1)=2;
h=0.4;
for i=1:6
    f(i)=(t(i)-t(i)*y(i));
    if(i<3)
        y(i+1)=y(i)+h*(t(i)-t(i)*y(i));
    else
        y(i+1)=y(i)+h/12*(23*f(i)-16*f(i-1)+5*f(i-2));
    end
    t(i+1)=t(i)+h;
end
t=[t(1),t(2),t(3),t(4),t(5),t(6)]
y=[y(1),y(2),y(3),y(4),y(5),y(6)]
t=[0:0.4:2]
yexact=[fex(t)]
%graphique de comparaison entre la solution exacte
%et la solution approchée par la méthode d''AB3
```

```
hold on plot(t,y,'r',t,fex(t),'b','Linewidth',2)
legend('AB3','solution exacte')
xlabel('temps t','fontsize',14)
ylabel('y(t)','fontsize',14)
```

Ce programme donne le résultat suivant :

```
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
y =
    2.0000    2.0000    1.8400    1.5381    1.3348    1.1565
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yexact =
    2.0000    1.9231    1.7261    1.4868    1.2780    1.1353
```

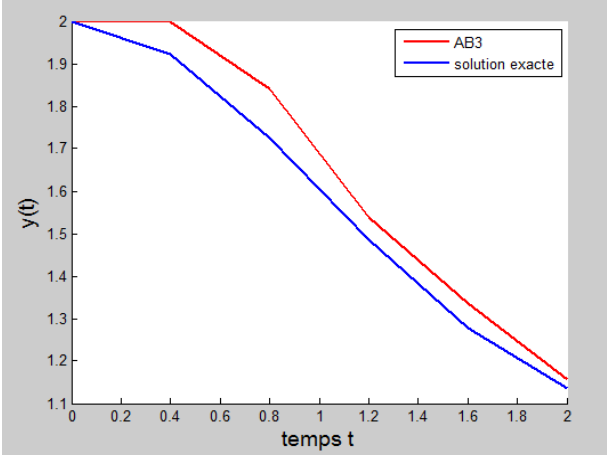

FIG. 4.3 – Graphique de comparaison entre la solution exacte et la solution approchée par la méthode d'AB3

4.1.5 La méthode de prédiction correction

On prend le prédicteur Euler explicite et le correcteur AM_1(d'ordre 2), le programme de cette méthode est le suivant

```
function[t,y]=pc(f,T,y0,h) N=T/h; y(1)=y0; t(1)=0;
for i=1:N
```

```
    k1=feval(f,t(i),y(i));
    t(i+1)=t(i)+h;
    k2=feval(f,t(i+1),y(i)+(h*k1));
    y(i+1)=y(i)+h*(k1+k2)/2;
end
```

Le programme principal est donné par :

```
y0=2;
N=5;
T=2;
[t,ypc]=pc('f',2,2,0.4)
t=[0:0.4:2]
yexact=[fex(t)]
%graphique de comparaison entre la solution exacte
%et la solution approchée par la méthodes de
%prédiction correction
plot(t,ypc,'r--',t,fex(t),'b','Linewidth',2)
xlabel('t','fontsize',14 )
ylabel('y(t)','fontsize',14)
legend('prédiction correction','solution exacte')
```

L'exécution de ce programme donne le résultat suivant :

```
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
ypc =
    2.0000    1.9200    1.7228    1.4892    1.2904    1.1556
t =
         0    0.4000    0.8000    1.2000    1.6000    2.0000
yexact =
    2.0000    1.9231    1.7261    1.4868    1.2780    1.1353
```

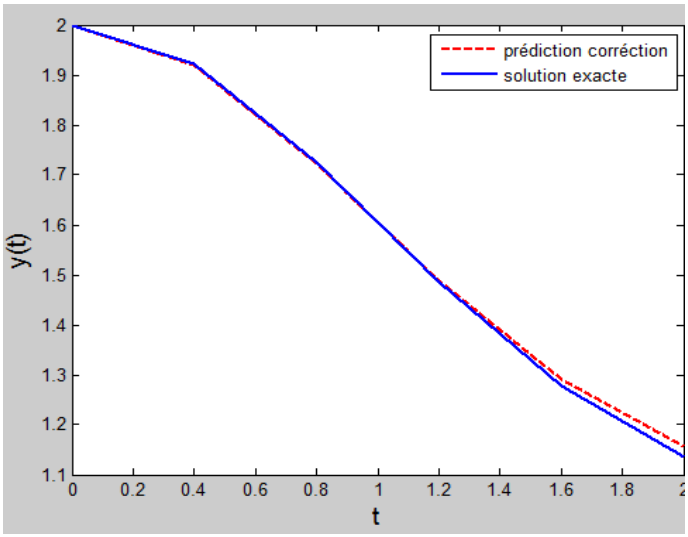

FIG. 4.4 – Graphique de comparaison entre la solution exacte et la solution approchée par la méthodes de prédiction correction

4.1.6 Estimation de l'ordre de convergence d'une méthode numérique

On veut estimer l'ordre de convergence de différentes méthodes en appliquant la formule (2.7). Considérons le problème de Cauchy suivant :

$$\begin{cases} y'(t) = \cos(2 * y(t)), \quad t \in]0, 1], \\ y(0) = 0, \end{cases}$$

Dont la solution exacte est

$$y(t) = \frac{1}{2} \arcsin((e^{4t} - 1)/(e^{4t} + 1)).$$

On le résout avec deux méthodes : Runge kutta 2 (i.e. d'ordre 2) et Runge Kutta 3 (i.e. d'ordre 3), ces méthodes sont programmées par des fonctions matlab rk2.m et rk3.m suivantes :

```
function [tt,u]=rk2(odefun,tspan,y0,N )
tt=linspace(tspan(1),tspan(2),N+1);
h=(tspan(2)-tspan(1))/N;
hh=h*0.5; u=y0;
for t=tt(1:end -1)
y = u(end ,:);
k1=feval(odefun,t,y);
t1 = t + h; y = y + h*k1;
```

```
k2=feval(odefun,t1,y);
u = [u; u(end ,:) + hh*(k1+k2)];
end

function[tt,u]=rk3(odefun,tspan,y0,N);
tt=linspace(tspan(1),tspan(2),N+1);
h=(tspan(2)-tspan(1))/N;
hh=h*0.5; h2=2*h; u=y0; h6=h/6;
for t=tt(1:end -1)
y = u(end ,:);
k1=feval(odefun,t,y);
t1 = t + hh;
y1 = y + hh* k1;
k2=feval(odefun,t1,y1);
t1 = t + h;
y1 = y+h*(2*k2-k1);
k3=feval(odefun,t1,y1);
u=[u;u(end,:)+h6*(k1+4*k2+k3)];
end
```

On considère dans le programme qui suit différentes valeures de h $(1/2, 1/4, 1/8, ..., 1/512)$.

```
t0=0; y0=0;
f=inline('cos(2*y)','t','y');
y=inline('0.5*asin((exp(4*t)-1)./(exp(4*t)+1))','t'); T=1; N=2;
fork=1:10;
 [tt,u]=rk2(f,[t0 ,T],y0,N);
 [t,r]=rk3(f,[t0 ,T],y0,N);
e(k)=abs(u(end)-feval(y,tt(end)));
e1(k)=abs(r(end)-feval(y,t(end)));
N=2*N;
end
prk2=log(abs(e(1:end-1)./e(2:end)))/log(2); prk2(1:2:end)
prk3=log(abs(e1(1:end-1)./e1(2:end)))/log(2); prk3(1:2:end)
```

L'exécution de ce programme donne le résultat suivant :

```
ans =
    2.4733    2.1223    2.0298    2.0074    2.0018
ans =
    3.1184    3.0510    3.0128    3.0032    3.0008
```

4.2 La résolution numérique d'un système d'EDO

4.2.1 Modèle proie/prédateur

Présentation du modèle : Le modèle se comporte de la façon suivante :
Lorsque la population des proies augmente, celle des prédateurs peut augmenter car elle a plus de proies pour se nourir ; inversement lorsque la population des prédateurs devient trop importante, elle se nourrit plus vite que ne peut renouveler la population de proies, conduisant au décroissement des deux populations.
Les populations de deux espèces sont notées x et y, les paramètres de leur lois d'évolution naturelle pour les proies et les prédateurs sont respectivement les coefficients μ_1 et μ_3, et les paramètres μ_2 et μ_4, sont les coefficients définissant les influences inter-populations. Le système qui régit ce comportement est le suivant :

$$\begin{cases} x'(t) = x(t)(\mu_1 - \mu_2 y(t)) \\ y'(t) = y(t)(-\mu_3 + \mu_4 x(t)) \end{cases} \tag{4.2}$$

La résolution numérique : Appliquons la méthode d'Euler explicite déja programmée avec $\mu_1 = 1.5$, $\mu_2 = 1$, $\mu_3 = 3$, $\mu_4 = 1$, et des populations initiales de 8 et 4. La méthode d'Euler est indépendante de la dimension. Pour résoudre ce système il suffit donc de redéfinir la fonction f.m comme suit :

```
function yp=f(t,y)
yp=[y(1)*(3-y(2));y(2)*(-2+2*y(1))];
end
```

Puis on appel la fonction eulerexplicite déja définie dans la section 4.1 comme suit :

```
T=10;
 N=5000;
y0=[8;4]
[tps,sol]=eulerexplicite(y0,N,T);
```

```
plot(tps,sol(1,:),tps,sol(2,:))
title('Evolution des populations de
proie prédateur pour le problème Lotka Volterra avec les paramètres
mu=(1.5,1,3,1))
legend('proie','prédateur')
```

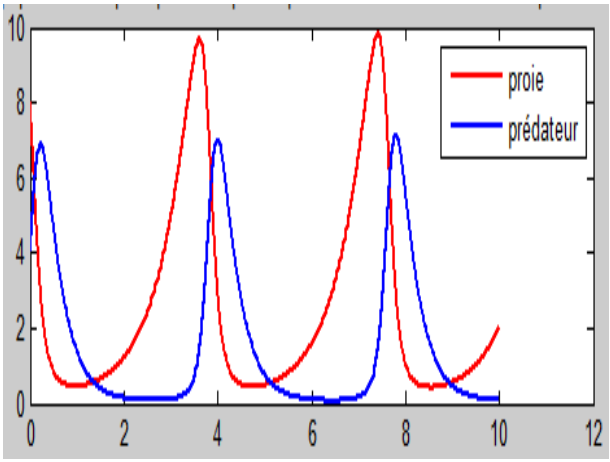

FIG. 4.5 – Evolution des populations de proie prédateur pour le problème Lotka Volterra avec les paramètres $\mu = (1.5, 1, 3, 1)$

4.2.2 Le pendule sphérique

Le mouvement d'un point $x(t) = (x_1(t), x_2(t), x_3(t))^T$ de masse m soumis à la gravité $F = (0, 0, -gm)^T$ (avec $g = 9.8m/s^2$) et contraint de se déplacer sur la surface sphérique d'équation $\Phi(x) = x_1^2 + x_2^2 + x_3^2 - 1 = 0$ est décrit par le système d'équations différentielles ordinaires suivant

$$\ddot{x} = \frac{1}{m}(F - \frac{m\dot{x}^T H\dot{x} + \nabla\Phi^T F}{|\nabla\Phi|^2}\nabla\Phi) \ \textit{pour} \ t > 0. \tag{4.3}$$

On note $\dot{x}$ la dérivée première et $\ddot{x}$ la dérivée seconde par rapport à t, $\nabla\Phi$ le gradient spatial de Φ, égal à $2x$, H la matrice hessienne de Φ dont les composantes sont $H_{ij} = \dfrac{\partial^2\Phi}{\partial x_i \partial x_j}$ pour $i, j = 1, 2, 3$. Dans notre cas, H est une matrice diagonale dont les coefficients valent 2. On complète le système (4.3) avec les conditions initiales $x(0) = x_0$ et $\dot{x}(0) = v_0$.

Pour résoudre numériquement le système (4.3), transformons-le en un système d'équations différentielles du premier ordre en la nouvelle variable y, qui est un vecteur à 6 composantes.

En posant $y_i = x_i$, $y_{i+3} = \dot{x}_i$ avec $i = 1, 2, 3$, et

$$\lambda = \frac{m(y_4, y_5, y_6)^T H(y_4, y_5, y_6) + \nabla\Phi^T F}{|\nabla\Phi|^2}, \tag{4.4}$$

on obtient, pour $i = 1, 2, 3$,

$$\begin{cases} \dot{y}_i = y_{i+3}, \\ \dot{y}_{i+3} = \dfrac{1}{m}(F_i - \lambda \dfrac{\partial\Phi}{\partial y_i}). \end{cases} \tag{4.5}$$

La résolution numérique : Pour résoudre ce système on utilise la méthode d'Euler explicite et la méthode de Runge Kutta 2.

On commence par définir une fonction matlab force.m qui fournit l'expression du second membre de (4.5) comme suit :

```
function [f]=force(t,y)
[n,m]=size(y);
f=zeros(n,m);
phix='2*y(1)';
phiy='2*y(2)';
phiz='2*y(3)';
H=2*eye(3);
mass=1; % Masse
F1='0*y(1)';
F2='0*y(2)';
F3='-mass*9.8'; % Gravité
xdot=zeros(3 ,1);
xdot (1:3)=y(4:6);
F=[eval(F1);eval(F2);eval(F3)];
G=[eval(phix);eval(phiy);eval(phiz)];
lambda=(mass*xdot'*H*xdot+F'*G)/(G'*G);
f(1:3)=y(4:6);
for k=1:3
f(k+3)=(F(k)-lambda*G(k))/mass;
end
return
```

On suppose que les conditions initiales sont données dans le vecteur $y0 = [0, 1, 0, .8, 0, 1.2]$; et que l'intervalle d'intégration

est $tspan = [0, 30]$.
Appliquons la méthode d'Euler explicite à l'aide de la fonction matlab feuler.m de la manière suivante :

```
[t,y]=feuler(@force,tspan,y0,N);
```

telle que la fonction feuler.m est la suivante

```
function [t,u]=feuler(odefun,tspan,y0,N)
h=(tspan(2)-tspan(1))/N;
y=y0;
w=y;
u=y;
tt=linspace (tspan(1),tspan(2),N+1);
fort=tt(1:end-1)
w=w+h*feval(odefun,t,w);
u = [u; w];
end
t=tt;
return
```

Appliquons la méthode de Runge Kutta 2 à l'aide de la fonction rk2.m déja définie précédemment comme suit :

```
[t,y]=rk2(@force,tspan,y0,N);
```

où N est le nombre d'intervalles (de longueur constante) utilisés pour discrétiser l'intervalle $[tspan(1), tspan(2)]$.
On prend $N = 10000$ puis $N = 100000$ noeuds de discrétisation.
Bien qu'on ne connaisse pas la solution exacte du problème, on peut avoir une idée de la précision en remarquant que la solution exacte vérifie

$$\Phi(x) = |x_1^2 + x_2^2 + x_3^2 - 1| = 0.$$

y_N étant l'approximation de la solution exacte construite au temps t_N.
En utilisant 10000 noeuds de discrétisation, on trouve $\Phi(y_N) = 1.4713$, tandis qu'avec 100000 noeuds on a $\Phi(y_N) = 0.1595$, la solution ne semble raisonnablement précise que dans le second cas, tandis que la méthode de Runge Kutta (d'ordre 2) donne avec 10000 noeuds $\Phi(y_N) = 0.1385^2 + 0.9903^2 + 0.0113^2 - 1 = 0.0000403$.
On voit donc sur cet exemple que rk2 est beaucoup mieux précise que celle d'Euler explicite.

Conclusion générale

Les méthodes numériques de résolution des équations différentielles sont nombreuses, dans ce mémoire on a présenté quelques méthodes qui sont les préférées mais elles ne sont pas forcément les meilleures, pour chaque problème, on peut toujours trouver une méthode optimale de résolution.

Enfin, nous tenons à reconnaître que notre modeste travail n'est q'une approche d'un vaste domaine que l'étude minutieuse aurait pris beaucoup plus d'années et d'effort pour être étudié en totalité.

Annexe

Théorème 4.2.1 théorème du point fixe

Soit (E,d) un espace métrique complet non vide, et $f : E \to E$ une contraction stricte, i.e. f lipschitzienne de rapport $L < 1$. Alors f admet un unique point fixe $x \in E$. De plus, la suite $(x^n)_{n\in\mathbb{N}}$ définie par $x^{n+1} := f(x^n)$ converge lorsque n tend vers l'infini pour toute initialisation x^0 vers ce point fixe.

Définition 4.2.2 *Une méthode d'intégration numérique est dite d'ordre N si la formule approchée est exacte pour tout polynôme de dégré $\leq N$ et inexacte pour au moin un polynôme de degré $\leq N-1$.*

Théorème 4.2.3 Formule d'erreur d'interpolation

On suppose que f est $n+1$ fois dérivable sur $[a,b]$. Alors pour tout $x \in [a,b]$, il existe un point $\xi_x \in]\min(x,x_i), \max(x,x_i)[$ tel que

$$f(x) - p_n(x) = \frac{1}{(n+1)!}\pi_{n+1}(x) f^{n+1}(\xi_x),$$

où $x_i \in [a,b]$ sont les points d'interpolation et $\pi_{n+1} = \prod_{i=0}^{n}(x - x_i)$.

Théorème 4.2.4 théorème de la moyenne

Si f est integrable sur $[a,b]$ et si l'on pose $m = \inf_{x\in[a,b]} f(x)$ et $M = \sup_{x\in[a,b]} f(x)$ alors

$$m(b-a) \leq \int_a^b f(t)dt \leq M(b-a).$$

Si f est continue sur $[a,b]$, alors il existe $c \in]a,b[$ tel que

$$\frac{1}{b-a}\int_a^b f(t)dt = f(c).$$

Définition 4.2.5 symbole de Landau

-On écrit $f(x) = O(g(x))$ s'il existe une constante $C \geq 0$ tel que $|f(x)| \leq C|g(x)|$, $\forall x$ dans un voisinage de 0.

-On écrit $f(x) = o(g(x))$ si $f(x) = g(x)\varepsilon(x)$ tel que $\lim_{x\to 0}\varepsilon(x) = 0$.

Théorème 4.2.6 théorème des accroissements finis

Soit $[a, b]$ *un intervalle non vide de* $\mathbb{R}$ *et* f *une application de* $[a, b]$ *dans* $\mathbb{R}$. *Si* f *est continue sur* $[a, b]$ *et dérivable sur* $]a, b[$, *alors il existe* $c \in]a, b[$ *tel que*

$$f'(c) = \frac{f(b) - f(a)}{b - a}.$$

Bibliographie

[1] C. Bernardi, Y. Maday, *Spectral Methods*, Handbook of Numerical Analysis, **V**, P. G. Ciarlet and J.-L. Lions. Eds., North-Holland, 1997.

[2] J.P. Demailly, *Analyse numérique et équations différentielles*, Grenoble sciences, EDP sciences, 2006.

[3] F. Jedrzejewski, *Introduction aux méthodes numériques*, Springer-Verlag France, Paris 2005.

[4] G. Legendre, *Introduction à l'analyse numérique et au calcul scientifique*, (version provisoire du 4 janvier 2017).

[5] J-L. Merriem, *Analyse numérique avec matlab*, Dunod, Paris, 2007.

[6] A. Quarteroni, P.Gervasio, F.Saleri, *Calcul scientifique*, Springer, 2008.

[7] M. Sibony, J.C. Mardon, *Analyse numérique* 1, *Systèmes linéaires et non linéaires*, Hermann, éditeurs des sciences et des arts, 1405, 1982.

[8] M. Sibony, J.C. Mardon, *Analyse numérique* 2, *Systèmes linéaires et non linéaires*, Hermann, éditeurs des sciences et des arts, 1406, 1982.

[9] Y. Daikh, W. Chicouche, *Spectral element discretization of the heat equation with variable diffusion coefficient*, Comment.Math.Univ.Carolin. $57, 2, 185 - 200$, 2016.

-Sites internet

[10] Jacques.lefrere@upmc.fr, *Résolution numérique des équations différentielle ordinaires (EDO).*

[11] L. Walter, *Résolution numérique d'équations différentielles chaotique par un algorithme adaptatif.*

[12] G. Bontenpi, C.Olsen, S. Bentaieb, *Calcul Formel et Numérique, Info-F*-205, Département d'informatique, Boulvard de Triophe.