Chapitre 2

Approximation des Équations Différentielles Ordinaires

2.1 Problème de Cauchy

Soit U un ouvert de $\mathbb{R}^2$, $f : U \to \mathbb{R}$ une fonction donnée et $(t_0, y_0) \in U$. On appelle problème de Cauchy le problème suivant

$$\begin{cases} \text{Trouver } y : I \longrightarrow \mathbb{R} \text{ telle que} \\ y'(t) = f(t, y(t)),\ t \in I \\ y(t_0) = y_0. \end{cases} \tag{2.1}$$

Le problème (2.1) consiste à trouver une fonction dérivable $y : I \longrightarrow \mathbb{R}$ qui vérifie (2.1). y_0 est dite valeur initiale de la solution y.

Définition 2.1 On dit qu'une fonction $f : U \subset \mathbb{R}^2 \to \mathbb{R}$ de deux variables $(t, x) \in U$ est localement lipschitzienne en x si pour tout $x_0 \in \mathbb{R}$ il existe $\eta > 0$ et $K > 0$ telle que

$$\forall (t, x_1), (t, x_2) \in U : |x_1 - x_0| < \eta \text{ et } |x_2 - x_0| < \eta \text{ alors } |f(t, x_1) - f(t, x_2)| \leq K |x_1 - x_2| .$$

Remarque On peut montrer que si f est de classe C^1 par rapport à x alors f est localement lipschitzienne en x.

Le théorème suivant appelé Théorème de Cauchy-Lipschitz assure l'existence et l'unicité de la solution du problème de Cauchy (2.1) (pour la preuve voir [3]).

Théorème 2.1 (Cauchy-Lipschitz)
Soit $f : U \to \mathbb{R}$ une fonction telle que :
(i) f est continue sur U
(ii) f est localement lipschitzienne par rapport à sa seconde variable.
Alors pour tout $(t_0, y_0) \in U$, il existe un intervalle maximal $I \subset \mathbb{R}$ contenant t_0 et une fonction $y : I \to \mathbb{R}$ telle que y soit une solution du problème de Cauchy (2.1).

Exemple Soit le problème de Cauchy suivant

$$\begin{cases} y'(t) = y^2(t) - t, & t \in \mathbb{R} \\ y(0) = 1 \end{cases} \tag{2.2}$$

La fonction $f(t, y) = y^2 - t$ est continue en $(t, y) \in \mathbb{R}^2$, de plus il est clair que f est continument dérivable en y. Donc d'après le Théorème 2.1 le problème (2.2) admet une solution unique y définie sur un certain intervalle $I \subset \mathbb{R}$.

2.2 Méthodes d'Euler

2.2.1 Les schémas

On cherche à approcher la solution $y(t)$ du problème de Cauchy (2.1) sur un intervalle borné $[t_0, T]$, où $T > t_0$ est un temps final donné. Considérons pour cela une discrétisation uniforme ou grille $t_0 < \cdots < t_n = T$ de pas $h = t_{i+1} - t_i$ de l'intervalle $[t_0, T]$. les points $\{t_i : i = 0, \cdots, n\}$ sont appelés noeuds de la grille. Écrivons que

$$y'(t_i) = f(t_i, y(t_i)), \quad i = 1, \cdots, n.$$

L'idée est d'approcher la dérivée $y'(t_i)$ en utilisant une formule de dérivation approchée. Par exemple la formule (A.11) (voir Annexe A) de dérivation approchée à droite de t_i donne

$$y'(t_i) \simeq \frac{y(t_{i+1}) - y(t_i)}{h}.$$

Posons $y_i = y(t_i)$ et $f_i = f(t_i, y_i)$, d'où

$$\frac{y_{i+1} - y_i}{h} \simeq f(t_i, y_i). \tag{2.3}$$

d'où le schéma suivant

$$\begin{cases} u_{i+1} = hf(t_i, u_i) + u_i, & i = 0, \cdots, n-1 \\ u_0 = y_0 \end{cases} \tag{2.4}$$

Remarque u_i est ici une valeur approchée de $y_i = y(t_i)$. Le rapport à gauche de la relation (2.3) faisant intervenir y_i est une approximation de f_i alors que les u_i réalisent l'égalité exacte.

Le schéma (2.4) est appelé schéma d'Euler explicite car il permet de calculer directement u_{i+1} à partir de u_i.

De la même manière si on approche la dérivée $y'(t_i)$ en utilisant la formule (A.12) de dérivation à gauche de t_i

$$y'(t_i) \simeq \frac{y(t_i) - y(t_{i-1})}{h}$$

on obtient le schéma suivant

$$\begin{cases} u_i = hf(t_i, u_i) + u_{i-1}, & i = 1, \cdots, n \\ u_0 = y_0 \end{cases} \tag{2.5}$$

appelé schéma d'Euler implicite car pour calculer u_i à partir de u_{i-1} on résout une équation dont l'inconnue est u_i. Dans le schéma explicite (2.4), u_{i+1} dépend uniquement de u_i alors que dans la schéma (2.5) u_i dépend de lui même à travers $f_i = f(t_i, u_i)$.
Le résultat suivant montre que si la grille des noeuds est suffisamment fine, le schéma d'Euler implicite (2.5) admet toujours une solution unique.

Proposition 2.1 Supposons que f est localement lipschitzienne par rapport à sa seconde variable. Alors si h suffisamment petit le schéma d'Euler implicite (2.5) admet à chaque itération i une solution unique u_i.

Preuve On montre le résultat par récurrence sur i. Supposons qu'a l'itération $i-1$ le schéma (2.5) admet une solution unique u_{i-1} et considérons l'application

$$\varphi(y) = u_{i-1} + hf(t_i, y).$$

Comme f est localement lipschitzienne par rapport à sa seconde variable, alors il existe un $\eta > 0$ et $K > 0$ tel que pour tout y_1 et y_2 vérifiant $|y_1 - u_{i-1}| \leq \eta$ et $|y_2 - u_{i-1}| \leq \eta$ on ait

$$|\varphi(y_1) - \varphi(y_2)| = h|f(t_i, y_1) - f(t_i, y_2)| \leq hK|y_1 - y_2|.$$

Si on choisit donc h de sorte que $hK < 1$ l'application φ devient contractante. D'après le Théorème de l'application contractante, φ admet un unique point fixe qui n'est autre que la solution du schéma (2.5) à l'itération i. □

Remarque Les méthodes d'Euler explicite et implicite sont dites méthodes à un pas car pour calculer la solution approchée u_i (ou u_{i+1}) on a seulement besoin des informations disponibles au nœud précédent t_{i-1}.

Exemple Calculons une solution approchée du problème (2.2) sur l'intervalle $[0, 1]$ en prenant $n = 4$.
Les nœuds sont donnés par

t_0	t_1	t_2	t_3	t_4
0	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	1

d'où le schéma explicite suivant

$$\begin{cases} u_{i+1} = u_i + \frac{1}{4}\left(u_i^2 - t_i\right), & i = 0, 1, 2, 3, \\ u_0 = 1. \end{cases}$$

A partir de ce schéma on obtient les valeurs suivantes

u_0	u_1	u_2	u_3	u_4
1	1.25	1.57	2.55	5.42

On en déduit les approximations suivantes $y\left(\frac{1}{4}\right) \simeq 1.25$, $y\left(\frac{1}{2}\right) \simeq 1.57$, $y\left(\frac{3}{4}\right) \simeq 2.55$, $y\left(1\right) \simeq 5.42$.

2.2.2 Convergence

Définition 2.2 Notons par y_i la valeur exacte de la solution du problème (2.1) en t_i et par u_i la valeur approchée associée. On dit qu'une méthode numérique converge à l'ordre $p > 0$ s'il existe une constante $C > 0$ indépendante de h telle que

$$|y_i - u_i| \leq Ch^p, \quad \forall i = 0, \cdots, n.$$

Étudions la convergence de la méthode d'Euler explicite. Pour cela on considère l'erreur d'approximation

$$e_i = y_i - u_i \tag{2.6}$$

qui représente la différence entre la valeur exacte y_i en t_i et la valeur approchée u_i. Posons

$$u_i^* = y_{i-1} + hf(t_{i-1}, y_{i-1}),$$

d'où

$$e_i = (y_i - u_i^*) + (u_i^* - u_i). \tag{2.7}$$

La méthode est donc convergente si les deux termes $(y_i - u_i^*)$ et $(u_i^* - u_i)$ tendent vers 0 lorsque $h \longrightarrow 0$. u_i^* est la valeur approchée de la solution qu'on obtiendrai en partant de la solution exacte y_{i-1} à l'instant t_{i-1}. Le terme $y_i - u_i^*$ représente donc l'erreur engendrée à l'itération i en remplaçant y_i (qui n'est pas connue exactement) par sa valeur approchée u_i dans le schéma (2.4).
Supposons que la solution y vérifie $y \in C^2([t_0, T])$, alors

$$y_i = y(t_i) = y(t_{i-1}) + hy'(t_{i-1}) + \tfrac{h^2}{2}y''(\xi_i), \quad \xi_i \in (t_{i-1}, t_i)$$

d'où puisque $u_i^* = y_{i-1} + hf(t_{i-1}, y_{i-1})$

$$y_i - u_i^* = \frac{h^2}{2}y''(\xi_i),$$

la quantité

$$\tau_i(h) = \frac{y_i - u_i^*}{h}, \quad i = 1, \cdots, n \tag{2.8}$$

est appelée erreur de consistance de la méthode d'Euler explicite. D'une manière générale on définit l'erreur de consistance de la manière suivante.

Définition 2.3 On appelle erreur de consistance d'une méthode numérique l'erreur qu'on obtient en insérant la solution exacte dans le schéma numérique de la dite méthode.

D'après la relation (2.8) l'erreur de consistance s'écrit

$$\tau_i(h) = \frac{h}{2}y''(\xi_i). \tag{2.9}$$

Si on pose $M = \sup_{t\in[t_0,T]} |y''(t)|$, on peut alors déduire de (2.9)

$$\tau(h) \leq \frac{M}{2}h \tag{2.10}$$

où $\tau(h) = \max_{0\leq i\leq n} |\tau_i(h)|$ est appelée l'erreur de consistance globale. D'après (2.10) $\tau(h) = o(h)$ la méthode d'Euler explicite est donc consistante d'ordre 1. On peut généraliser en énonçant la définition suivante.

Définition 2.4 On dit qu'une méthode numérique est consistante d'ordre $p > 0$ si

$$\tau(h) = o(h^p).$$

On a de (2.8) $\max_{0\leq i\leq n} |y_i - u_i^*| \leq h\tau(h) \longrightarrow 0$ lorsque $h \longrightarrow 0$.
Il nous reste à montrer que $(u_i^* - u_i) \longrightarrow 0$ lorsque $h \longrightarrow 0$. Écrivons pour cela que

$$u_i^* - u_i = e_{i-1} + h(f(t_{i-1}, y_{i-1}) - f(t_{i-1}, u_{i-1})),$$

comme f est localement lipschitzienne par rapport à sa seconde variable

$$|u_i^* - u_i| \leq (1 + Kh)|e_{i-1}|,$$

ce qui entraine à partir de (2.7) et (2.8)

$$|e_i| \leq \tau h + (1 + Kh)|e_{i-1}|,$$

d'où par récurrence sur i

$$\begin{aligned} |e_i| &\leq \tau h + (1 + Kh)\left(\tau h + (1 + Lh)|e_{i-1}|\right) \\ &= \tau h + (1 + Kh)\tau h + (1 + Kh)^2|e_{i-2}| \\ &\vdots \\ &\leq \tau h\left(1 + \tau h + (1 + Kh) + (1 + Lh)^2 + \cdots + (1 + Kh)^{i-1} + (1 + Kh)^i|e_0|\right) \\ &= \tau h\left(\tfrac{(1+Kh)^i-1}{hK} + (1 + Kh)^i|e_0|\right) \end{aligned} \tag{2.11}$$

d'après le schéma (2.4), $e_0 = y_0 - u_0 = 0$ on obtient donc de l'inégalité (2.11) et puisque $i \leq n$

$$|e_i| \leq \tau(h)\frac{(1 + Kh)^i - 1}{L} \leq \tau(h)\frac{(1 + Kh)^n - 1}{L}. \tag{2.12}$$

En utilisant l'inégalité $(1 + x) \leq e^x$, $\forall x \geq 0$ et puisque $nh = T - t_0$ (2.12) devient

$$|e_i| \leq \tau(h)\frac{e^{K(T-t_0)} - 1}{K}. \tag{2.13}$$

D'après (2.10) $\tau(h) \leq \frac{M}{2}h$, on obtient finalement à partir de (2.13) l'estimation d'erreur

$$\max_{0\leq i\leq n} |e_i| \leq \frac{M}{2}h\frac{e^{K(T-t_0)} - 1}{K}. \tag{2.14}$$

(2.14) entraine naturellement que $\max_{0\leq i\leq n} |e_i| \longrightarrow 0$ lorsque $h \longrightarrow 0$. La méthode d'Euler explicite est donc convergente d'ordre 1. On remarque que

l'ordre de cette méthode coïncide avec son ordre de consistance. On retrouve cette propriété dans de nombreuses méthodes de résolution numériques des équations différentielles.
On démontre de la même manière la convergence de la méthode d'Euler implicite. On peut donc énoncer le résultat de convergence suivant.

Théorème 2.2 Supposons que la solution y du problème (2.1) est $C^2([t_0, T])$. Alors les méthodes d'Euler explicite et implicite sont convergente d'ordre 1.

Exemple Soit $a \in \mathbb{R}$ et considérons le problème de Cauchy suivant

$$\begin{cases} y'(t) = ay(t), & t \in [0, T] \\ y(0) = 1 \end{cases}$$

Notons par $0 = t_0 < \cdots < t_n = T$ une subdivision de l'intervalle $[0, T]$ de pas $h = \frac{T}{n}$. Écrivons le schéma implicite associé au problème précédent

$$\begin{cases} u_i = hau_i + u_{i-1}, & i = 1, \cdots, n \\ u_0 = 1 \end{cases}$$

si $ah \neq 1$, on a alors à partir du schéma précédent

$$u_i = \frac{u_{i-1}}{1 - ah} = \frac{u_{i-2}}{(1 - ah)^2} = \cdots = \frac{u_0}{(1 - ah)^i}$$

comme $u_0 = 1$, on obtient la valeur approchée u_i de $y(t_i)$

$$u_i = (1 - ah)^{-i}$$

d'où en déduit la valeur approchée de y en $t_n = T$

$$y(T) \simeq u_n = (1 - ah)^{-n} = \left(1 - \frac{aT}{n}\right)^{-n}.$$

Il est clair de la relation précédente que $u_n \longrightarrow e^{aT}$ lorsque $n \longrightarrow \infty$.
Remarquons que la solution exacte du problème précédent est $y(t) = e^{at}$, $t \in [0, T]$. On voit sur cet exemple simple que la solution approchée obtenue par le schéma d'Euler implicite converge vers la solution exacte en $t = T$.

Analyse numérique de la convergence de la méthode d'Euler

Considérons le problème de Cauchy suivant

$$\begin{cases} y'(t) = \cos 2y(t), & 0 < t \leq 1 \\ y(0) = 0. \end{cases}$$

dont la solution exacte est $y(t) = \frac{1}{2}\arcsin\left(\frac{e^{4t}-1}{e^{4t}+1}\right)$. Les algorithmes (2.4) et (2.5) s'écrivent dans ce cas

$$\begin{cases} u_0 = v_0 = 0 \\ u_{i+1} = u_i + h\cos 2u_i, & i = 0, \ldots, n \\ v_i = v_{i-1} + h\cos 2v_i, & i = 1, \ldots, n \end{cases}$$

avec $h = \frac{1}{n}$ où u_i dénote la solution approchée par le schéma explicite et v_i celle par le schéma implicite au noeud t_i.
Le script Matlab suivant renvoi la solution approchée par les méthodes d'Euler explicite et implicite.

```
% fun=@(t,x)fun(t,x) fonction definissant
% l'equation differentielle y'(t)=f(t,y(t)), avec
% y(0)=y0.
% tspan=[t0 T] vecteur contenant temps initial t0
% et final T
% e='expl' si on utilise le schema explicite et
% e='impl' si on utilise le schema implicite
function [tt,u]=euler(fun,tspan,n,y0,e)
h=(tspan(2)-tspan(1))/n;
tt=linspace(tspan(1),tspan(2),n+1);
u=zeros(n,1);
u(1)=y0;
if e=='impl';
for i=2:n+1;
% la solution du schema implicite est obtenue par
% la commande fzero
    u(i)=fzero(@(x)h*fun(tt(i),x)-x+u(i-1),u(i-1));
end
%plot(tt,u,'k');
elseif e=='expl';
for i=2:n+1;
     u(i)=u(i-1)+h*fun(tt(i-1),u(i-1));
end
end
%plot(tt,u,'r');
```

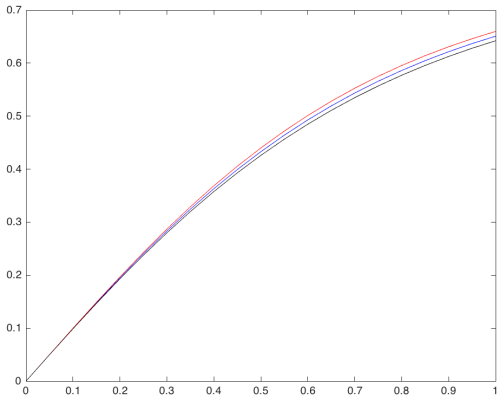

FIGURE 2.1 – En rouge la solution approchée par la méthode d'Euler explicite et en noir celle par le schéma implicite dans le cas $n = 10$. Le graphique en bleu désigne la solution exacte.

La figure 2.1 montre les graphiques de la solution approchée par le schéma explicite (graphe en rouge) et par le schéma implicite (en noir) dans le cas $n = 10$.
Étudions maintenant l'erreur en $t = 1$ des deux méthodes. Pour cela notons par $er_1 = |y(1) - u_n|$ (resp : $er_2 = |y(1) - v_n|$) l'erreur commise en $t = 1$ par le schéma explicite (resp : l'erreur commise par le schéma implicite). Dans la figure 2.2 on a tracé le graphique des deux erreurs en fonction de n variant de 10 j'usqu'a 20. On voit bien que l'erreur dépend linéairement de h comme prévu par le Théorème 2.2.

Remarque Les schémas d'Euler sont un cas particulier de schémas dits à un pas. D'une manière générale les schémas à un pas s'écrivent sous la forme suivante

$$u_{i+1} = u_i + a_0 h f_i + b_{-1} h f_{i+1},\ i \geq 0 \tag{2.15}$$

où $a_0, b_{-1} \in \mathbb{R}$ et $f_i = f(t_i, u_i)$. Lorsque $b_{-1} = 0$ le schéma est explicite sinon il est implicite. Le schéma (2.15) est un schéma dit linéaire car il dépend linéairement de u_i et f_i. Dans le paragraphe suivant on verra d'autres types de méthodes numérique non linéaire.

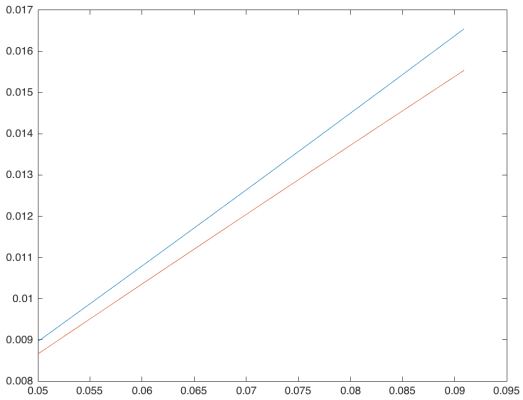

FIGURE 2.2 – Erreur en $t = 1$ des schémas explicite (en rouge) et implicite (en jaune) en fonction de h pour n variant de 10 à 20.

2.3 Méthodes de Runge-Kutta

2.3.1 Méthode de Runge-Kutta à 2 étapes

Revenons au problème de Cauchy (2.1) et intégrons les deux cotés sur l'intervalle (t_i, t_{i+1})

$$y_{i+1} - y_i = \int_{t_i}^{t_{i+1}} f(t, y(t))dt. \tag{2.16}$$

Au lieu d'approcher y' à l'aide des formules de dérivation approchée on pourrai approcher l'intégrale dans (2.16) à l'aide d'une formule de quadrature numérique. Par exemple la formule du trapèze donne

$$y_{i+1} - y_i \simeq \frac{h}{2}\left(f(t_i, y_i) + f(t_{i+1}, y_{i+1})\right)$$

d'où le schéma implicite suivant

$$\begin{cases} u_0 = y_0, \\ K_1 = f(t_i, u_i), \\ K_2 = f\left(t_i + h, u_i + h\left(\frac{1}{2}K_1 + \frac{1}{2}K_2\right)\right) \\ u_{i+1} = u_i + h\left(\frac{1}{2}K_1 + \frac{1}{2}K_2\right), \qquad i = 1, \cdots, n-1 \end{cases} \tag{2.17}$$

appelée méthode de Runge-Kutta implicite à deux étapes. Dans ce schéma à chaque itération i on calcule K_2 en résolvant une équation non linéaire.

La méthode est dite à deux étapes car à chaque itération i le schéma (2.17) utilise deux évaluations de la fonction f.
On représente généralement la méthode de Runge-Kutta (2.17) par son tableau dit de Butcher

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

où les coefficients de ce tableau sont déduit à partir de la forme suivante du schéma (2.17)

$$\begin{aligned} K_1 &= f(t_i + \boxed{0}h, u_i + h(\boxed{0}K_1 + \boxed{0}K_2)), \\ K_2 &= f\left(t_i + \boxed{1}h, u_i + h\left(\boxed{\tfrac{1}{2}}K_1 + \boxed{\tfrac{1}{2}}K_2\right)\right) \\ u_{i+1} &= u_i + h\left(\boxed{\tfrac{1}{2}}K_1 + \boxed{\tfrac{1}{2}}K_2\right) \end{aligned}$$

Lorsque les coefficients du tableau de Butcher sont nuls on omet en général de les écrire, de sorte que le tableau précédent se réécrit avec cette convention

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

On peut encore établir une méthode de Runge-Kutta explicite à partir du schéma (2.17) en estimant la valeur de y_{i+1} à l'aide du schéma d'Euler explicite

$$y_{i+1} \simeq y_i + hf(t_i, y_i),$$

d'où

$$\begin{cases} u_0 = y_0, \\ K_1 = f(t_i, u_i) \\ K_2 = f\left(t_i + h, u_i + hK_1\right) \\ u_{i+1} = u_i + h\left(\frac{1}{2}K_1 + \frac{1}{2}K_2\right), \quad i = 0, \cdots, n-1 \end{cases} \tag{2.18}$$

Le schéma (2.18) est appelé méthode de Runge-Kutta explicite à deux étapes. Son tableau de Butcher est

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

Le résultat suivant montre que les méthodes de Runge-Kutta à 2 étapes explicite et implicite sont consistante d'ordre ≥ 2.

Théorème 2.3 Supposons que f est de classe $C^2(I \times \mathbb{R})$. Alors les méthodes de Runge-Kutta (2.17) et (2.18) sont consistante d'ordre ≥ 2.

Preuve Montrons que la méthode implicite (2.17) est consistante d'ordre 2. Écrivons l'erreur de consistance

$$\tau_i(h) = \frac{y_{i+1} - y_i}{h} - \left(\frac{1}{2}K_1 + \frac{1}{2}K_2\right). \tag{2.19}$$

Le développement limité de y en t_i à l'ordre 3 donne

$$y_{i+1} = y_i + hf\left(t_i, y_i\right) + \tfrac{1}{2}h^2 y''(t_i) + o(h^3)$$

en dérivant la relation $y'(t) = f(t, y(t))$ par rapport à t et en utilisant la formule $B.3$ de l'Annexe B on a

$$\begin{aligned} y''(t_i) &= \frac{\partial f}{\partial t}(t_i, y_i) + y'(t_i)\frac{\partial f}{\partial y}(t_i, y_i) \\ &= \frac{\partial f}{\partial t}(t_i, y_i) + f(t_i, y_i)\frac{\partial f}{\partial y}(t_i, y_i) \\ &= \frac{\partial f}{\partial t}(t_i, y_i) + K_1\frac{\partial f}{\partial y}(t_i, y_i) \end{aligned}$$

d'où

$$y_{i+1} = y_i + hK_1 + \frac{1}{2}h^2\frac{\partial f}{\partial t}(t_i, y_i) + \frac{1}{2}h^2K_1\frac{\partial f}{\partial y}(t_i, y_i) + o(h^3). \tag{2.20}$$

D'autre part un développement limité à l'ordre 2 donne

$$\begin{aligned} K_2 &= f\left(t_i + h, y_i + h\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right)\right) \\ &= f(t_i, y_i) + h\frac{\partial f}{\partial t}(t_i, y_i) + h\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right)\frac{\partial f}{\partial y}(t_i, y_i) \\ &\quad + \tfrac{1}{2}\left(\frac{\partial^2 f}{\partial t^2}(t_i, y_i)h^2 + 2h^2\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right)\frac{\partial^2 f}{\partial t\partial y}(t_i, y_i)\right. \\ &\quad \left. + h^2\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right)^2\frac{\partial^2 f}{\partial y^2}(t_i, y_i)\right) \\ &\quad + o\left\|\left(h, h\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right)\right)\right\|_2^2 \epsilon_1(h), \end{aligned}$$

avec $\epsilon_1(h) \to 0$ lorsque $h \to 0$. Puisque $o\left\|\left(h, h\left(\frac{1}{2}K_1 + \frac{1}{2}K_2\right)\right)\right\|_2^2 = o(h^2)$ et les dérivées partielles seconde de f bornées on arrive à la relation

$$K_2 = K_1 + h\frac{\partial f}{\partial t}(t_i, y_i) + h\left(\frac{1}{2}K_1 + \frac{1}{2}K_2\right)\frac{\partial f}{\partial y}(t_i, y_i) + o(h^2)(1 + \epsilon_1(h)). \tag{2.21}$$

En reportant (2.20) et (2.21) dans (2.19) on obtient

$$\tau_i(h) = \frac{1}{2}hK_1\frac{\partial f}{\partial y}(t_i, y_i) - \frac{h}{2}\left(\frac{K_1}{2} + \frac{K_2}{2}\right)\frac{\partial f}{\partial y}(t_i, y_i) + o(h^2)(1+\epsilon_1(h)). \quad (2.22)$$

Un dernier développement limité à l'ordre 1 implique

$$\begin{aligned} K_2 &= f\left(t_i + h, y_i + h\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right)\right) \\ &= f(t_i, y_i) + h\frac{\partial f}{\partial t}(t_i, y_i) + h\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right)\frac{\partial f}{\partial y}(t_i, y_i) + o(h)\epsilon_2(h) \end{aligned}$$

avec $\epsilon_2(h) \to 0$ lorsque $h \to 0$. En reportant cette dernière égalité dans (2.22) on a

$$\tau_i(h) = -\tfrac{h^2}{4}\frac{\partial f}{\partial t}(t_i, y_i)\frac{\partial f}{\partial y}(t_i, y_i) - \tfrac{h^2}{4}\left(\frac{\partial f}{\partial y}(t_i, y_i)\right)^2\left(\tfrac{1}{2}K_1 + \tfrac{1}{2}K_2\right) + o(h^2).$$

Étant donné que les dérivées partielles $\dfrac{\partial f}{\partial t}(t_i, y_i)$ et $\dfrac{\partial f}{\partial y}(t_i, y_i)$ sont bornées alors

$$\tau_i(h) = o(h^2).$$

La méthode de Runge-Kutta (2.17) est donc consistante d'ordre ≥ 2. □

2.3.2 Méthode de Runge-Kutta à 4 étapes

Introduisons le point milieu $t_{i+1/2} = t_i + \frac{h}{2}$ et approchons l'intégrale (2.16) par la formule de Simpson

$$\int_{t_i}^{t_{i+1}} f(t, y(t))dt \simeq \frac{h}{6}\left(f(t_i, y_i) + 4f(t_{i+1/2}, y_{i+1/2}) + f(t_{i+1}, y_{i+1})\right)$$

ce qui donne

$$y_{i+1} \simeq y_i + \tfrac{h}{6}\left(f(t_i, y_i) + 4f(t_{i+1/2}, y_{i+1/2}) + f(t_{i+1}, y_{i+1})\right) \quad (2.23)$$

Si on veut obtenir un schéma explicite on doit exprimer $y_{i+1/2}$ et y_{i+1} en fonction de y_i. Pour cela écrivons que

$$4f(t_{i+1/2}, y_{i+1/2}) = 2f(t_{i+1/2}, y_{i+1/2}) + 2f(t_{i+1/2}, y_{i+1/2}) \quad (2.24)$$

et approchons le premier terme en utilisant le schéma d'Euler explicite

$$y_{i+1/2} \simeq y_i + \frac{h}{2}f(t_i, y_i). \quad (2.25)$$

Pour le second terme on utilise la méthode d'Euler implicite, posons $K_1 = f(t_i, y_i)$ soit

$$y_{i+1/2} \simeq y_i + \frac{h}{2} f(t_{i+1/2}, y_{i+1/2})$$

d'ou d'après (2.25)

$$\begin{array}{rcl} y_{i+1/2} & \simeq & y_i + \frac{h}{2} f\left(t_i + \frac{h}{2}, y_i + \frac{h}{2} f(t_i, y_i)\right) \\ & = & y_i + \frac{h}{2} f\left(t_i + \frac{h}{2}, y_i + \frac{h}{2} K_1\right) \\ & = & y_i + \frac{h}{2} K_2 \end{array} \tag{2.26}$$

où on a encore posé $K_2 = f\left(t_i + \frac{h}{2}, y_i + \frac{h}{2} K_1\right)$. En reportant (2.25) et (2.26) dans (2.24) on obtient l'approximation

$$4f(t_{i+1/2}, y_{i+1/2}) \simeq 2f\left(t_i + \frac{h}{2}, y_i + \frac{h}{2} K_1\right) + f\left(t_i + \frac{h}{2}, y_i + \frac{h}{2} K_2\right).$$

Il nous reste à approcher le dernier terme $f(t_{i+1}, y_{i+1})$ dans (2.23). Pour cela on approche y_{i+1} par le schéma du point milieu ce qui donne

$$y_{i+1} \simeq y_i + hf\left(t_i + \frac{h}{2}, y_{i+1/2}\right). \tag{2.27}$$

Or d'après (2.26) $y_{i+1/2} \simeq y_i + \frac{h}{2} K_2$. Posons une dernière fois $K_3 = f\left(t_i + \frac{h}{2}, y_i + \frac{h}{2} K_2\right)$, (2.27) entraine donc

$$f(t_{i+1}, y_{i+1}) \simeq f\left(t_i + h, y_i + hK_3\right),$$

d'où le schéma

$$\left\{ \begin{array}{l} u_0 = y_0, \\ K_1 = f(t_i, u_i), \\ K_2 = f\left(t_i + \frac{h}{2}, u_i + \frac{h}{2} K_1\right), \\ K_3 = f\left(t_i + \frac{h}{2}, u_i + \frac{h}{2} K_2\right) \\ K_4 = f\left(t_i + h, u_i + hK_3\right) \\ u_{i+1} = u_i + h\left(\frac{K_1}{6} + \frac{K_2}{3} + \frac{K_3}{3} + \frac{K_4}{6}\right), \quad 1 \leq i \leq n \end{array} \right. \tag{2.28}$$

appelé méthode de Runge-Kutta explicite à 4 étapes. Son tableau de Butcher s'écrit

$$\begin{array}{c|cccc} \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & & \frac{1}{2} & & \\ 1 & & & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

Remarque On verra dans le paragraphe suivant que la méthode de Runge-Kutta à 4 étapes est convergente avec un ordre de convergence égal à 4.

Exemple Calculons une solution approchée en $t = 1$ du problème suivant par la méthode de Runge-Kutta à 4 étapes explicite en prenant $h = \frac{1}{2}$

$$\begin{cases} y'(t) = y^2 - t, & t \in [0,1] \\ y(0) = 1 \end{cases}$$

Les noeuds sont $\{t_0 = 0, t_1 = \frac{1}{2}, t_2 = 1\}$ l'algorithme de la méthode de Runge-Kutta à 4 étapes s'écrit

$$\begin{cases} u_0 = 1, \\ u_{i+1} = u_i + \frac{1}{2}\left(\frac{K_1}{6} + \frac{K_2}{3} + \frac{K_3}{3} + \frac{K_4}{6}\right), & i = 0, 1 \\ K_1 = u_i^2 - t_i, \\ K_2 = \left(u_i + \frac{1}{4}K_1\right)^2 - t_i - \frac{1}{4}, \\ K_3 = \left(u_i + \frac{1}{4}K_2\right)^2 - t_i - \frac{1}{4}, \\ K_4 = \left(u_i + \frac{1}{2}K_3\right)^2 - t_i - \frac{1}{2}, \end{cases}$$

d'où le tableau des valeurs suivant

t_i	u_i	K_1	K_2	K_3	K_4
0	1	1	1.3125	1.5139	2.5869
0.5	1.77	2.6328	5.1461	8.5922	35.7971
1	7.2622	—	—	—	—

on a alors l'approximation $y(1) \simeq 7.2622$.

2.3.3 Forme générale d'une méthode de Runge-Kutta

D'une manière générale une méthode de Runge-Kutta à s étapes s'écrit sous la forme

$$\begin{cases} K_l = f\left(t_i + c_l h, u_i + h\sum_{j=1}^{s} a_{lj}K_j\right), & 1 \le l \le s, \\ u_{i+1} = u_i + h\sum_{j=1}^{s} b_j K_j, & i \ge 0 \end{cases} \tag{2.29}$$

La méthode se caractérise par son tableau de Butcher

$$\begin{array}{c|c} c & A \\ \hline & b \end{array}$$

où $A = (a_{ij})_{s\times s}$, $c = (c_1, \cdots, c_s)^\top$ et $b = (b_1, \cdots, b_s)$. On supposera toujours que la condition suivante est vérifiée

$$\sum_{j=1}^{s} a_{ij} = c_i, \quad i = 1, \ldots, s. \tag{2.30}$$

La méthode est dite d'ordre s car à chaque étape de temps elle nécessite s évaluations de la fonction f. Si $a_{ij} = 0$ pour $i \leq j$ la méthode est explicite, si $a_{ij} = 0$ pour $i < j$ elle est dite semi-implicite sinon elle est implicite.

Proposition 2.2 Supposons que $f : I \times \mathbb{R} \to \mathbb{R}$ est localement lipschitzienne par rapport à sa seconde variable. Alors si h est suffisamment petit la méthode de Runge-Kutta (2.29) admet à chaque itération i une solution unique u_i.

Preuve On montre la proposition par récurrence sur i. Supposons qu'a l'itération i la méthode de Runge-Kutta admet une solution u_i et notons par $\varphi : \mathbb{R}^n \to \mathbb{R}^n$ l'application définie par

$$\varphi(K_1, \ldots, K_l) = (\varphi_1(K_1, \ldots, K_l), \ldots, \varphi_s(K_1, \ldots, K_l)),$$
$$\varphi_l(K_1, \ldots, K_l) = f\left(t_i + c_l h, u_i + h\sum_{j=1}^{s} a_{lj} K_j\right), \qquad l = 1, \ldots, s.$$

L'idée est de montrer que φ est une application contractante. Pour cela posons $K = (K_1, \ldots, K_s)$, $\tilde{K} = (\tilde{K}_1, \ldots, \tilde{K}_s)$, $a = \max\limits_{1\leq i,j\leq s} |a_{ij}|$ et écrivons que

$$\begin{aligned} |\varphi_l(K) - \varphi_l(\tilde{K})| &= \left| f\left(t_i + c_l h, u_i + h\sum_{j=1}^{s} a_{lj} K_j\right) - \right. \\ &\left. f\left(t_i + c_l h, u_i + h\sum_{j=1}^{s} a_{lj} \tilde{K}_j\right)\right| \\ &\leq Lha \sum_{j=1}^{s} |K_j - \tilde{K}_j| \end{aligned}$$

d'où en utilisant par exemple la norme $\|.\|_1$ de $\mathbb{R}^n$

$$\|\varphi(K) - \varphi(\tilde{K})\|_1 \leq sLha\|K - \tilde{K}\|_1,$$

donc si $sLha < 1$, φ est une contraction elle admet donc un point fixe $(K_1, \ldots, K_s)$ à l'itération i et par suite d'après la deuxième équation de (2.29) on obtient la solution u_{i+1} à l'itération $i + 1$. □

Le critère suivant permet de savoir si une méthode de Runge-Kutta est consistante à partir de son tableau de Butcher.

Théorème 2.4 Supposons que f est de classe $C^1(I \times \mathbb{R})$. Alors la méthode de Runge-Kutta à s étapes (2.29) est consistante si et seulement si

$$\sum_{i=1}^{s} b_i = 1.$$

Preuve L'erreur de consistance s'écrit dans le cas de la méthode de Runge-Kutta (2.29)

$$\tau_i(h) = \frac{y_{i+1} - y_i}{h} - \sum_{j=1}^{s} a_{lj} K_j, \quad 1 \leq l \leq s, \tag{2.31}$$

avec

$$K_l = f\left(t_i + c_l h, y_i + h \sum_{j=1}^{s} a_{lj} K_j\right), \quad 1 \leq l \leq s,$$

où y_i dénote la valeur exacte de la solution en t_i. On a le développement limité suivant de y_i à l'ordre 2

$$\begin{array}{rcl} y_{i+1} & = & y_i + h y'(t_i) + o(h^2) \\ y_{i+1} & = & y_i + h f(t_i, y_i) + o(h^2). \end{array} \tag{2.32}$$

D'autre part un développpment limité à l'odre 1 de f (voir formule B.1 de l'annexe B) donne

$$\begin{array}{rcl} K_l & = & f\left(t_i + c_l h, y_i + h \sum_{j=1}^{s} a_{lj} K_j\right) \\ & = & f(t_i, y_i) + c_l h \dfrac{\partial f}{\partial t}(t_i, y_i) + h \dfrac{\partial f}{\partial y}(t_i, y_i) \sum_{j=1}^{s} a_{lj} K_j \\ & & + o\left(\left\| \left(c_l h, h \sum_{j=1}^{s} a_{lj} K_j\right) \right\|_2\right) \epsilon(h), \end{array} \tag{2.33}$$

avec $\epsilon(h) \to 0$ lorsque $h \to 0$. En reportant (2.32) et (2.33) dans (2.31) on obtient

$$\begin{array}{rcl} \tau_i(h) & = & f(t_i, y_i) - \sum_{j=1}^{s} b_j \left(f(t_i, y_i) + h c_j \dfrac{\partial f}{\partial t}(t_i, y_i) + \right. \\ & & \left. h \sum_{k=1}^{s} a_{jk} K_k \dfrac{\partial f}{\partial y}(t_i, y_i) + o(h^2) + o\left(\left\| \left(c_j h, h \sum_{k=1}^{s} a_{jk} K_k\right) \right\|_2\right) \epsilon(h) \right) \\ & = & f(t_i, y_i) \left(1 - \sum_{j=1}^{s} b_j\right) - h \dfrac{\partial f}{\partial t}(t_i, y_i) \sum_{j=1}^{s} c_j - h \dfrac{\partial f}{\partial y}(t_i, y_i) \times \\ & & \sum_{k=1}^{s} a_{jk} K_k + o(\|(h,h)\|_2) \epsilon(h) \\ & = & f(t_i, y_i) \left(1 - \sum_{j=1}^{s} b_j\right) + o(h). \end{array}$$

On voit donc que la méthode est consistante si et seulement si $\sum_{j=1}^{s} b_j = 1$. □

Le Théorème 2.4 nous permet de savoir si une méthode de Runge-Kutta est consistante mais ne donne aucune information sur son ordre de consistance. Le résultat suivant permet de comparer l'ordre de consistance d'une méthode de Runge-Kutta avec son nombre d'étapes s. Pour la preuve voir [1].

Théorème 2.5 L'ordre de consistance p d'une méthode de Runge-Kutta à s étapes vérifie toujours $p \leq s$.

Remarque D'après les Théorèmes 2.5 et 2.3 l'ordre de consistance des méthodes de Runge-Kutta à 2 étapes (2.17) et (2.18) est 2. Quant à la méthode de Runge-Kutta à 4 étapes on peut montrer qu'elle est d'ordre 4. D'autre part on démontre (voir [1]) qu'au delà de 4 étapes il n'existe pas de méthodes de Runge-Kutta dont l'ordre est égale au nombres d'étapes s.

2.3.4 Zero-stabilité

Dans ce paragraphe on s'intéresse à la notion de stabilité des méthodes de Runge-Kutta. L'introduction de la notion stabilité est motivée par l'étude de la sensibilité des schémas numérique par rapport aux erreurs d'arrondis.

Définition 2.5 On dira que la méthode de Runge-Kutta (2.29) est zéro-stable s'il existe $h_0 > 0$ et $C > 0$ tels que pour tout h vérifiant $0 < h < h_0$ et $\forall \varepsilon > 0$, si $|\delta_i| < \varepsilon$, $0 \leq i \leq n$ alors

$$|z_i - u_i| \leq C\varepsilon, \quad i = 0, \ldots, n, \tag{2.34}$$

où u_i et z_i sont les solutions des problèmes (2.35) et (2.36) respectivement

$$\begin{cases} u_0 = y_0, \\ u_{i+1} = u_i + h \sum_{j=1}^{s} b_j K_j^u, & i = 0, \ldots, n-1 \\ K_l^u = f\left(t_i + c_l h, u_i + h \sum_{j=1}^{s} a_{lj} K_j^u\right), & l = 1, \ldots, s \end{cases} \tag{2.35}$$

$$\begin{cases} z_0 = y_0 + \delta_0, \\ z_{i+1} = z_i + h\left(\sum_{j=1}^{s} b_j K_j^z + \delta_{i+1}\right), \qquad i = 0, \ldots, n-1 \\ K_l^z = f\left(t_i + c_l h, z_i + h\sum_{j=1}^{s} a_{lj} K_j^z\right), \quad l = 1, \ldots, s \end{cases} \tag{2.36}$$

Autrement dit une petite perturbation sur la donnée initiale et la fonction f engendre une petite perturbation sur la solution.
Le théorème suivant nous donne une condition nécéssaire de zéro-stabilité de la méthode de Runge-Kutta (2.29).

Théorème 2.6 Supposons que $f : I \times \mathbb{R} \to \mathbb{R}$ est localement lipschitzienne par rapport à sa seconde variable. Alors la méthode de Runge-Kutta (2.29) est zéro-stable.

Preuve Posons $w_i = z_i - u_i$, on a alors

$$w_{i+1} = w_i + h\sum_{j=1}^{s} b_j (K_j^z - K_j^u) + h\delta_{i+1}. \tag{2.37}$$

D'un autre coté comme f est localement lipschitzienne par rapport a sa seconde variable, on a pour h_0 et δ_0 assez petit

$$\begin{aligned} |K_l^z - K_l^u| &= \left| f\left(t_i + c_l h, z_i + hz_i + h\sum_{j=1}^{s} a_{lj} K_j^z\right) \right. \\ &\quad \left. - f\left(t_i + c_l h, u_i + h\sum_{j=1}^{s} a_{lj} K_j^u\right) \right| \\ &\leq L|w_i| + Lh\sum_{j=1}^{s} |a_{lj}||K_j^z - K_j^u| \\ &\leq L|w_i| + Lha\sum_{j=1}^{s} |K_j^z - K_j^u| \end{aligned}$$

où on a posé $a = \max_{1 \leq i \leq n} |a_{ij}|$. En sommant cette dernière inégalité sur l on obtient

$$\sum_{l=1}^{s} |K_l^z - K_l^u| \leq sL|w_i| + sLha\sum_{j=1}^{s} |K_j^z - K_j^u|$$

d'où

$$\sum_{l=1}^{s} |K_l^z - K_l^u| \leq \frac{Ls}{1 - sLha}|w_i|. \tag{2.38}$$

Posons $b = \max_{1\leq j\leq s} |b_j|$, on a partir de (2.37)

$$|w_{i+1}| \leq |w_i| + bh \sum_{j=1}^{s} |K_j^z - K_j^u| + h|\delta_{i+1}|,$$

l'inégalité (2.38) appliquée à cette dernière inégalité donne

$$\begin{aligned}
|w_{i+1}| &\leq |w_i| + \frac{hbLs}{1-sLha}|w_i| + h|\delta_{i+1}| \\
&= |w_i|\left(1 + \frac{hbLs}{1-sLha}\right) + h|\delta_{i+1}| \\
&\leq |w_{i-1}|\left(1 + \frac{hbLs}{1-sLha}\right)^2 + h|\delta_i|\left(1 + \frac{hbLs}{1-sLha}\right) \\
&\quad + h|\delta_{i+1}| \\
&\vdots \\
&\leq |w_0|\left(1 + \frac{hbLs}{1-sLha}\right)^{i+1} + h|\delta_1|\left(1 + \frac{hbLs}{1-sLha}\right)^{i} + \ldots + \\
&\quad + h|\delta_i|\left(1 + \frac{hbLs}{1-sLha}\right) + h|\delta_{i+1}| \\
&\leq |\delta_0|\left(1 + \frac{hbLs}{1-sLha}\right)^{n} + h|\delta_1|\left(1 + \frac{hbLs}{1-sLha}\right)^{n} + \ldots + \\
&\quad + h|\delta_i|\left(1 + \frac{hbLs}{1-sLha}\right)^{n} + h|\delta_{i+1}| \\
&= \left(1 + \frac{hbLs}{1-sLha}\right)^{n} \{|\delta_0| + h|\delta_1| + \ldots + h|\delta_i|\} + h|\delta_{i+1}|.
\end{aligned} \tag{2.39}$$

D'après la définition $|\delta_i| \leq \varepsilon$, $\forall i = 0, \ldots, n$ d'où d'après (2.39)

$$|w_{i+1}| \leq \left(1 + \frac{hbLs}{1-sLha}\right)^{n} \{\varepsilon + ih\varepsilon\} + h\varepsilon \tag{2.40}$$

or puisque $\left(1 + \frac{hbLs}{1-sLha}\right)^n \leq e^{TbLs}$ et $ih \leq nh = T$ on obtient finalement de (2.40) la majoration

$$|w_{i+1}| \leq \left(e^{TbLs}(1+T) + h_0\right)\varepsilon, \quad \forall i = 0, \ldots, n-1 \tag{2.41}$$

la méthode de Runge-Kutta (2.29) est donc zéro-stable. □

Intéressons maintenant à la convergence de la méthode de Runge-Kutta (2.29). On a le résultat fondamental suivant.

Théorème 2.7 Supposons que $f : I \times \mathbb{R} \to \mathbb{R}$ est localement lipschitzienne par rapport à sa seconde variable et notons par $\tau(h)$

l'erreur de consistance globale de la méthode de Runge-Kutta (2.29). Alors on l'estimation d'erreur suivante

$$|y_i - u_i| \leq Te^{bTLs}\tau(h), \quad i = 1, \ldots, n. \tag{2.42}$$

L'estimation d'erreur (2.42) implique que si la méthode de Runge-Kutta est consistance d'ordre p alors elle est convergente d'ordre p.

Preuve Posons $w_i = y_i - u_i$ où y_i est la solution exacte du problème de Cauchy (2.1) en t_i. On a de (2.31) et de la seconde équation de (2.29)

$$h\tau_i(h) = y_{i+1} - y_i - h\sum_{j=1}^{s} b_j K_j^y$$
$$0 = u_{i+1} - u_i - h\sum_{j=1}^{s} b_j K_j^u$$

d'ou

$$w_{i+1} = w_i + h\sum_{j=1}^{s} b_j(K_j^y - K_j^u) + h\tau_i(h)$$

cette dernière relation entraine en posant $b = \max\limits_{1\leq j\leq s} |b_j|$

$$|w_{i+1}| \leq |w_i| + hb\sum_{j=1}^{s} |K_j^y - K_j^u| + h|\tau_i(h)| \tag{2.43}$$

en appliquant la relation (2.39) à (2.43) avec $\delta_0 = w_0 = y_0 - u_0 = 0$ on obtient

$$\begin{aligned} |w_{i+1}| &\leq \left(1 + \frac{hbLs}{1 - sLha}\right)^n \{h|\tau_1(h)| + \ldots + h|\tau(h)_{i-1}|\} + h|\tau(h)_i| \\ &\leq e^{TbLs}(nh\tau(h) \\ &= Te^{TbLs}\tau(h). \end{aligned}$$

□

Remarque Le Théorème 2.7 est un cas particulier d'un théorème plus général qui énonce qu'une méthode numérique à un pas consistante est convergente si et seulement si elle est zéro-stable. Les Théorèmes 2.4 et 2.7 nous donnent deux conditions nécessaires et suffisantes pour la convergence de la méthode de Runge-Kutta (2.29).

2.3.5 Construction d'une méthode de Runge-Kutta

Dans ce paragraphe on va indiquer une méthode permettant de déterminer toutes les méthodes de Runge-Kutta explicites ayant un nombre d'étapes $s \geq 1$ donné. Pour simplifier on s'intéressera aux cas $s = 1, 2$.

Méthode à 1 étape

Le tableau de Butcher d'une méthode de Runge-Kutta à 1 étape explicite s'écrit d'après la condition du Théorème 2.7 et (2.30)

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

qui n'est autre que la méthode d'Euler explicite (2.4). Il y a donc une seule méthode de Runge-Kutta à 1 étape.

Méthode à 2 étapes

Le tableau de Butcher d'une méthode à 2 étapes explicite s'écrit

$$\begin{array}{c|cc} c_1 & 0 & 0 \\ c_2 & a & 0 \\ \hline & b_1 & b_2 \end{array}$$

Si on s'intéresse aux méthodes consistantes d'après le Théorème 2.4 on a $b_1 + b_2 = 1$. D'autre part la condition (2.30) entraine $c_1 = 0$ et $c_2 = a$.
Écrivons maintenant le développement de la solution exacte y_{i+1} à l'ordre 2

$$y_{i+1} = y_i + hy'(t_i) + \tfrac{h^2}{2}y''(t_i) + o(h^3). \tag{2.44}$$

En dérivant la relation $y'(t) = f(t, y(t))$ par rapport à t on obtient

$$\begin{array}{rcl} y''(t) & = & \frac{\partial f}{\partial t}(t,y) + y'\frac{\partial f}{\partial y}(t,y) \\ & = & \frac{\partial f}{\partial t}(t,y) + f(t,y)\frac{\partial f}{\partial y}(t,y). \end{array} \tag{2.45}$$

En reportant (2.45) dans (2.44) on arrive a

$$y_{i+1} = y_i + hf + \tfrac{h^2}{2}(f_t + ff_y) + o(h^3) \tag{2.46}$$

où on a posé $f = f(t_i, y_i)$, $f_t = \frac{\partial f}{\partial t}(t_i, y_i)$ et $f_y = \frac{\partial f}{\partial y}(t_i, y_i)$.
Notons maintenant par u_{i+1} la solution approchée par la méthode de Runge-Kutta à 2 étapes obtenue à partir de y_i i.e

$$u_{i+1} = y_i + h(b_1K_1 + b_2K_2) \tag{2.47}$$

les développement limité de K_1 et K_2 à l'ordre 2 donnent puisque $c_1 = 0$ et $a_{11} = a_{12} = a_{22} = 0$

$$\begin{array}{rcl} K_1 & = & f(t_i, y_i) = f, \\ K_2 & = & f(t_i + c_2 h, y_i + ahK_1) \\ & = & f(t_i + c_2 h, y_i + ahf) \\ & = & f + c_2 h f_t + ahf f_y + o(h^2) \end{array}$$

d'où en reportant les expressions précédentes de K_1 et K_2 dans (2.47) on obtient en tenant compte de $c_2 = a$

$$\begin{array}{rcl} u_{i+1} & = & y_i + h(b_1 f + b_2 f + c_2 b_2 h f_t + b_2 ahf f_y + o(h^2)) \\ & = & y_i + hf(b_1 + b_2) + b_2 ah^2(f_t + f f_y) + o(h^3) \end{array} \tag{2.48}$$

en supposant que les développement de u_{i+1} et y_{i+1} coïncident on alors en comparant (2.46) et (2.48)

$$b_1 + b_2 = 1, \quad c_2 a = \tfrac{1}{2}. \tag{2.49}$$

La condition (2.49) montre qu'il y a une infinité de méthodes Runge-Kutta à 2 étapes. Parmi ces méthodes citons la méthode dite d'Euler modifiée dont le tableau de Butcher est

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

et la méthode d'Euler améliorée donnée par son tableau de Butcher suivant

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

2.3.6 Estimations d'erreurs et principe des solveurs Matlab

La formule d'estimation d'erreur (2.42) montre que si la méthode (2.29) est consistante d'ordre p alors $\max\limits_{1\leq i\leq n} |y(t_i) - u_i| = o(h^p)$, la méthode est donc convergente d'ordre p. Malheureusement l'estimation (2.42) ne permet pas déterminer le pas h à partir d'une tolérance donnée étant donné que l'erreur

de consistance $\tau_i(h)$ dépend elle même de la solution exacte y (et des ses dérivées) qui n'est pas connue. On peut contourner ce problème en utilisant différents artifices. L'une des techniques les plus répondues consiste à obtenir une estimation de l'erreur locale en utilisant deux méthodes de Runge-Kutta ayant la même matrice A (donc le même nombres d'étapes s) et le même vecteur c. Ceci entraine que les deux méthodes ont les même coefficients K_i. Décrivons plus en détail cette technique.
Notons par $y(t_i)$ la solution exacte du problème (2.1) au noeud t_i et supposons qu'on utilise une méthode de Runge-Kutta à s étapes et d'ordre p donné par

$$\begin{cases} K_l = f\left(t_i + c_l h, u_i + h\sum_{j=1}^{s} a_{lj}K_j\right), & 1 \leq l \leq s, \\ u_{i+1} = u_i + h\sum_{j=1}^{s} b_j K_j, & i \geq 0. \end{cases} \tag{2.50}$$

Supposons qu'à l'étape i on dispose d'une valeur approchée u_i obtenue par la méthode (2.44) et notons par $y(t_{i+1})$ la valeur exacte de la solution en partant de $y(t_i) = u_i$ i.e

$$\begin{cases} y'(t) = f(t, y), \\ y(t_i) = u_i \end{cases} \tag{2.51}$$

Supposons qu'on se donne une seconde méthode de Runge-Kutta d'ordre $(p+1)$ ayant le tableau de Butcher suivant

$$\begin{array}{c|c} c & A \\ \hline & \widehat{b} \end{array} \tag{2.52}$$

où $A = (a_{ij})_{s\times s}$ et $c = (c_i)_s$. Remarquons que les méthodes (2.44) et (2.46) ont les mêmes K_i car elles ont la même matrice A et le même vecteur c. Notons par $\widehat{u}_{i+1}$ la valeur approchée de la solution à l'étape $i+1$ obtenue en partant de u_i c'est à dire que

$$\widehat{u}_{i+1} = u_i + h\sum_{j=1}^{s} \widehat{b}_j K_j, \tag{2.53}$$

On a alors de (2.45) et (2.47)

$$\widehat{u}_{i+1} - y(t_{i+1}) = y(t_i) - y(t_{i+1}) + h\sum_{j=1}^{s} \widehat{b}_j K_j \tag{2.54}$$

où

$$K_l = f\left(t_i + c_l h, y(t_i) + h\sum_{j=1}^{s} a_{lj}K_j\right), \quad 1 \leq l \leq s.$$

Le membre de droite de (2.48) n'est autre que l'erreur de consistance $h\tau_i(h)$ de la méthode (2.46). Comme $h\tau_{i+1}(h) = o(h^{p+2})$ on déduit que

$$\widehat{u}_{i+1} - y(t_{i+1}) = o(h^{p+2}). \tag{2.55}$$

Écrivons d'autre part que

$$\widehat{u}_{i+1} - u_{i+1} = (y(t_{i+1}) - u_{i+1}) + (\widehat{u}_{i+1} - y(t_{i+1}))$$

d'où d'après (2.49)

$$\widehat{u}_{i+1} - u_{i+1} = y(t_{i+1}) - u_{i+1} + o(h^{p+2}). \tag{2.56}$$

La relation (2.50) fournit un estimateur de la solution. En effet comme

$$\widehat{u}_{i+1} - u_{i+1} = h\sum_{j=1}^{s}(\widehat{b}_j - b_j)K_j,$$

un estimateur de la solution locale u_{i+1} est alors donné par

$$h\sum_{j=1}^{s}(\widehat{b}_j - b_j)K_j < \epsilon \tag{2.57}$$

où ϵ est une tolérance donnée. On notera cette méthode par son tableau de Butcher dit augmenté qui rassemble les deux méthodes

$$\begin{array}{c|c} c & A \\ \hline & b \\ \hdashline & \widehat{b} \\ \hline & \\ & \widehat{b} - b \end{array}$$

Le critère d'estimation d'erreur (2.51) n'est possible que si les deux méthodes ont les mêmes coefficients K_i et ne nécessite, à chaque pas d'itération i, que le calcul des K_i et de la somme $\sum_{j=1}^{s}(\widehat{b}_j - b_j)K_j$ à la différence que si on avait utilisé deux méthodes différentes mais avec des K_i différents ceci aurait nécessité, en plus du calcul de chaque coefficient K_l et $\widehat{K}_l$, le calcul de deux sommes $\sum_{j=1}^{s} b_j K_j$ et $\sum_{j=1}^{s} \widehat{b}_j \widehat{K}_j$ différentes ce qui représente donc un gain en temps de calcul. On voit donc que l'utilisation de deux méthodes jouissant de cette propriété à exactement le meme cout

en temps de calcul que l'utilisation d'une seule méthode, en l'occurrence dans notre exemple la méthode d'ordre p.
La plupart des solveurs Matlab sont basés sur l'estimation d'erreur (2.51) et utilisent donc deux méthodes de Runge-Kutta d'ordre p et $p+1$.
L'algorithme le plus connu et qui est implémenté dans le solveur `ode45`, est celui de *Dormand & Prince* (1980) (voir [1]) qui utilisent deux méthodes de Runge-Kutta d'ordre respectivement 4 et 5. On le désigne dans la littérature scientifique par RK45. Le Tableau de Butcher augmenté de cette méthode est donné par

$$
\begin{array}{c|ccccccc}
0 & & & & & & & \\
\frac{1}{5} & \frac{1}{5} & & & & & & \\
\frac{3}{10} & \frac{3}{40} & \frac{9}{40} & & & & & \\
\frac{4}{5} & \frac{44}{45} & -\frac{56}{15} & \frac{32}{9} & & & & \\
\frac{8}{9} & \frac{19372}{6561} & -\frac{25360}{2187} & \frac{64448}{6561} & -\frac{212}{769} & & & \\
1 & \frac{9017}{3168} & -\frac{355}{33} & \frac{46732}{5247} & \frac{49}{176} & -\frac{5103}{18656} & & \\
1 & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} & \\
\hline
 & \frac{35}{384} & 0 & \frac{500}{1113} & \frac{125}{192} & -\frac{2187}{6784} & \frac{11}{84} & 0 \\
\hdashline
 & \frac{5179}{57600} & 0 & \frac{7571}{16695} & \frac{393}{640} & -\frac{92097}{339200} & \frac{187}{2100} & \frac{1}{40} \\
\hline
 & -\frac{71}{57600} & 0 & \frac{71}{16695} & -\frac{71}{1920} & \frac{17253}{339200} & -\frac{22}{525} & \frac{1}{40}
\end{array}
$$

c'est donc une méthode à 7 étapes. Comme le vecteur $b^\top$ est identique à la dernière ligne de A et que son dernier coefficient est nul alors
$\sum_{j=1}^{6} \widehat{b}_j K_j = \sum_{j=1}^{6} a_{7j} K_j$, on voit donc que la méthode est en réalité à 6 étapes.
Les méthodes de Runge-Kutta jouissant d'une telle propriété sont dite FSAL (first same as last).
Ils existent d'autres algorithmes basés sur un couple de méthodes Runge-Kutta, parmi ces méthodes citons :

- `ode23` implémente un couple de méthodes RK explicites dit de Bogacki-Shampine
- `ode23tb` implémente une méthode de RK implicite
- `ode15s` implémente une méthode multi-pas
- `ode113` implémente une méthode dite d'Adams-Moulton-Bashforth

Test numérique de la méthode RK45

Dans but de tester la méthode RK45 considérons le problème suivant du a *Dormand & Prince*

$$\begin{cases} y' = y\cos t, \quad t \in (0, 20], \\ y(0) = 1 \end{cases}$$

dont la solution exacte est $y(t) = e^{\sin t}$. Le script Matlab suivant implémente cette méthode.

```
% la fonction Matlab rk45 retourne la solution approchee
% u du probleme y'=f(t,y) avec la donnee initiale
% y(t0)=y0 sur l intervalle [t0,T] par l'algorithme de
% Dormand & Prince (voir le cours M. Kouche, Introduction
% a la mehtode des differences finies).
% les arguments d entrees sont:
% fun=@(t,y)fun(t,y) fonction vectorisee de deux variables
% (t,y) definissant l'equation differentielle
% y'(t)=f(t,y(t)), avec y(0)=y0.
% tspan=[t0 T] vecteur contenant le temps initial t0
% et final T
% er la tolerance avec laquelle est calculee la solution
% approchee u
% les arguments de sortie sont:
% la solution approchee u
% le vecteur t contenant les noeuds ti
% le nombre d'iterations n

function [t,u,n]=rk45(fun,y0,tspan,er)
format long
a=[0 0 0 0 0 0 0;
    1/5 0 0 0 0 0 0;
    3/40 9/40 0 0 0 0 0;
    44/45 -56/15 32/9 0 0 0 0;
    19372/6561 -25360/2187 64448/6561 -212/769 0 0 0;
    9017/3168 -355/33 46732/5247 49/176 -5103/18656 0 0;
    35/384 0 500/1113 125/192 -2187/6784 11/84 0];
b=[35/384 0 500/1113 125/192 -2187/6784 11/84 0];
c=[0 1/5 3/10 4/5 8/9 1 1];
bb=[71/57600 0 71/16695 -71/1920 17253/339200 ...
    -22/525 1/40];
n=fix(tspan(2)-tspan(1))+1;
```

```
u(1:n+1)=0;u(1)=y0;
h=(tspan(2)-tspan(1))/n;
t(1:n+1)=0;
t=linspace(tspan(1),tspan(2),n+1);
k=zeros(7,1);
%k=fun(t+h*c',u+h*a*k);
for i=1:n;
    k=fun(t(i)+h*c',u(i)+h*a*k);
u(i+1)=u(i)+h*dot(b,k);
end
while h*dot(abs(bb),k)>=er;
    n=n+1;
    t(1:n+1)=0;
    u(1:n+1)=0;u(1)=y0;
    h=(tspan(2)-tspan(1))/n;
    t=linspace(tspan(1),tspan(2),n+1);
    k=zeros(7,1);
    for i=1:n;
    k=fun(t(i)+h*c',u(i)+h*a*k);
    u(i+1)=u(i)+h*dot(b,k);
    end
end
```

Un premier test en prenant $er = 10^{-2}$ donne le graphique de la figure 2.3 qui montre que l'erreur e_i aux noeuds t_i est légèrement supérieure à la tolérance er au voisinage de $t = 20$. Notons que dans notre exemple $\max_{1\leq i\leq n} |e_i| = 0.01346$ ce qui est une bonne approximation. Le nombre d'itérations requis est de $n = 329$.
Une version plus optimisée du script `rk45` est codée dans le solveur Matlab `ode45`.

2.4 Méthodes à un pas non linéaires

Les méthodes de Runge-Kutta sont un cas particulier de méthodes dites à un pas non linéaires. D'une manière générale on appelle méthode numérique

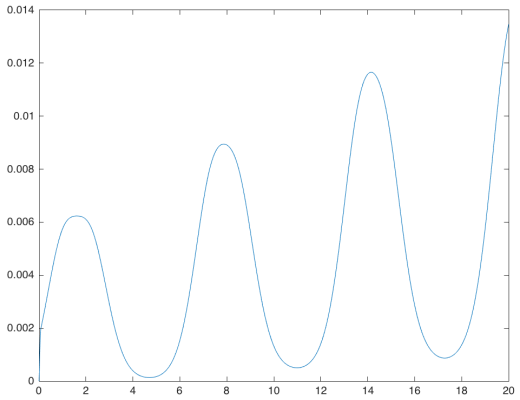

FIGURE 2.3 – Graphique représentant l'erreur aux noeuds t_i dans le cas $er = 10^{-2}$.

à un pas une méthode de la forme

$$\begin{cases} u_0 = y_0, \\ u_{i+1} = u_i + h\Phi(t_i, u_i, u_{i+1}, f_i, f_{i+1}, h), \quad i = 0, \ldots, n-1 \end{cases} \tag{2.58}$$

où $f_i = f(t_i, u_i)$. La fonction Φ est appelée fonction d'incrément et dépend de f à travers f_i. L'erreur de consistance de la méthode (2.52) en t_{i+1} s'écrit alors

$$h\tau_{i+1}(h) = y_{i+1} - y_i - h\Phi(t_i, y_i, y_{i+1}, f(t_{i+1}, y_{i+1}), f(t_i, y_i), h), \quad i = 1, \ldots, n-1, \tag{2.59}$$

de même on définit l'erreur de consistance globale par

$$\tau(h) = \max_{0 \le i \le n} |\tau_i(h)|.$$

Exemple La méthode d'Euler (2.15) est un cas particulier de la méthode (2.52) avec $\Phi_f(t_i, u_i, u_{i+1}, f_i, f_{i+1}, h) = u_i + a_0 f_i + b_{-1} f_{i+1}$.

Donnons maintenant la définition de la zéro-stabilité dans le cas de la méthode (2.52).

Définition 2.6 On dira que la méthode (2.52) est zéro-stable s'il existe $h_0 > 0$ et $C > 0$ tels que pour tout h vérifiant $0 < h < h_0$ et $\forall \varepsilon > 0$, si $|\delta_i| < \varepsilon$, $0 \le i \le n$ alors

$$|z_i - u_i| \le C\varepsilon, \quad i = 0, \ldots, n,$$

où u_i et z_i sont les solutions des deux problèmes suivants respectivement

$$\begin{cases} u_0 = y_0, \\ u_{i+1} = u_i + h\Phi(t_i, u_i, u_{i+1}, f_i, f_{i+1}, h), \quad i = 0, \ldots, n-1, \end{cases}$$

$$\begin{cases} z_0 = y_0 + \delta_0, \\ z_{i+1} = z_i + h\left(\Phi(t_i, z_i, z_{i+1}, f_i, f_{i+1}, h) + \delta_{i+1}\right), \quad i = 0, \ldots, n-1. \end{cases}$$

Lorsque la fonction d'incrément Φ ne dépend pas de u_{i+1} et f_{i+1} la méthode est dite explicite. Le schéma (2.52) devient alors

$$\begin{cases} u_0 = y_0, \\ u_{i+1} = u_i + h\Phi(t_i, u_i, f_i, h), \quad i = 0, \ldots, n-1 \end{cases} \tag{2.60}$$

Dans ce cas on dispose du critère suivant qui assure la zéro-stabilité de la méthode.

Théorème 2.8 Supposons que la fonction d'incrément Φ est lipschitzienne en u_i c'est à dire qu'il existe $L > 0$ et $h_0 > 0$ tel que $\forall h \in (0, h_0]$ on a

$$|\Phi(t_i, z_i, f(t_i, z_i), h) - \Phi(t_i, u_i, f(t_i, u_i), h)| \leq L|z_i - u_i|, \quad i = 0, \ldots, n. \tag{2.61}$$

Alors la méthode explicite à un pas (2.54) est zéro-stable.

Preuve Posons $w_i = z_i - u_i$, on obtient à partir de la définition 2.6 et puisque Φ est lipschitzienne

$$\begin{aligned} |w_{i+1}| &\leq |w_i| + h\,|\Phi(t_i, z_i, f(t_i, z_i), h) - \Phi(t_i, u_i, f(t_i, u_i), h)| \\ &\leq |w_i| + h(L|w_i| + |\delta_{i+1}|) \\ &= (1 + hL)|w_i| + h|\delta_{i+1}| \\ &\leq (1 + hL)^2|w_{i-1}| + (1 + hL)h|\delta_i| + h|\delta_{i+1}| \\ &\vdots \\ &\leq (1 + hL)^{i+1}|w_0| + h(1 + hL)^i|\delta_1| + \ldots + h(1 + hL)|\delta_i| \\ &\quad + h|\delta_{i+1}| \\ &\leq (1 + hL)^{i+1}|\delta_0| + h(1 + hL)^i|\delta_1| + \ldots + h(1 + hL)|\delta_i| \\ &\quad + h|\delta_{i+1}| \end{aligned} \tag{2.62}$$

or puisque $|\delta_j| \leq \varepsilon$ pour tout $j = 0, \ldots, n$, il s'ensuit d'après (2.56)

$$|w_{i+1}| \leq \varepsilon(1 + hL)^{i+1} + \varepsilon\left(\frac{(1 + hL)^{i+1} - 1}{L}\right). \tag{2.63}$$

l'inégalité $(1 + hL)^{i+1} \leq e^{(i+1)hL} \leq e^{TL}$, et (2.57) impliquent alors

$$|w_{i+1}| \leq \varepsilon\left(e^{TL} + \frac{(e^{TL} - 1}{L}\right),$$

la méthode est donc zéro-stable. □

Finalement on donne un résultat de convergence de la méthode (2.54).

Théorème 2.9 Supposons que l'hypothèse du Théorème 2.8 est satisfaite. Alors si la méthode (2.54) est consistante elle est convergente. De plus l'ordre de convergence est égal à l'ordre de consistance.

Preuve Elle est similaire à celle du Théorème 2.7. Définissons $e_i = y_i - u_i$ l'erreur d'approximation. On a à partir de (2.53)

$$h\tau_i(h) = e_{i+1} - e_i - h\left(\Phi(t_i, y_i, f(t_i, y_i), h) - \Phi(t_i, u_i, f(t_i, u_i), h)\right)$$

d'où puisque Φ est lipschitzienne et que $e_0 = y_0 - u_0 = 0$

$$\begin{aligned} |e_{i+1}| &\leq |e_i| + h\,|\Phi(t_i, y_i, f(t_i, y_i), h) - \Phi(t_i, u_i, f(t_i, u_i), h)| \\ &\quad + h|\tau_i(h)| \\ &\leq (1 + hL)|e_i| + h\tau(h) \\ &\vdots \\ &\leq h\tau(h)\left((1 + hL)^i + \ldots + (1 + hL) + 1\right) \\ &= \tau(h)\left(\frac{(1 + hL)^{i+1} - 1}{L}\right) \\ &\leq \tau(h)\frac{e^{TL} - 1}{L}. \end{aligned} \tag{2.64}$$

L'estimation d'erreur (2.58) montre que si $\tau(h) = o(h^p)$ alors $\max_{1\leq i\leq n} |e_i| = o(h^p)$, ceci démontre le théorème. □

2.5 Méthodes Multi-Pas

2.5.1 Méthode BDF2 "Backword Difference Formula"

Revenons au problème de Cauchy (2.1) et approchons $y'(t_i)$ par la formule de dérivation approchée à trois points à gauche de t_i (A.15) de l'annexe A

$$y'(t_{i+1}) \simeq \frac{3y_{i+1} - 4y_i + y_{i-1}}{2h}$$

on obtient alors le schéma suivant

$$\begin{cases} u_0 = y_0, \\ u_{i+1} = \dfrac{2}{3}hf_{i+1} + \dfrac{4}{3}u_i - \dfrac{1}{3}u_{i-1}, \quad i \geq 1 \end{cases} \tag{2.65}$$

avec $f_i = f(t_i, u_i)$. Le schéma (2.65) est dit schéma de différentiation rétrograde ou BDF2 ("*Backword Difference Formula*") à 2 pas. C'est un schéma implicite dit à 2 pas car u_{i+1} dépend de u_i et de u_{i-1}. Pour calculer u_2 on aura besoin de u_1 qui n'est pas donné par le schéma. Il nous faudra utiliser un schéma dit "d'amorçage" qui permet d'estimer u_1 à partir de u_0. N'importe quel schéma à un pas peut servir à estimer u_1.

Exemple Considérons le problème de Cauchy suivant

$$\begin{cases} y'(t) = 2y(t), & t \in [0,1] \\ y(0) = 1 \end{cases}$$

Calculons une valeur approchée de $y(1)$ par le schéma BDF2 en prenant les noeuds $\left\{0, \frac{1}{2}, 1\right\}$. Pour estimer u_1 utilisons le schéma d'Euler explicite comme schéma d'amorçage

$$u_1 = hf_0 + u_0 = \frac{1}{2}(2u_0) + u_0 = 2$$

d'où d'après (2.65)

$$\begin{array}{rcl} u_2 & = & \frac{2}{3}hf(t_2, u_2) + \frac{4}{3}u_1 - \frac{1}{3}u_0 \\ & = & \frac{2}{3}\left(\frac{1}{2}\right)(2u_0) + \frac{8}{3} - \frac{1}{3} \\ & = & \frac{2}{3}u_2 + \frac{7}{3} \end{array}$$

d'où l'on tire $u_2 = 7$, on obtient donc l'approximation $y(1) \simeq 7$. Comme la solution exacte est $y(t) = e^{2t}$ on voit que l'erreur commise est $er = |y(2) - u_2| = 0.3891$.

2.5.2 Méthodes d'*Adams*

Méthodes d'*Adams-Bashforth*

Revenons à la relation (2.16) et écrivons le polynôme d'interpolation P_1 de $f(t, y(t))$ aux noeuds $(t_{i-1}, y_{i-1}), (t_i, y_i)$

$$P_1(t) = f_{i-1} + [f_{i-1}, f_i](t - t_i),$$

où $[f_{i-1}, f_i] = \frac{f_i - f_{i-1}}{h}$ sont les différences divisées d'ordre 1. En approchant f dans l'intégrale (2.16) par P_1 on obtient

$$\begin{array}{rcl} y_{i+1} - y_i & \simeq & \int_{t_i}^{t_{i+1}} P_1 dt \\ & = & hf_{i-1} + \dfrac{f_i - f_{i-1}}{h}\left(2h - \dfrac{1}{2}h\right) \\ & = & h\left(\dfrac{3}{2}f_i - \dfrac{1}{2}f_{i-1}\right) \end{array}$$

d'où l'on tire le schéma

$$u_{i+1} = u_i + h\left(\frac{3}{2}f_i - \frac{1}{2}f_{i-1}\right), \quad 1 \leq i \leq n-1 \tag{2.66}$$

Ce schéma est appelé schéma d'*Adams-Bashforth*. C'est un schéma explicite à deux pas car u_{i+1} dépend de u_i et u_{i-1}.

De la même manière si on interpole f aux noeuds t_{i-2}, t_{i-1}, t_i par son polynôme de degré 2 noté P_2 on obtient

$$P_2(t) = f_{i-2} + [f_{i-1}, f_i](t - t_{i-2}) + [f_{i-2}, f_{i-1}, f_i](t - t_{i-2})(t - t_{i-1}),$$

avec $[f_{i-2}, f_{i-1}, f_i] = \dfrac{f_i + f_{i-2} - 2f_{i-1}}{2h^2}$. En remplaçant dans (2.16) f par son polynôme d'interpolation P_2 on a

$$\begin{aligned} y_{i+1} - y_i &\simeq hf_{i-2} + \frac{f_i - f_{i-1}}{h}\left.\frac{(t-t_{i-2})^2}{2}\right|_{t_i}^{t_{i+1}} \\ &\quad + \frac{f_i + f_{i-2} - 2f_{i-1}}{2h^2}\left((t-t_{i-1})\left.\frac{(t-t_{i-2})^2}{2}\right|_{t_i}^{t_{i+1}} - \left.\frac{(t-t_{i-2})^3}{6}\right|_{t_i}^{t_{i+1}}\right) \\ &= h\left(\frac{5}{12}f_{i-2} - \frac{4}{3}f_{i-1} + \frac{23}{12}f_i\right) \end{aligned}$$

on a finalement le schéma

$$u_{i+1} = u_i + h\left(\frac{5}{12}f_{i-2} - \frac{4}{3}f_{i-1} + \frac{23}{12}f_i\right), \quad 2 \leq i \leq n-1 \tag{2.67}$$

qui est un schéma explicite à trois pas. On l'appelle schéma d'Adams-Bashforth à trois pas.

D'une manière générale on obtient les méthodes d'Adams-Bashforth à p pas ($p \geq 1$) en approchant f dans l'intégrale (2.16) par son polynôme d'interpolation sur les noeuds $t_{i-p}, t_{i-p+1} \ldots, t_i$.

Méthodes d'*Adams-Moulton*

Pour obtenir des schémas implicites par la méthode d'Adams on approche f par son polynôme d'interpolation P_2 aux noeuds t_{i-1}, t_i, t_{i+1}. P_2 qui s'écrit alors

$$P_2(t) = f_{i-1} + [f_{i-1}, f_i](t - t_{i-1}) + [f_{i-1}, f_i, f_{i+1}](t - t_{i-1})(t - t_i)$$

d'où en reportant l'expression de P_2 dans (2.16) on arrive a

$$\begin{aligned} y_{i+1} - y_i &\simeq hf_{i-1} + \frac{f_i - f_{i-1}}{h} \left. \frac{(t-t_{i-1})^2}{2} \right|_{t_i}^{t_{i+1}} \\ &= +\frac{f_{i+1} + f_{i-1} - 2f_i}{2h^2} \left((t-t_{i-1}) \left. \frac{(t-t_i)^2}{2} \right|_{t_i}^{t_{i+1}} - \left. \frac{(t-t_i)^3}{6} \right|_{t_i}^{t_{i+1}} \right) \\ &= h\left(-\frac{1}{12} f_{i-1} + \frac{2}{3} f_i + \frac{5}{12} f_{i+1} \right), \end{aligned}$$

d'où le schéma

$$u_{i+1} = u_i + h\left(-\frac{1}{12} f_{i-1} + \frac{2}{3} f_i + \frac{5}{12} f_{i+1} \right), \quad 2 \leq i \leq n-1 \tag{2.68}$$

appelé schéma d'*Adams-Moulton*. C'est un schéma implicite à deux pas. Les schémas d'*Adams-Moulton* à p pas ($p \geq 1$) sont des schémas implicites, ils sont obtenus en approchant f dans (2.16) par son polynôme d'interpolation obtenu avec les noeuds $t_{i-p+1}, \ldots, t_{i+1}$.

Test numérique de la méthode d'*Adams*

Le script Matlab suivant calcule la solution approchée du problème (2.1) par une méthode à $p+1$ pas.

```
% Le script renvoi la solution approchee u de
% l equation differentiele y'(t)=f(t,y(t)) avec y(t0)=y0
% par une methode a (p+1) pas.
% On utilise le schema d Euler explicite comme
% schema d amorcage. Les donnees d entrees sont:
% e='expl' si on utilise la methode explicite
% e='impl' si on utilise la methode implicite
% la fonction vectorisee f
% la donne initiale y0
% le vecteur tspan contenant le temps initial t0 et
% final T
% le nombre de noeuds n
% les vecteurs a et b contiennent les
% coefficients de la methode multi-pas:
% dans le cas de la methode implicite la derniere
% composante de b est le coefficient du terme
% hf(t_{i+1},u_{i+1}).
```

```
function [t,u]=adams(a,b,f,y0,tspan,n,e)
p=length(a)-1;
t=linspace(tspan(1),tspan(2),n+1);
h=t(2)-t(1);
t=t';
u=zeros(n+1,1);
u(1)=y0;
% initialiser les variables manquantes
% a l'aide du schema d'Euler explicite
u(2:p+1)=u(1:p)+h*f(t(1:p),u(1:p));
if e=='expl';
for i=p+1:n;
    u(i+1)=dot(a,u(i:-1:i-p))+h*dot(b,f(t(i:-1:i-p)...
        ,u(i:-1:i-p)));
end
elseif e=='impl';
  for i=p+1:n;
  u(i+1)=fzero(@(x)dot(a,u(i:-1:i-p))+h*dot(b(1:end-1),...
      f(t(i:-1:i-p),u(i:-1:i-p)))...
     -x+h*b(end)*f(t(i+1),x),u(i));
  end
end
```

Considérons le problème de Cauchy suivant

$$\begin{cases} y' = 1 - y^2, & t \in (0,1] \\ y(0) = 0 \end{cases}$$

dont la solution exacte est $y(t) = \frac{e^{2t}-1}{e^{2t}+1}$. Notons par $\epsilon(h) = |u - y(t)|$ le vecteur erreur aux noeuds $t = (t_0, \ldots, t_n)^\top$, où $u = (u_0, \ldots, u_n)^\top$ désigne la solution approchée par une méthode mutli-pas. La figure 2.4 représente les graphiques de l'erreur en fonction du pas h par les méthodes d'Adams-Bashforth en rouge et d'Adams-Moulton en bleu dans le cas $n = 10$ et la figure 2.5 le graphique de la même erreur par les deux méthodes dans le cas $n = 20$. Ces deux graphiques montrent que la méthode implicite d'Adams-Moulton et plus rapide que celle d'Adams-Bashforth malgré un cout en temps de calcul plus important.
Le script ci-dessous

```
% Ce script calcule l ordre de convergnece p de chacune
% des methodes d Adams–Bashforth et d Adams–Moulton a 2 pas.
% l ordre de convergence p est calcule a partir des
```

```
% vecteurs er1 et er2 representant chacun les erreurs
% obtenues par les deux methodes en t=1 pour n=2^i
% ou i varie de 1 jusqu a 10.
f=@(t,y)(1-y.^2);
% la solution excate est:
y=@(t)(exp(2*t)-1)./(exp(2*t)+1);
% les coefficients des deux methodes:
a=[1  0];
bb=[2/3  -1/12  5/12];
b=[3/2  -1/2];
y0=0;
tspan=[0  1];
for i=1:10;
[t,u]=adams(a,b,f,y0,tspan,2^i,'expl');
[t,uu]=adams(a,bb,f,y0,tspan,2^i,'impl');
% l erreur en t=1 par les deux methodes
er1(i)=abs(u(end)-y(1));
er2(i)=abs(uu(end)-y(1));
end
p1=log(er1(1:end-1)./er1(2:end))/log(2);
p2=log(er2(1:end-1)./er2(2:end))/log(2);
```

renvoi les ordres p_1 et p_2 des deux méthodes et montrent que $p_1 = 2$ et $p_2 = 3$. La méthode d'Adams-Moulton est donc d'ordre 3 ce qui explique sa précision par rapport à celle d'Adams-Bashforth qui est elle d'ordre 2 comme observé dans les figures 2.4 et 2.5.

2.5.3 Forme générale des schémas Multi-Pas

D'une manière générale les schémas linéaires à $(p+1)$ pas, avec $p \geq 0$, s'écrivent sous la forme suivante

$$u_{i+1} = \sum_{j=0}^{p} a_j u_{i-j} + h \sum_{j=0}^{p} b_j f_{i-j} + h b_{-1} f_{i+1}, \quad p \leq i \leq n-1, \tag{2.69}$$

où $(a_j)_{0\leq j\leq p}$, $(b_j)_{-1\leq j\leq p}$, sont des coefficients donnés. Ce schéma est dit à $(p+1)$ pas. Pour démarrer le schéma il nous faudra connaitre les valeurs initiales $u_0, \cdots, u_p$. Comme $u_0 = y_0$ les autres valeurs $u_1, \cdots, u_p$ ne sont pas connues et doivent être estimées à partir d'un schéma d'amorçage. Lorsque $b_{-1} = 0$ le schéma est explicite sinon il est implicite.

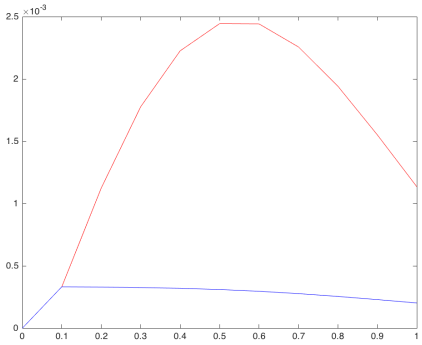

FIGURE 2.4 – *Tracé de l'erreur en fonction de h dans le cas n = 10 : En rouge par la méthode d'Adams-Bashforth et en bleu par celle d'Adams-Moulton*

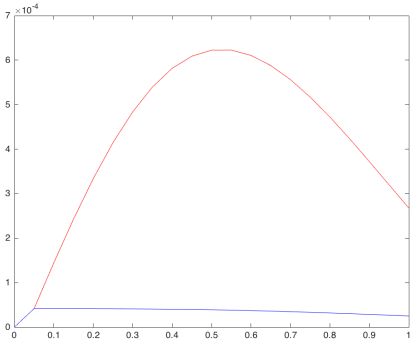

FIGURE 2.5 – *Tracé de l'erreur en fonction de h dans le cas n = 20 : En rouge la méthode d'Adams-Bashforth et en bleu la méthode d'Adams-Moulton*

Les schémas de type (2.69) sont dit linéaires car il dépendent linéairement des u_j et f_j à la différence des méthodes de Runge-Kutta qui sont elles des méthodes à un pas non linéaire.

Consistance

L'erreur de consistance de la méthode (2.69) au noeud t_{i+1} s'écrit

$$h\tau_{i+1}(h) = y_{i+1} - \sum_{j=0}^{p} a_j y_{i-j} - h \sum_{j=-1}^{p} b_j f(t_{i-j}, y_{i-j}),$$

qu'on peut réécrire sous la forme

$$h\tau_{i+1}(h) = y_{i+1} - \sum_{j=0}^{p} a_j y_{i-j} - h \sum_{j=-1}^{p} b_j y'(t_{i-j}, y_{i-j}), \quad p \leq i \leq n-1, \tag{2.70}$$

où y désigne la solution exacte du problème de Cauchy (2.1). L'erreur de consistance globale est définie de manière analogue par

$$\tau(h) = \max_{1 \leq i \leq n} |\tau_i(h)|. \tag{2.71}$$

Théorème 2.10 Supposons que la solution exacte y du problème de Cauchy (2.1) vérifie $y \in C^{q+1}(I)$. Alors la méthode multi-pas (2.69)

est consistante d'ordre $\geq q$ si et seulement si

$$\sum_{j=0}^{p} a_j = 1, \quad \sum_{j=0}^{p} (-j)^k a_j + k \sum_{j=-1}^{p} (-j)^{k-1} b_j = 1, \quad k = 1, \ldots, q. \tag{2.72}$$

Preuve Écrivons le développement limité de y_{i-j} autour de y_i on a

$$y_{i-j} = \sum_{k=0}^{q} \frac{(-j)^k h^k}{k!} y_i^{(k)} + o(h^{q+1}),$$

puis celui de y'_{i-j} autour de y'_i

$$y'_{i-j} = \sum_{k=0}^{q-1} \frac{(-j)^k h^k}{k!} y_i^{(k+1)} + o(h^q),$$

en reportant ces deux expressions dans (2.70) on obtient

$$\begin{aligned} h\tau_{i+1}(h) &= y_{i+1} - \sum_{j=0}^{p} a_j \left\{ \sum_{k=0}^{q} \frac{(-j)^k h^k}{k!} y_i^{(k)} \right\} \\ &\quad -h \sum_{j=-1}^{p} b_j \left\{ \sum_{k=0}^{q-1} \frac{(-j)^k h^k}{k!} y_i^{(k+1)} \right\} + o(h^{q+1}) \\ &= y_{i+1} - y_i \sum_{j=0}^{p} a_j - \sum_{k=1}^{q} \frac{h^k}{k!} y_i^{(k)} \left[\sum_{j=0}^{p} a_j (-j)^k \right. \\ &\quad \left. +k \sum_{j=-1}^{p} b_j (-j)^{k-1} \right] + o(h^{q+1}) \end{aligned}$$

d'où l'expression suivante de $\tau_{i+1}(h)$

$$\begin{aligned} \tau_{i+1}(h) &= \frac{y_{i+1} - y_i}{h} + y_i \left(\frac{1 - \sum_{j=0}^{p} a_j}{h} \right) - \sum_{k=1}^{q} \frac{h^{k-1}}{k!} y_i^{(k)} \left[\sum_{j=0}^{p} a_j (-j)^k \right. \\ &\quad \left. +k \sum_{j=-1}^{p} b_j (-j)^{k-1} \right] + o(h^q) \end{aligned} \tag{2.73}$$

Un dernier développement limité de y_{i+1} à l'ordre q donne

$$y_{i+1} - y_i = \sum_{k=1}^{q} \frac{h^k}{k!} y_i^{(k)} + o(h^{q+1}).$$

En reportant cette dernière expression dans (2.73) on arrive a

$$\begin{aligned}\tau_{i+1}(h) &= y_i\left(\frac{1-\sum_{j=0}^{p} a_j}{h}\right)+\sum_{k=1}^{q}\left\{\frac{h^{k-1}}{k!}y_i^{(k)}\left(1-\sum_{j=0}^{p}(-j)^k a_j\right.\right.\\ &\left.\left.-k\sum_{j=-1}^{p}(-j)^{k-1}b_j\right)\right\}+o(h^q).\end{aligned} \tag{2.74}$$

(2.74) entraine que la méthode est consistante d'ordre $\geq q$ si et seulement si

$$\sum_{j=0}^{p} a_j = 1, \quad \sum_{j=0}^{p}(-j)^k a_j + k\sum_{j=-1}^{p}(-j)^{k-1}b_j = 1, \quad k = 1,\ldots,q.$$

En particulier la méthode (2.69) est consistante (d'ordre $q \geq 1$) si et seulement si

$$\sum_{j=0}^{p} a_j = 1, \quad \sum_{j=0}^{p} b_j = 1 + \sum_{j=0}^{p} j a_j. \tag{2.75}$$

□

Exemple Revenons aux méthodes d'Adams vues au paragraphes 2.5.2. On a dans la cas de la méthode d'Adams-Bashforth à 2 pas $a_0 = 1$, $a_1 = 0$, $b_{-1} = 0$, $b_0 = \frac{3}{2}$ et $b_1 = -\frac{1}{2}$. On peut alors vérifier que

$$\begin{aligned}&\sum_{j=0}^{1} a_j = 1,\\ k=1;\quad &\sum_{j=0}^{1} -j a_j + \sum_{j=-1}^{1} b_j = 1,\\ k=2;\quad &\sum_{j=0}^{1} j^2 a_j - 2\sum_{j=-1}^{1} j b_j = 1,\\ k=3;\quad &-\sum_{j=0}^{1} j^3 a_j + 3\sum_{j=-1}^{1} j^2 b_j = \tfrac{3}{2} \neq 1,\end{aligned}$$

La méthode d'Adams-Bashforth est donc d'ordre 2. Pour ce qui est de la méthode d'Adams-Moulton à 2 pas on a

$a_0 = 1,\ a_1 = 0,\ b_{-1} = \frac{5}{12},\ b_0 = \frac{2}{3},\ b_1 = -\frac{1}{12}$ donc

$$\begin{aligned}
&\sum_{j=0}^{1} a_j = 1,\\
k = 1;\quad &-\sum_{j=0}^{1} j a_j + \sum_{j=-1}^{1} b_j = 1,\\
k = 2;\quad &\sum_{j=0}^{1} j^2 a_j - 2\sum_{j=-1}^{1} j b_j = 1,\\
k = 3;\quad &-\sum_{j=0}^{1} j^3 a_j + 3\sum_{j=-1}^{1} j^2 b_j = 1,\\
k = 4;\quad &-\sum_{j=0}^{1} j^4 a_j - 4\sum_{j=-1}^{1} j^3 b_j = 2 \neq 1,
\end{aligned}$$

la méthode d'Adams-Moulton est donc bien d'ordre 3 comme on a l'a vu dans la partie simulation numérique.

Zéro-stabilité

On définit la zéro-stabilité des méthodes multi-pas de manière analogue à celles des méthodes à un pas. Plus précisément on a la définition suivante.

Définition 2.7 On dira que la méthode multi-pas (2.69) est zéro-stable s'il existe $h_0 > 0$ et $C > 0$ tels que pour tout h vérifiant $0 < h < h_0$ et $\forall \varepsilon > 0$, si $|\delta_i| < \varepsilon$, $0 \leq i \leq n$ alors

$$|z_i - u_i| \leq C\varepsilon, \quad i = 0, \ldots, n,$$

où u_i et z_i sont les solutions respectives des deux problèmes suivants

$$\begin{cases} u_k = y_k,\ k = 0, \ldots, p \\ u_{i+1} = \sum_{j=0}^{p} a_j u_{i-j} + h \sum_{j=0}^{p} b_j f^u_{i-j} + h b_{-1} f^u_{i+1}, \quad p \leq i \leq n-1, \end{cases}$$

$$\begin{cases} z_k = y_k + \delta_k,\ k = 0, \ldots, p \\ z_{i+1} = \sum_{j=0}^{p} a_j z_{i-j} + h \sum_{j=0}^{p} b_j f^z_{i-j} + h b_{-1} f^z_{i+1} + h\delta_{i+1}, \quad p \leq i \leq n-1, \end{cases}$$

où $f^u_i = f(t_i, u_i)$, $f^z_i = f(t_i, z_i)$ et $(y_k)_{0 \leq k \leq p}$ sont les valeurs initiales obtenues par un ou plusieurs schémas d'amorçages.

Le polynôme

$$\pi(r) = r^{p+1} - \sum_{j=0}^{p} a_j r^{p-j} \tag{2.76}$$

est appelé polynôme caractéristique de la méthode multi-pas (2.69).

Définition 2.5 Notons par $r_0, \cdots, r_p$ les racines réelles ou complexes du polynôme caractéristique (2.76) de la méthode multi-pas (2.69). On dira que la méthode (2.69) satisfait la condition des racines si

$$\begin{array}{ll} |r_j| \leq 1, & j = 0, \ldots, p \\ \pi'(r_j) \neq 0, & \text{si } |r_j| = 1 \end{array} \tag{2.77}$$

Le résultat suivant montre que la stabilité d'une méthode multi-pas est déterminée par les coefficients $(a_j)_{0 \leq j \leq p}$ de sa partie linéaire.

Théorème 2.11 Supposons que f est localement lipschitzienne par rapport à sa seconde variable. Alors la méthode multi-pas (2.69) est zéro-stable si et seulement si la condition des racines (2.77) est vérifiée.

Preuve On montrera seulement que la condition est nécessaire. Pour la preuve complète voir [7].
Supposons pour cela que la méthode est zéro-stable et considérons l'équation différentielle

$$\begin{cases} y'(t) = 0, & t \in (t_0, T], \\ y(t_0) = y_0. \end{cases}$$

dont la solution exacte est $y(t) = y_0$, $\forall t \in (0, T]$. Le schéma (2.69) s'écrit dans ce cas

$$\begin{cases} u_{i+1} = \sum_{j=0}^{p} a_j u_{i-j}, & p \leq i \leq n-1, \\ u_k = v_k, & 0 \leq k \leq p \end{cases} \tag{2.78}$$

où les $(v_k)_k$ sont les données initiales du schéma. La solution de l'équation aux différences (2.78) s'écrit en vertu de la relation (1.16)

$$u_i = \sum_{j=0}^{p} v_j \psi_i^{(j)}, \quad i \geq p+1 \tag{2.79}$$

où les $\psi_i^{(j)}$ sont un système fondamental de solutions de l'équation aux différences (2.78). D'après (1.18) on peut exprimer les $\psi_i^{(j)}$ en fonction des racines du polynôme caractéristique (2.76) comme suit

$$\psi_i^{(j)} = \left(\sum_{s=0}^{m_j - 1} \alpha_{s,j} i^s \right) r_j^i, \quad 0 \leq j \leq p, \quad i \geq p+1. \tag{2.80}$$

Supposons maintenant que la condition des racines n'est pas satisfaite. Il existe alors au moins une racine r_j du polynôme

caractéristique π tel que $|r_j| > 1$ si cette racine est simple ou bien $|r_j| \geq 1$ si elle est multiple. D'après (2.79) et (2.80) il s'en suit que u_i est non bornée ce qui contredit la zéro-stabilité de la méthode étant donné que la solution exacte est bornée. □

Exemple Revenons une seconde fois aux méthodes d'Adams-Bashforth et d'Adams-Moulton à 2 pas. Le polynôme caractéristique des deux méthodes s'écrit $\pi(r) = r^2 - 1$ qui admet les deux racines simples $r_0 = -1$ et $r_1 = 1$. On voit donc que la condition des racines est satisfaite. Les deux méthodes sont donc zéro-stable.

Plus généralement toutes les méthodes d'Adams à $p+1$ pas de la forme

$$u_{i+1} = u_i + h \sum_{j=0}^{p} b_j f_{i-j} + h b_{-1} f_{i+1}, \quad p \leq i \leq n-1,$$

sont zéro-stables puisqu'elles admettent le polynôme caractéristique $\pi(r) = r^{p+1} - 1$ dont les $p+1$ racines simples sont données par $r_j = e^{i\frac{2j\pi}{p+1}}$, $j = 0, \ldots, p$.

Théorème 2.12 (Lax-Richtmyer) Supposons que f est localement lipschitzienne par rapport à sa seconde variable. Alors une méthode multi-pas du type (2.69) consistante est convergente si et seulement si elle est zéro-stable.

Remarque Ce théorème fondamental dit "d'équivalence", qu'on a démontré dans le cas des schémas à un pas, reste vrai même pour des méthodes multi-pas non linéaires. Pour la preuve voir [5, 7].

2.6 Cas des systèmes différentiels

Considérons le cas d'un système différentiel du premier ordre

$$\left\{ \begin{array}{lcll} y_1'(t) & = & f_1(t, y_1, \cdots, y_m), & \\ & \vdots & & t \in [t_0, T] \\ y_m'(t) & = & f_m(t, y_1, \cdots, y_m), & \end{array} \right. \tag{2.81}$$

avec les conditions initiales

$$y_1(t_0) = y_{01}, \quad \cdots, \quad y_m(t_0) = y_{0m}. \tag{2.82}$$

Considérons une grille uniforme $t_0 < t_1 < \cdots < t_n = T$ de l'intervalle $[t_0, T]$ de pas $h = t_{i+1} - t_i$. Pour calculer une solution approchée du problème (2.81)-(2.82) aux nœuds t_i on peut appliquer une méthode numérique qu'on a déjà vu à chaque équation du système.

Méthode d'Euler

Notons par $u_{i,j}$ une valeur approchée de y_j (la j ème équation) en t_i. La méthode d'Euler explicite s'écrit dans ce cas

$$\begin{cases} u_{0,j} = y_{0,j}, & j = 1, \cdots, m \\ u_{i+1,j} = u_{i,j} + hf_j(t_i, u_{i,1}, \cdots, u_{i,m}), & 0 \leq i \leq n-1, \quad j = 1, \cdots, m \end{cases}$$

De même la méthode d'Euler implicite appliquée au système (2.81)-(2.82) donne

$$\begin{cases} u_{0,j} = y_{0,j}, & j = 1, \cdots, m \\ u_{i+1,j} = u_{i,j} + hf_j(t_i, u_{i+1,1}, \cdots, u_{i+1,m}), & 0 \leq i \leq n-1, \quad j = 1, \cdots, m \end{cases}$$

Méthode de Runge-Kutta

Écrivons maintenant le schéma de la méthode de Runge-Kutta (2.18) dans le cas du système différentiel (2.81). Notons par $K_{1,j}$ et $K_{2,j}$ les deux coefficients de la méthode de Runge-Kutta correspondant à la $j$ème équation. On a alors le schéma

$$\begin{cases} u_{0,j} = y_{j0}, & 1 \leq j \leq m, \\ K_{1,j} = f_j(t_i, u_{i,1}, \cdots, u_{i,m}), & 1 \leq j \leq m, \\ K_{2,j} = f_j(t_i + h, u_{i,1} + hK_{1,1}, \cdots, u_{i,m} + hK_{1,m}), & 1 \leq j \leq m, \\ u_{i+1,j} = u_{i,j} + h\left(\frac{1}{2}K_{1,j} + \frac{1}{2}K_{2,j}\right), & 0 \leq i \leq n-1, \ 1 \leq j \leq m, \end{cases} \tag{2.83}$$

Cas d'une équation différentielle d'ordre $m \geq 1$

Considérons maintenant le cas de l'équation différentielle d'ordre $m \geq 1$ suivante

$$\begin{cases} y^{(m)}(t) = f(t, y, y', \cdots, y^{(m-1)}), & t \in [t_0, T], \\ y(t_0) = y_0, \cdots, y^{(m-1)}(t_0) = y_{m-1} \end{cases} \tag{2.84}$$

Écrivons le problème (2.84) sous la forme d'un système différentiel d'ordre m en posant $w_1 = y, \cdots, w_m = y^{(m-1)}$, d'où le système différentiel suivant équivalent au problème (2.84)

$$\begin{cases} w_1'(t) &= w_2(t), \\ w_2'(t) &= w_3(t), \\ &\vdots \\ w_m'(t) &= f(t, w_1, \cdots, w_m) \end{cases} \tag{2.85}$$

avec les conditions initiales

$$w_1(t_0) = y_0, \cdots, w_m(t_0) = y_m. \tag{2.86}$$

Ainsi on peut toujours approcher la solution d'une équation différentielle d'ordre $m \geq 1$ en discrétisant un système différentiel de taille m.

Exemple [10]

Le déplacement $x(t)$ d'un système oscillant composé d'une masse et d'un ressort soumis à une force de frottement proportionnelle à la vitesse est décrit par l'équation différentielle du second ordre suivante

$$\begin{cases} x''(t) + 5x'(t) + 6x(t) = 0, \quad t \in [0, 2] \\ x(0) = 1, \ x'(0) = 0 \end{cases}$$

Résolvons ce problème par la méthode de Runge-Kutta à 2 étapes explicite (2.83) en prenant les nœuds $\{0, \frac{1}{2}, 1, \frac{3}{2}, 2\}$.
Posons $w_1 = x$, $w_2 = x'$ d'où le système différentiel

$$\begin{cases} w_1' = w_2, \\ w_2' = -5w_2 - 6w_1 \\ w_1(0) = 1, \\ w_2(0) = 0 \end{cases} \tag{2.87}$$

Le schéma de Runge-Kutta (2.83) appliqué au cas du système (2.87) donne

$$\begin{cases} u_{0,1} = 1, \\ u_{0,2} = 0, \\ K_{1,1} = u_{i,2}, \\ K_{1,2} = -5u_{i,2} - 6u_{i,1}, \\ K_{2,1} = u_{i,2} + \frac{1}{2}K_{1,2}, \\ K_{2,2} = -5\left(u_{i,2} + \frac{1}{2}K_{1,2}\right) - 6\left(u_{i,1} + \frac{1}{2}K_{1,1}\right), \\ u_{i+1,1} = u_{i,1} + \frac{1}{2}\left(\frac{1}{2}K_{1,1} + \frac{1}{2}K_{2,1}\right), \\ u_{i+1,2} = u_{i,2} + \frac{1}{2}\left(\frac{1}{2}K_{1,2} + \frac{1}{2}K_{2,2}\right), \end{cases}$$

d'où le tableau des valeurs suivant

t_i	$u_{i,1}$	$u_{i,2}$	$K_{1,1}$	$K_{1,2}$	$K_{2,1}$	$K_{2,2}$
0	1	0	0	-6	-3	9
0.5	0.25	0.75	0.75	-5.25	-1.875	5.625
1	-0.0312	0.8438	0.8438	-4.0312	-1.1719	3.5156
1.5	-0.1133	0.7148	0.7148	-2.8945	-0.7324	2.1973
2	-0.1177	0.5405	—	—	—	—

et les approximations correspondantes $x\left(\frac{1}{2}\right) \simeq 0.25$, $x(1) \simeq -0.0312$, $x(1.5) \simeq -0.1133$, $x(2) \simeq -0.1177$.

Exercices

1. Déterminer toutes les méthodes de Runge-Kutta explicite à 3 étapes.
2. Considérons la méthode de Runge-Kutta définie par son tableau de Butcher suivant :

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

   — Écrire le schéma de cette méthode.
   — Montrer que cette méthode est consistante. En déduire qu'elle est convergente.
   TP : Implémenter cette méthode dans un script Matlab `rk3(fun,tspan,y0,n)`. Faire un test au point $t = 1$ dans le cas du problème de Cauchy suivant

$$\begin{cases} y^{'}(t) = \sin(t) + y(t), & t \in [0,1] \\ y(0) = 0 \end{cases}$$

   dont la solution exacte est $y(t) = \frac{1}{2}\left(\exp(t) - \sin(t) - \cos(t)\right)$ et en déduire que la méthode est bien d'ordre 3.
3. On considère le schéma suivant dit de "*Crank-Nicolson*"

$$\begin{cases} u_0 = y_0, \\ u_{i+1} = u_i + \frac{h}{2}\left(f_i + f_{i+1}\right), & 0 \leq i \leq n-1. \end{cases}$$

   — Ce schéma est-il implicite ou explicite ? Que peut-on dire sur sa stabilité ?
   — Écrire l'erreur de consistance de ce schéma.
   — Notons par $t_{i+1/2}$ le milieu de l'intervalle (t_i, t_{i+1}). En écrivant les développement limités de y_i et y_{i+1} à l'ordre 3 par rapport à $y(t_{i+1/2})$ puis celui de $y^{''}(t_{i+1})$ à l'ordre 1 par rapport à $y^{''}(t_i)$ montrer que la schéma est consistant d'ordre 2. En déduire que le schéma converge à l'ordre 2.
   — Montrer que le schéma de *Crank-Nicolson* peut être vu comme une méthode de Runge-Kutta à 2 étapes. Donner dans ce cas son tableau de Butcher.
4. *Un modèle Prédateur-Proie*
   Considérons le système différentiel suivant qui modélise l'interaction

entre un prédateur y et sa proie x :

$$\begin{cases} \dfrac{dx}{dt}(t) = x(t)(1-y(t)), & 0 < t \leq 50 \\ \dfrac{dy}{dt}(t) = -y(t)(1-x(t)), & 0 < t \leq 50 \\ x(0) = 2, \\ y(0) = 2 \end{cases}$$

— Donner une discrétisation de ce système à l'aide de la méthode de Runge-Kutta à 2 étapes (2.18).
— Faire une simulation numérique de ce problème en prenant $n = 10, 15, 20$ et montrer que le système admet une solution périodique.

5. Considérons une tasse de café à la température de 75°C dans une salle à 25°C. On suppose que la température du café suit la lois de Newton c'est à dire que la vitesse de refroidissement du café est proportionnelle à la différence des températures selon la lois

$$T'(t) = -\frac{\ln 2}{5}\left(T(t) - T_{ext}\right),$$

où T_{ext} est la température ambiante.
— Écrire une discrétisation de ce problème en utilisant la méthode d'Adams-Moulton à 2 pas.
— Tracer avec Matlab la température en fonction du temps. Après combien de temps la température du café atteint 30°C ?

6. *(Un schéma d'ordre élevé)*
Soit f une fonction $C^\infty(\mathbb{R}_+ \times \mathbb{R}, \mathbb{R})$. Considérons l'équation différentielle suivante :

$$\begin{cases} x'(t) = f(t, x(t)), & t > 0 \\ x(0) = x_0. \end{cases}$$

Définissons $f_m \in C^\infty(\mathbb{R}_+ \times \mathbb{R}, \mathbb{R})$ par

$$\begin{aligned} &f_0(t,x) = f(t,x), \\ &f_{m+1}(t,x) = \frac{\partial f_m}{\partial t}(t,x) + \left(\frac{\partial f_m}{\partial x}(t,x)\right) f(t,x), \quad m \geq 0 \end{aligned}$$

— Montrer par récurrence que $x^{(m+1)}(t) = f_m(t, x(t))$, où $x^{(m)}$ désigne la dérivée d'ordre m de x.

Posons $\Psi_p(t,u,h) = \sum_{j=0}^{p-1} \frac{h^j}{(j+1)!} f_j(t,u)$ et définissons la schéma

$$\begin{cases} u_0 = x_0, \\ u_{i+1} = u_i + h\Psi_p(t_i, u_i, h). \end{cases} \tag{2.88}$$

— Montrer que le schéma (2.88) est consistant d'ordre p.
— Montrer que ce schéma est stable et en déduire qu'il est convergent d'ordre p.
TP : On considère le problème de Cauchy suivant

$$\begin{cases} x'(t) = e^{x(t)}, \quad t \in [0, \frac{1}{2}] \\ x(0) = 0, \end{cases}$$

— Faire une simulation numérique de ce problème en $t = \frac{1}{2}$ à l'aide du schéma (2) pour $p = 2, 4, 7, 10$. Vérifier numériquement l'ordre de la méthode dans chaque cas.

7. *(L'exemple du pendule sphérique du à [10])*
Le mouvement d'un point $X(t) = (x_1(t), x_2(t), x_3(t))$ de masse m soumis à la gravité $F = (0, 0, -gm)$ (avec $g = 9.81m/s^2$) et contraint de se déplacer sur la sphère d'équation $\Phi(X) = x_1^2 + x_2^2 + x_3^2 - 1 = 0$ est décrit par l'équation différentielle suivante

$$\begin{cases} X'' = \frac{1}{m}\left(F - \frac{mX'^T H X' + \nabla\Phi^T F}{|\nabla\Phi|^2}\nabla\Phi\right), \quad t > 0, \\ X(0) = X_0, \\ X'(0) = V_0. \end{cases} \tag{2.89}$$

On note X' la dérivée première et X'' la dérivée seconde par rapport à t, $\nabla\Phi$ la gradient spatial de Φ et H la matrice Hessienne de Φ dont les composantes sont $H_{ij} = \frac{\partial^2\Phi}{\partial x_i \partial x_j}$, $i, j = 1, 2, 3$. X_0 et V_0 dénote la position et la vitesse initiale respectivement.
— Écrire le problème (2.89) sous la forme d'un système du premier ordre.
— Faire une simulation numérique à l'aide de la fonction `ode23` de Matlab en prenant $X_0 = (0, 1, 0)$ et $V_0 = (0.8, 0, 1.2)$; on prendra $m = 1$ et $T = 25$.
— Tracer les solutions dans le plan des phases $Ox_1x_2x_3$.

8. Considérons le problème de Cauchy suivant :

$$\begin{cases} y'(t) = y(t) + 1, \quad 0 < t \leq T \\ y(0) = 1. \end{cases} \tag{2.90}$$

— Écrire le schéma d'Euler implicite avec un pas constant $h = \frac{T}{n}$, $(n \geq 1)$ du problème (2.90).
— En déduire l'expression de u_i en fonction de h et i. Calculer $\lim_{n\to\infty} u_n$ et en déduire la valeur exacte de $y(T)$.
— Vérifier que l'expression obtenue est bien la solution exacte du problème (2.90).

9. Considérons l'équation différentielle

$$\begin{cases} x''(t) - x(t) = 0, & 0 \leq t \leq 1 \\ x(0) = 0, \\ x'(0) = 2. \end{cases} \tag{2.91}$$

— Écrire le problème (2.91) sous la forme d'un système différentiel.
— Calculer la solution exacte x.
— Calculer une solution approchée en $t = 1$ du système (2.91) à l'aide de la méthode de Runge-Kutta explicite à 2 étapes en prenant $n = 4$. Quelle est l'erreur commise ?

10. Calculer la fonction d'incrément Φ de la méthode de Runge-Kutta à 2 étapes (2.18). En déduire que la méthode de Runge-Kutta est une méthode à un pas non linéaire.

11. Montrer que si f et localement lipschitzienne par rapport à sa seconde variable et que si la fonction d'incrément Φ est lipschitzienne en f_i et f_{i+1} alors la méthode (2.58) admet à chaque itération une solution unique pour h assez petit.

12. Trouver l'ordre de convergence de la méthode BDF2 (2.65).

13. Proposer un schéma d'Adams-Moulton à 3 pas. Quelle est l'ordre de convergence de cette méthode ?
Vérifier à l'aide de l'exemple suivant

$$\begin{cases} y'(t) = \sin(t) + y^3(t), & t \in [0, 1] \\ y(0) = 0 \end{cases}$$

l'ordre de convergence de la méthode au point $t = 1$.

14. Pour quelles valeurs de α la méthode multi-pas suivante

$$u_{i+1} = \alpha u_i + (1 - \alpha)u_{i-1} + 2hf_i + \frac{h\alpha}{2}(f_{i-1} - 3f_i)$$

est-elle convergente ?

15. En utilisant la méthode d'Euler implicite donner une approximation du problème de Cauchy suivant

$$\begin{cases} y' = -te^{-y}, & t \in (0,1], \\ y(0) = 0, \end{cases} \tag{2.92}$$

aux nœuds $\{0, \frac{1}{3}, \frac{2}{3}, 1\}$.
Indication : On utilisera la méthode de Newton pour calculer u_i à chaque itération avec 2 chiffres après la virgule.
Méthode de Newton :

$$\begin{cases} u_{i+1} = u_i - \frac{f(u_i)}{f'(u_i)}, & i \geq 0 \\ u_0. \end{cases}$$

Le choix de u_0 est soumis à la condition $f(u_0)f'(u_0) > 0$ et le test d'arrêt est $|u_{i+1} - u_i| < \epsilon$ où ϵ est la tolérance égale à 0.5×10^{-2} dans notre cas.

16. En approchant $y'(t_i)$ par la formule de différence finie centrée (A.13) proposer un schéma pour l'approximation du problème (2.1). Étudier la convergence de ce schéma. Quel son ordre de convergence ?