Chapitre III

Equations Différentielles Ordinaires

Ce chapitre est consacré à la résolution numérique d'un système d'équations différentielles ordinaires

$$\begin{aligned} y_1' &= f_1(t, y_1, \ldots, y_n), \qquad y_1(t_0) = y_{10}, \\ &\vdots \\ y_n' &= f_n(t, y_1, \ldots, y_n), \qquad y_n(t_0) = y_{n0}. \end{aligned} \tag{0.1}$$

En notation vectorielle, ce système s'écrit

$$y' = f(t, y), \qquad y(t_0) = y_0 \tag{0.2}$$

où $y = (y_1, \ldots, y_n)^T$ et $f : I\!R \times I\!R^n \to I\!R^n$. Voici quelques livres qui traitent de ce sujet.

Bibliographie sur ce chapitre

K. Burrage (1995): *Parallel and sequential methods for ordinary differential equations.* The Clarendon Press, Oxford University Press. [MA 65/369]

J.C. Butcher (1987): *The Numerical Analysis of Ordinary Differential Equations.* John Wiley & Sons. [MA 65/276]

J.C. Butcher (2003): *Numerical Methods for Ordinary Differential Equations.* John Wiley & Sons. [MA 65/470]

M. Crouzeix & A.L. Mignot (1984): *Analyse Numérique des Equations Différentielles.* Masson. [MA 65/217]

P. Deuflhard & F. Bornemann (1994): *Numerische Mathematik II. Integration gewöhnlicher Differentialgleichungen.* Walter de Gruyter. [MA 65/309]

E. Hairer, S.P. Nørsett & G. Wanner (1993): *Solving Ordinary Differential Equations I. Nonstiff Problems.* Springer Series in Comput. Math., vol. 8, 2nd edition. [MA 65/245]

E. Hairer & G. Wanner (1996): *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems.* Springer Series in Comput. Math., vol. 14, 2nd edition. [MA 65/245]

E. Hairer, C. Lubich & G. Wanner (2002): *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations.* Springer Series in Comput. Math., vol. 31, 2nd edition in preparation. [MA 65/448]

P. Henrici (1962): *Discrete Variable Methods in Ordinary Differential Equations.* John Wiley & Sons. [MA 65/50]

A. Iserles (1996): *A First Course in the Numerical Analysis of Differential Equations.* Cambridge Texts in Applied Mathematics, Cambridge University Press.

J.D. Lambert (1991): *Numerical Methods for Ordinary Differential Equations.* John Wiley & Sons. [MA 65/367]

A.M. Stuart & A.R. Humphries (1996): *Dynamical Systems and Numerical Analysis.* Cambridge Univ. Press. [MA 65/377]

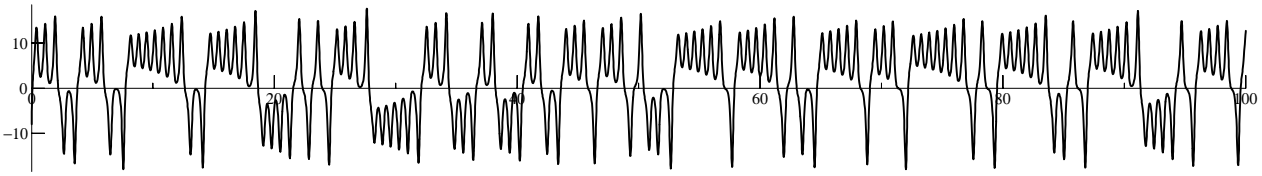
III.1 Quelques exemples typiques

Pour des équations différentielles d'un intérêt pratique on trouve rarement la solution $y(t)$ exprimée avec une formule exacte. On est alors obligé d'utiliser des méthodes numériques.

Exemple 1.1 (modèle de Lorenz) Une équation très celèbre est celle de Lorenz (1979)

$$\begin{aligned} y_1' &= -\sigma y_1 + \sigma y_2 & y_1(0) &= -8 \\ y_2' &= -y_1 y_3 + r y_1 - y_2 & y_2(0) &= 8 \\ y_3' &= y_1 y_2 - b y_3 & y_3(0) &= r - 1 \end{aligned} \tag{1.1}$$

avec $\sigma = 10$, $r = 28$ et $b = 8/3$. La solution est chaotique et ne devient jamais périodique. Voici la composante $y_1(t)$ comme fonction de t sur l'intervalle $[0, 100]$.



Les méthodes classiques comme les *méthodes de Runge–Kutta* (voir le paragraphe III.2) ou les *méthodes multipas* (paragraphe III.5) nous permettent de trouver sans difficultés des bonnes approximations.

La solution $y(t) = (y_1(t), y_2(t), y_3(t))^T$ peut aussi être interprétée comme une courbe paramétrique dans l'espace $\mathbb{R}^3$ (avec paramètre t). Leurs projections sur le plan des composantes (y_1, y_2) et (y_1, y_3) sont dessinées dans la figure III.1. L'intervalle d'intégration est $[0, 25]$. La valeur initiale est marquée par un point noir épais.

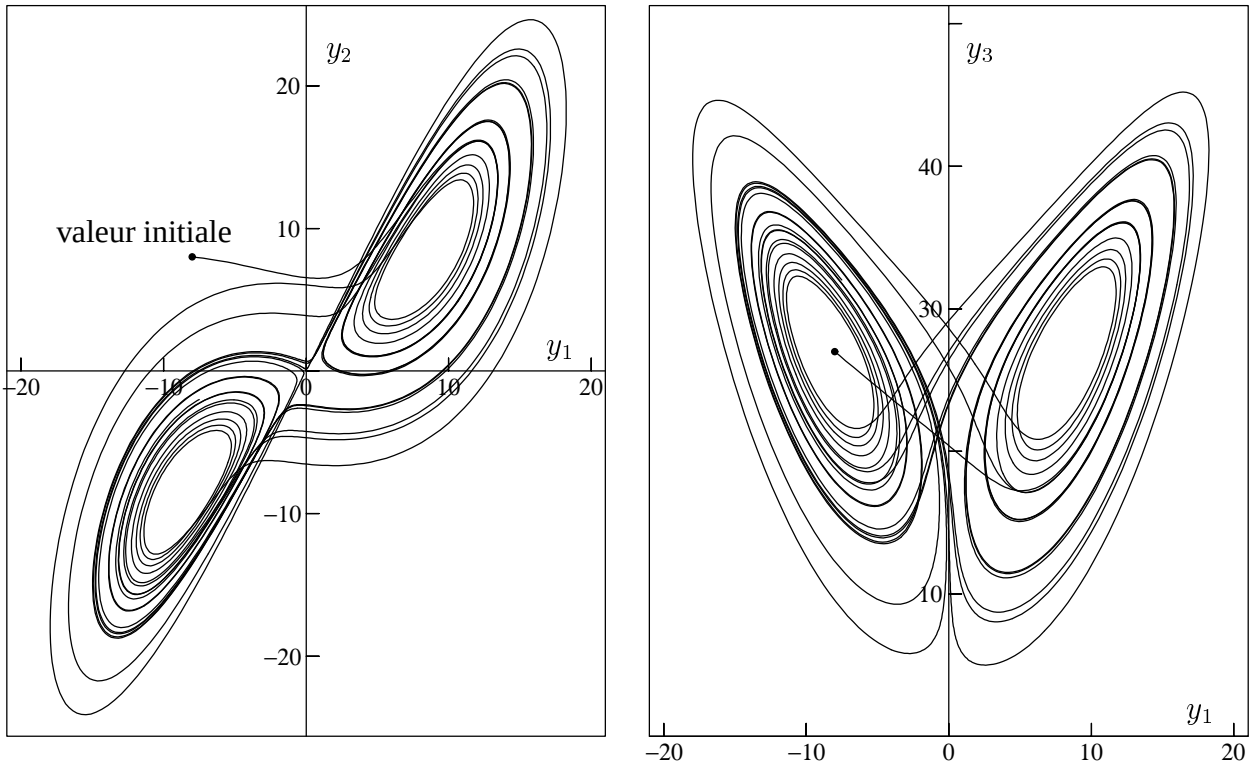


FIG. III.1: Deux vues de la solution $y(t) = (y_1(t), y_2(t), y_3(t))^T$ du problème de Lorenz (1.1)

> When the equations represent the behaviour of a system containing a number of fast and slow reactions, a forward integration of these equations becomes difficult. (H.H. Robertson 1966)

Exemple 1.2 (réactions chimiques) L'exemple suivant de Robertson (1966) est devenu célèbre comme équation test pour des études numériques (Willoughby 1974): la réaction chimique

$$\begin{array}{lcll} A & \xrightarrow{0.04} & B & \text{(lente)} \\ B+B & \xrightarrow{3\cdot 10^7} & C+B & \text{(très rapide)} \\ B+C & \xrightarrow{10^4} & A+C & \text{(rapide)} \end{array}$$

conduit au système d'*équations différentielles raides*

$$\begin{array}{lllll} A: & y_1' = -0.04y_1 + 10^4 y_2 y_3 & & & y_1(0)=1 \\ B: & y_2' = 0.04y_1 - 10^4 y_2 y_3 & -3\cdot 10^7 y_2^2 & & y_2(0)=0 \\ C: & y_3' = & 3\cdot 10^7 y_2^2 & & y_3(0)=0. \end{array} \tag{1.2}$$

Si on utilise une méthode classique (par exemple, la méthode de Runge–Kutta DOPRI5) on est obligé de prendre des pas d'intégration très petits pour obtenir une approximation correcte (voir la figure III.2). Dans le paragraphe III.9 nous étudierons des méthodes numériques adaptées à ce type de problèmes (dont un représentant est RADAU5). Elles donnent une grande précision avec des grands pas d'intégration.

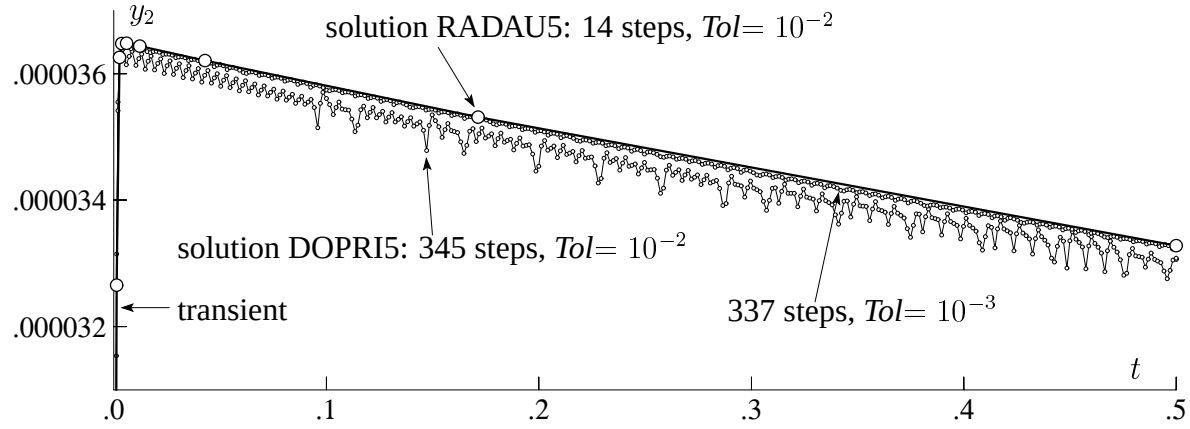


FIG. III.2: Solution numérique pour (1.2) obtenue par une méthode classique (DOPRI5) et par une méthode pour des équations différentielles raides (RADAU5)

Exemple 1.3 (intégration à long terme du système planétaire) Comme dernier exemple, considérons le problème à N corps qui est important aussi bien dans l'astronomie (mouvement des planètes) que dans la biologie moléculaire (mouvement des atomes). Les équations sont

$$y_i'' = G\sum_{j\neq i} m_j \frac{y_j - y_i}{\|y_j - y_i\|^3} \tag{1.3}$$

où $y_i \in I\!R^3$ est la position du $i$ème corps, m_i sa masse, et G la constante gravitationnelle. En introduisant la vitesse $v_i = y_i'$ comme nouvelle variable, on obtient un système d'équations différentielles pour les y_i et v_i de dimension $6N$ où N est le nombre des corps considérés.

Comme exemple concret, considérons le mouvement de cinq planètes extérieures autour du soleil ($N = 6$ corps).[1] La figure III.3 montre la différence entre une méthode classique (méthode d'Euler explicite) et une méthode adaptée à ce problème; voir le paragraphe III.10 pour une explication.

[1]La constante G, les masses m_i est les valeurs initiales pour les positions et vitesses sont données dans le paragraphe I.2.3 du livre *Geometric Numerical Integration* de Hairer, Lubich & Wanner (2002).

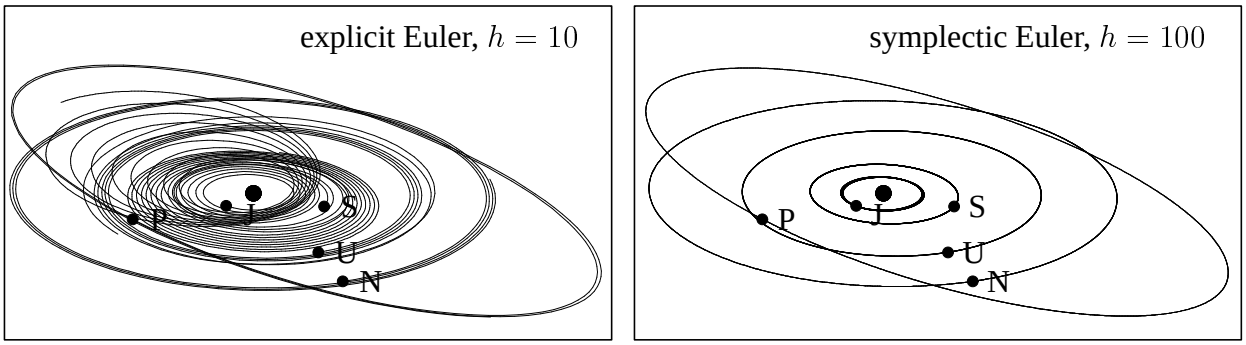


FIG. III.3: Solution numérique pour (1.3) obtenue par une méthode classique (méthode d'Euler explicite) et par une méthode étudiée au paragraphe III.10

Avant de discuter la résolution numérique des équations différentielles, nous rappelons un théorème sur l'existence et l'unicité de la solution (pour une démonstration, voir le cours d'Analyse II).

Théorème 1.4 *Soit $f(t, y)$ une fonction continûment différentiable dans un voisinage de (t_0, y_0). Alors, il existe $\alpha > 0$ tel que le problème $y' = f(t, y), y(t_0) = y_0$ possède exactement une solution sur l'intervalle $(t_0 - \alpha, t_0 + \alpha)$.* □

III.2 Méthodes de Runge-Kutta

Pour calculer une approximation de la solution de

$$y' = f(t, y), \qquad y(t_0) = y_0 \tag{2.1}$$

sur l'intervalle $[t_0, T]$, on procède comme suit: on subdivise $[t_0, T]$ en sous-intervalles d'extrémités $t_0 < t_1 < \ldots < t_N = T$, on dénote $h_n = t_{n+1} - t_n$ et on calcule l'approximation $y_n \approx y(t_n)$ par une formule de type

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

Une telle formule s'appelle "méthode à *un* pas", car le calcul de y_{n+1} utilise uniquement les valeurs h_n, t_n, y_n et non $h_{n-1}, t_{n-1}, y_{n-1}, \ldots$.

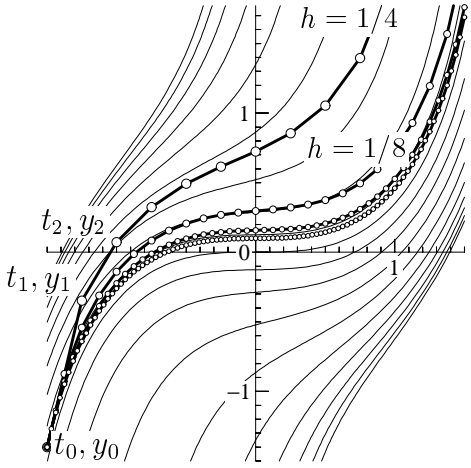
Méthode d'Euler (1768). La méthode la plus simple est

$$y_1 = y_0 + h f(t_0, y_0)$$

(pour simplifier la notation nous considérons uniquement le premier pas ($n = 0$ dans (2.2)) et nous notons $h_0 = h$). Elle est obtenue en remplaçant la solution $y(t)$ par sa tangente au point (t_0, y_0). Le dessin à droite montre la solution numérique pour le problème

$$y' = t^2 + y^2, \quad y(-1.5) = -1.4$$

et pour $h = 1/4$, $h = 1/8$, etc. On peut observer la convergence vers une fonction qui, comme on verra dans le paragraphe III.3, est la solution du problème.



Pour la dérivation d'autres méthodes numériques, intégrons (2.1) de t_0 à $t_0 + h$

$$y(t_0 + h) = y_0 + \int_{t_0}^{t_0+h} f(\tau, y(\tau))\, d\tau. \tag{2.3}$$

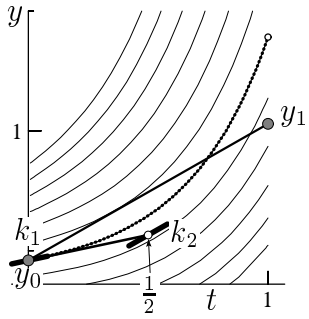
Si l'on remplace l'intégrale de (2.3) par $hf(t_0, y_0)$, on obtient la méthode d'Euler. L'idée évidente est d'approcher cette intégrale par une formule de quadrature ayant un ordre plus élevé.

Méthode de Runge (1895). On prend la formule du point milieu

$$y(t_0 + h) \approx y_0 + hf\Big(t_0 + \frac{h}{2}, y(t_0 + \frac{h}{2})\Big)$$

et on remplace la valeur inconnue $y(t_0 + h/2)$ par la méthode d'Euler. Ceci nous donne

$$y_1 = y_0 + hf\Big(t_0 + \frac{h}{2}, y_0 + \frac{h}{2} f(t_0, y_0)\Big).$$



Méthode de Heun (1900). On prend une formule de quadrature d'ordre 3

$$y(t_0 + h) \approx y_0 + \frac{h}{4}\Big(f(t_0, y_0) + 3f(t_0 + \frac{2h}{3}, y(t_0 + \frac{2h}{3}))\Big) \tag{2.4}$$

et on remplace la valeur inconnue $y(t_0 + 2h/3)$ par l'approximation de la méthode de Runge. Ceci nous donne

$$y_1 = y_0 + \frac{h}{4}\Big(f(t_0, y_0) + 3f\Big(t_0 + \frac{2h}{3}, y_0 + \frac{2h}{3} f\Big(t_0 + \frac{h}{3}, y_0 + \frac{h}{3} f(t_0, y_0)\Big)\Big)\Big). \tag{2.5}$$

En généralisant cette idée à une formule de quadrature plus générale et en introduisant la notation $k_i = f(\ldots)$ pour les expressions de $f(t, y)$ qui apparaissent, on est conduit à la définition suivante (Kutta 1901).

Définition 2.1 *Une méthode de Runge-Kutta à s étages est donnée par*

$$\begin{aligned} k_1 &= f(t_0, y_0) \\ k_2 &= f(t_0 + c_2 h, y_0 + h a_{21} k_1) \\ k_3 &= f(t_0 + c_3 h, y_0 + h(a_{31} k_1 + a_{32} k_2)) \\ &\ldots \\ k_s &= f(t_0 + c_s h, y_0 + h(a_{s1} k_1 + \ldots + a_{s,s-1} k_{s-1})) \\ y_1 &= y_0 + h(b_1 k_1 + \ldots + b_s k_s) \end{aligned} \tag{2.6}$$

où c_i, a_{ij}, b_j sont des coefficients. On la représente à l'aide du schéma

c_i	a_{ij}
	b_i

Exemples. Les méthoded d'Euler, de Runge et de Heun sont données par les tableaux suivants:

0	
	1

0		
$1/2$	$1/2$	
	0	1

0			
$1/3$	$1/3$		
$2/3$	0	$2/3$	
	$1/4$	0	$3/4$

Par la suite nous supposerons toujours que les c_i satisfont

$$c_1 = 0, \qquad c_i = \sum_{j=1}^{i-1} a_{ij}, \quad i = 2, \ldots, s. \tag{2.7}$$

Ceci signifie que $k_i = f(t_0 + c_i h, y(t_0 + c_i h)) + \mathcal{O}(h^2)$. Nous étendons maintenant la notion de l'ordre (voir la Définition I.1.2 pour les formules de quadrature) aux méthodes de Runge-Kutta.

Définition 2.2 *On dit que la méthode (2.6) a l'ordre p si, pour chaque problème $y' = f(t,y)$, $y(t_0) = y_0$ (avec $f(t,y)$ suffisamment différentiable), l'erreur après un pas satisfait*

$$y_1 - y(t_0+h) = \mathcal{O}(h^{p+1}) \qquad \textit{pour} \qquad h \to 0. \tag{2.8}$$

La différence (2.8) s'appelle erreur locale de la méthode.

La méthode d'Euler est une méthode d'ordre $p = 1$, car

$$y(t_0+h) = y_0 + hy'(t_0) + \mathcal{O}(h^2) = y_0 + hf(t_0,y_0) + \mathcal{O}(h^2) = y_1 + \mathcal{O}(h^2).$$

La méthode de Runge est basée sur la formule du point milieu qui est une formule de quadrature d'ordre 2:

$$y(t_0+h) = y_0 + hf\Big(t_0 + \frac{h}{2}, y(t_0+\frac{h}{2})\Big) + \mathcal{O}(h^3).$$

En remplaçant $y(t_0+h/2)$ par la valeur $y_0 + (h/2)f(t_0,y_0)$ de la méthode d'Euler, on ajoute un terme d'erreur qui est de $\mathcal{O}(h^3)$ grâce au facteur h devant f. Ainsi, cette méthode a l'ordre $p = 2$. De la même manière on voit que la méthode de Heun a l'ordre $p = 3$.

Pour construire des méthodes d'ordre plus élevé, il faut développer la solution exacte $y(t_0+h)$ et la solution numérique $y_1 = y_1(h)$ en série de Taylor autour de $h = 0$. Une comparaison des coefficients de h^i pour $i = 1,\dots,p$ donne des conditions pour les paramètres a_{ij} et b_i. L'idée est simple, mais l'exécution de ce plan est loin d'être triviale (8 conditions pour l'ordre $p = 4$, 200 pour $p = 8$ et 1205 pour $p = 10$). Mais c'est de cette manière que Kutta (1901) a trouvé des méthodes d'ordre 4. Les plus célèbres sont données dans le tableau III.1. Celle de gauche est basée sur la formule de Simpson, l'autre sur la formule de quadrature de Newton.

TAB. III.1: Méthodes de Kutta (1901)

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

"La" méthode de Runge-Kutta

0				
$1/3$	$1/3$			
$2/3$	$-1/3$	1		
1	1	-1	1	
	$1/8$	$3/8$	$3/8$	$1/8$

règle 3/8

Les méthodes de Runge-Kutta, qui sont actuellement les plus utilisées, ont été construites autour de 1980 par Dormand et Prince (Angleterre). Pour des méthodes d'ordre $p = 5$ avec $s = 6$ et d'ordre $p = 8$ avec $s = 12$, des programmes informatiques DOPRI5 et DOP853 sont disponibles sur la page internet <http://www.unige.ch/∼hairer>.

Expérience numérique. Considérons les cinq méthodes vues jusqu'à maintenant (Euler, Runge, Heun et les deux méthodes du tableau III.1 de Kutta) et comparons leurs performances pour le problème (équation Van der Pol)

$$\begin{aligned} y_1' &= y_2 & y_1(0) &= 2.00861986087484313650940188 \\ y_2' &= (1-y_1^2)y_2 - y_1 & y_2(0) &= 0. \end{aligned} \tag{2.9}$$

La valeur initiale est choisie pour que la solution soit périodique de période

$$T = 6.6632868593231301896996820305.$$

Nous subdivisons l'intervalle $[0,T]$ en n parties équidistantes et appliquons n fois la méthode. L'erreur à la fin de l'intervalle est alors dessinée en fonction du travail (nombre total d'évaluations de f) dans la figure III.4.

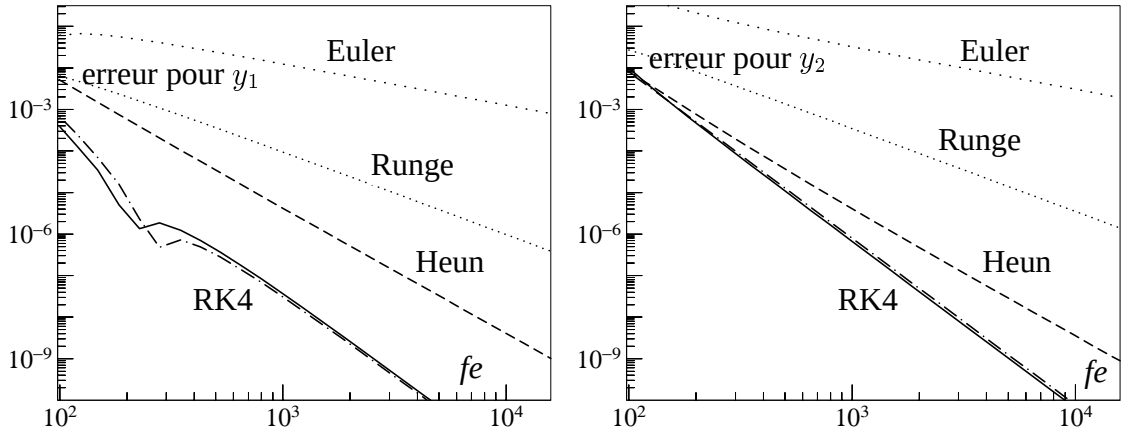


FIG. III.4: Erreur globale en fonction du travail numérique

Comme dans la fig. I.4 (intégration numérique), on peut constater que $\log_{10}(\mathit{err})$ dépend linéairement de $\log_{10}(\mathit{fe})$ et que cette droite est de pente $-p$, où p est l'ordre de la méthode. Il est donc important d'utiliser des méthodes d'ordre élevé.

III.3 Convergence des méthodes de Runge-Kutta

Dans la figure III.4, on a constaté que pour un calcul avec des pas constants, l'erreur globale se comporte comme $\log_{10}(\mathit{err}) \approx C_0 - p \cdot \log_{10}(\mathit{fe})$, ce qui est équivalent à $\mathit{err} \approx C_1(\mathit{fe})^{-p} \approx C_2 h^p$. Ceci montre que la solution numérique converge vers la solution exacte si $h \to 0$. Dans ce paragraphe, nous allons démontrer ce résultat.

Nous appliquons une méthode à un pas

$$y_{n+1} = y_n + h_n \Phi(t_n, y_n, h_n) \tag{3.1}$$

à une équation différentielle $y' = f(t,y)$, $y(t_0) = y_0$, et nous cherchons à estimer l'*erreur globale* $y(t_n) - y_n$.

Théorème 3.1 *Soit $y(t)$ la solution de $y' = f(t,y)$, $y(t_0) = y_0$ sur $[t_0, T]$. Supposons que*

a) l'erreur locale satisfasse pour $t \in [t_0, T]$ et $h \le h_{\max}$

$$\|y(t+h) - y(t) - h\Phi(t, y(t), h)\| \le C \cdot h^{p+1} \tag{3.2}$$

b) la fonction $\Phi(t,y,h)$ satisfasse une condition de Lipschitz

$$\|\Phi(t,y,h) - \Phi(t,z,h)\| \le \Lambda \cdot \|y - z\| \tag{3.3}$$

pour $h \le h_{\max}$ et $(t,y), (t,z)$ dans un voisinage de la solution.

Alors, l'erreur globale admet pour $t_n \le T$ l'estimation

$$\|y(t_n) - y_n\| \le h^p \cdot \frac{C}{\Lambda} \cdot \left(e^{\Lambda(t_n - t_0)} - 1 \right) \tag{3.4}$$

où $h = \max_i h_i$, sous la condition que h soit suffisamment petit.

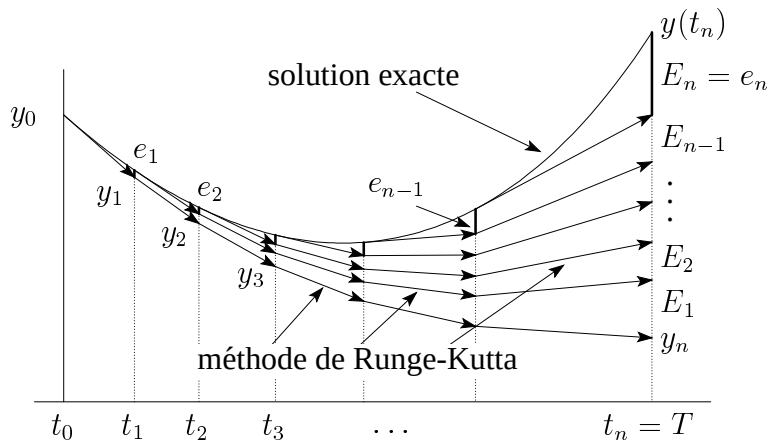


FIG. III.5: Estimation de l'erreur globale

Démonstration. L'idée est d'étudier l'influence de l'erreur locale, commise au $i^{\text{ème}}$ pas, sur l'approximation y_n. Ensuite, on va additionner les erreurs accumulées.

Propagation de l'erreur. Soient $\{y_n\}$ et $\{z_n\}$ deux solutions numériques obtenues par (3.1) avec pour valeurs initiales y_0 et z_0, respectivement. En utilisant la condition de Lipschitz (3.3), leur différence peut être estimée comme

$$\|y_{n+1} - z_{n+1}\| \leq \|y_n - z_n\| + h_n\Lambda\|y_n - z_n\| = (1 + h_n\Lambda)\|y_n - z_n\| \leq e^{h_n\Lambda}\|y_n - z_n\|. \quad (3.5)$$

Récursivement, on obtient alors

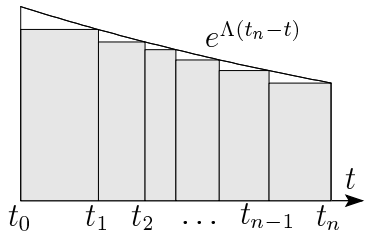
$$\|y_n - z_n\| \leq e^{h_{n-1}\Lambda} \cdot e^{h_{n-2}\Lambda} \cdot \ldots \cdot e^{h_i\Lambda}\|y_i - z_i\| = e^{\Lambda(t_n - t_i)}\|y_i - z_i\|.$$

et l'erreur propagée E_i (voir la figure III.5) satisfait

$$\|E_i\| \leq e^{\Lambda(t_n - t_i)}\|e_i\| \leq Ch_{i-1}^{p+1}e^{\Lambda(t_n - t_i)}. \quad (3.6)$$

Accumulation des erreurs propagées. L'inégalité du triangle et l'estimation (3.6) nous donnent

$$\begin{aligned}
\|y(t_n) - y_n\| &\leq \sum_{i=1}^{n} \|E_i\| \leq C\sum_{i=1}^{n} h_{i-1}^{p+1}e^{\Lambda(t_n - t_i)} \\
&\leq Ch^p\Big(h_0e^{\Lambda(t_n - t_1)} + \ldots + h_{n-2}e^{\Lambda(t_n - x_{t-1})} + h_{n-1}\Big) \\
&\leq Ch^p\int_{t_0}^{t_n} e^{\Lambda(t_n - \tau)}\,d\tau = Ch^p\frac{1}{-\Lambda}e^{\Lambda(t_n - \tau)}\Big|_{t_0}^{t_n} \\
&= \frac{Ch^p}{\Lambda}\Big(e^{\Lambda(t_n - t_0)} - 1\Big).
\end{aligned}$$



Cette estimation montre que, pour h suffisamment petit, la solution numérique ne sort pas du voisinage où Φ satisfait à une condition de Lipschitz. Ceci justifie les estimations dans (3.5). □

Supposons maintenant que (3.1) représente une méthode de Runge-Kutta et vérifions les hypothèses du théorème précédent. La condition (3.2) est satisfaite pour une méthode d'ordre p (par définition de l'ordre). Il reste à vérifier la condition de Lipschitz (3.3) pour la fonction

$$\Phi(t, y, h) = \sum_{i=1}^{s} b_i k_i(y) \quad (3.7)$$

où $k_i(y) = f(t + c_ih, y + h\sum_{j=1}^{i-1} a_{ij}k_j(y))$.

Lemme 3.2 *Si $f(t,y)$ satisfait une condition de Lipschitz $\|f(t,y)-f(t,z)\| \le L\|y-z\|$ dans un voisinage de la solution de $y'=f(t,y)$, l'expression $\Phi(t,y,h)$ de (3.7) vérifie (3.3) avec*

$$\Lambda = L\Big(\sum_i |b_i| + (h_{\max}L)\sum_{i,j}|b_i a_{ij}| + (h_{\max}L)^2\sum_{i,j,k}|b_i a_{ij}a_{jk}| + \ldots\Big). \tag{3.8}$$

Démonstration. La condition de Lipschitz pour $f(t,y)$ appliquée à $k_i(y)-k_i(z)$ nous donne

$$\|k_1(y)-k_1(z)\| = \|f(t,y)-f(t,z)\| \le L\|y-z\|$$
$$\|k_2(y)-k_2(z)\| \le L\|y-z+ha_{21}(k_1(y)-k_1(z))\| \le L(1+h_{\max}L|a_{21}|)\|y-z\|$$

etc. Ces estimations insérées dans

$$\|\Phi(t,y,h)-\Phi(t,z,h)\| \le \sum_{i=1}^{s}|b_i|\cdot\|k_i(y)-k_i(z)\|$$

impliquent (3.3) avec Λ donné par (3.8). □

III.4 Un programme à pas variables

Pour résoudre un problème réaliste, un calcul à pas constants est en général inefficace. Mais comment choisir la division? L'idée est de choisir les pas afin que l'erreur locale soit partout environ égale à *Tol* (fourni par l'utilisateur). A cette fin, il faut connaître une estimation de l'erreur locale. Inspiré par le programme TEGRAL pour l'intégration numérique (voir le paragraphe I.6), nous construisons une deuxième méthode de Runge-Kutta avec $\widehat{y}_1$ comme approximation numérique, et nous utilisons la différence $\widehat{y}_1 - y_1$ comme estimation de l'erreur locale du moins bon résultat.

Méthode emboîtée. Soit donnée une méthode d'ordre p à s étages (coefficients c_i, a_{ij}, b_j). On cherche une approximation $\widehat{y}_1$ d'ordre $\widehat{p} < p$ qui utilise les mêmes évaluations de f, c.-à-d.,

$$\widehat{y}_1 = y_0 + h(\widehat{b}_1 k_1 + \ldots + \widehat{b}_s k_s)$$

où les k_i sont donnés par la méthode (2.6). Pour avoir plus de liberté, on ajoute souvent un terme avec $f(x_1,y_1)$, qu'il faut en tous cas calculer pour le pas suivant, et on cherche $\widehat{y}_1$ de la forme

$$\widehat{y}_1 = y_0 + h\Big(\widehat{b}_1 k_1 + \ldots + \widehat{b}_s k_s + \widehat{b}_{s+1} f(x_1,y_1)\Big). \tag{4.1}$$

Exemple. Pour la méthode de Runge, basée sur la règle du point milieu, on peut prendre la méthode d'Euler comme méthode emboîtée. L'expression $err = h(k_2-k_1)$ est donc une approximation de l'erreur locale (pour la méthode d'Euler).

Pour une méthode générale, il faut développer les k_i et $f(x_1,y_1)$ en série de Taylor et comparer avec la solution exacte. Comme les c_i et les a_{ij} sont déjà connus, on obtient un système linéaire pour le $\widehat{b}_i$.

En faisant ce calcul pour la méthode "règle $3/8$" (ordre $p=4$, tableau III.1), les coefficients d'une méthode emboîtée avec $\widehat{p}=3$ sont

$$\widehat{b}_1 = b_1 - \frac{c}{24} \qquad \widehat{b}_2 = b_2 + \frac{c}{8} \qquad \widehat{b}_3 = b_3 - \frac{c}{8} \qquad \widehat{b}_4 = b_4 - \frac{c}{8} \qquad \widehat{b}_5 = \frac{c}{6}, \tag{4.2}$$

et avec $c=1$ on obtient donc

$$err = \frac{h}{24}\Big(-k_1 + 3k_2 - 3k_3 - 3k_4 + 4f(x_1,y_1)\Big) \tag{4.3}$$

comme estimation de l'erreur locale.

Calcul du h "optimal". Si l'on applique la méthode avec une certaine valeur h, l'estimation de l'erreur satisfait ($\widehat{p} < p$)

$$y_1 - \widehat{y}_1 = (y_1 - y(t_0 + h)) + (y(t_0 + h) - \widehat{y}_1) = \mathcal{O}(h^{p+1}) + \mathcal{O}(h^{\widehat{p}+1}) \approx C \cdot h^{\widehat{p}+1}. \qquad (4.4)$$

Le h "optimal", noté par h_{opt}, est celui où cette estimation est proche de *Tol*:

$$\mathit{Tol} \approx C \cdot h_{\text{opt}}^{\widehat{p}+1}. \qquad (4.5)$$

En éliminant C de (4.4) et de (4.5), on obtient

$$h_{\text{opt}} = 0.9 \cdot h \cdot \sqrt[\widehat{p}+1]{\frac{\mathit{Tol}}{\|y_1 - \widehat{y}_1\|}}. \qquad (4.6)$$

Le facteur 0.9 est ajouté pour rendre le programme plus sûr. En général on supose en plus une restriction du type $0.2h \le h_{\text{opt}} \le 5h$ pour éviter des grandes variations dans h.

Pour la norme dans (4.6) on utilise en général

$$\|y_1 - \widehat{y}_1\| = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Big(\frac{y_{i1} - \widehat{y}_{i1}}{\mathit{sc}_i}\Big)^2} \qquad \text{où} \qquad \mathit{sc}_i = 1 + \max(|y_{i0}|, |y_{i1}|) \qquad (4.7)$$

($y_{i0}, y_{i1}, \widehat{y}_{i1}$ est la $i^{\text{ème}}$ composante de $y_0, y_1, \widehat{y}_1$, respectivement). Ceci représente un mélange entre erreur relative et erreur absolue.

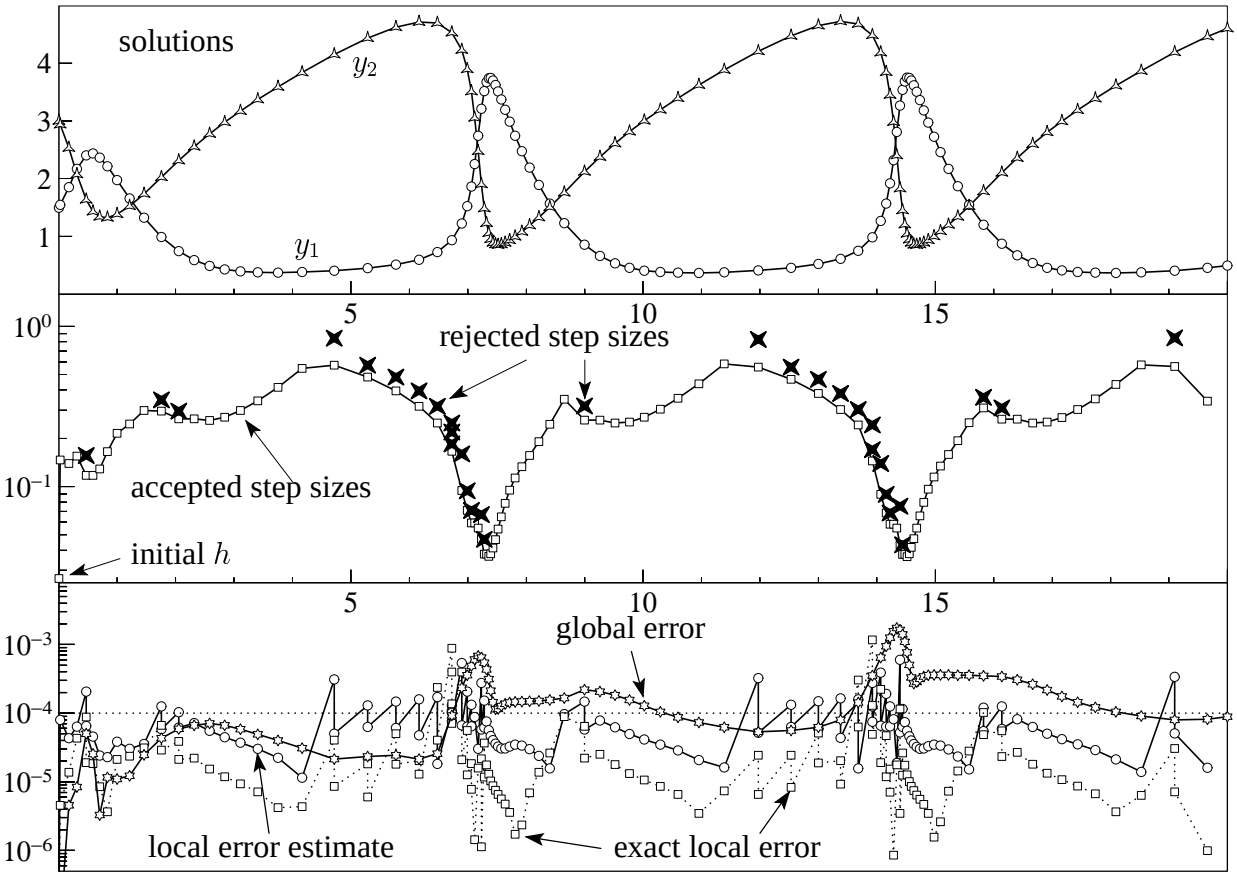


FIG. III.6: Sélection du pas, *Tol* $= 10^{-4}$, 96 pas acceptés + 32 pas rejetés

Algorithme pour la sélection automatique du pas. Au début, l'utilisateur fournit un sous-programme qui calcule la valeur de $f(t,y)$, les valeurs initiales t_0, y_0 et un premier choix de h.

A) Avec h, calculer y_1, *err* $= \|y_1 - \widehat{y}_1\|$ et h_{opt} de (4.6).

B) **If** *err* $\leq$ *Tol* (le pas est accepté) **then**
$t_0 := t_0 + h, \quad y_0 := y_1, \quad h := \min(h_{\text{opt}}, t_{\text{end}} - t_0)$
else (le pas est rejeté)
$h := h_{\text{opt}}$
end if

C) Si $t_0 = t_{\text{end}}$ on a terminé, sinon on recommence en (A) et on calcule le pas suivant.

Exemple numérique. Cet algorithme a été programmé (en utilisant la "règle $3/8$" et l'estimation de l'erreur (4.3)) et il a été appliqué au problème (une réaction chimique, le "Brusselator")

$$\begin{aligned} y_1' &= 1 + y_1^2 y_2 - 4y_1 \qquad & y_1(0) &= 1.5 \\ y_2' &= 3y_1 - y_1^2 y_2 & y_2(0) &= 3 \end{aligned} \tag{4.8}$$

sur l'intervalle $[0, 20]$. Les résultats obtenus avec *Tol* $= 10^{-4}$ sont présentés dans la figure III.6 :

i) en haut, les deux composantes de la solution avec tous les pas acceptés;

ii) au milieu les pas; les pas acceptés étant reliés, les pas rejetés étant indiqués par $\times$;

iii) les dessin du bas montre l'estimation de l'erreur locale *err*, ainsi que les valeurs exactes de l'erreur locale et de l'erreur globale.

III.5 Méthodes multipas (multistep methods)

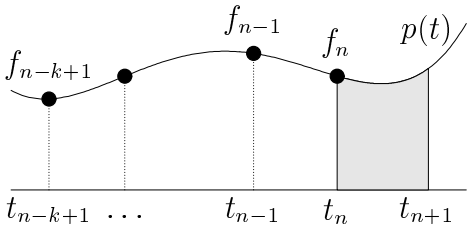
Déjà longtemps avant la parution des premières méthodes de Runge-Kutta, J.C. Adams a résolu numériquement des équations différentielles (1855, publié dans un livre de Bashforth 1883). Son idée était d'utiliser l'information de plusieurs pas précédents (en particulier $y_n, y_{n-1}, \ldots, y_{n-k+1}$) pour obtenir une approximation précise de $y(t_{n+1})$. C'est la raison pour laquelle ces méthodes s'appellent aujourd'hui méthodes multipas (contrairement aux méthodes à un pas).

Méthodes d'Adams explicites. Soit donnée une division $t_0 < \ldots < t_n < t_{n+1} < \ldots$ de l'intervalle sur lequel on cherche à résoudre l'équation différentielle $y' = f(t,y)$ et supposons qu'on connaisse des approximations $y_n, y_{n-1}, \ldots, y_{n-k+1}$ de la solution pour k pas consécutifs ($y_j \approx y(t_j)$). Comme pour la dérivation des méthodes de Runge-Kutta, on intègre l'équation différentielle pour obtenir

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(\tau, y(\tau))\, d\tau. \tag{5.1}$$

Mais, au lieu d'appliquer une formule de quadrature standard à l'intégrale de (5.1), on remplace $f(t,y(t))$ par le polynôme $p(t)$ de degré $k-1$ qui satisfait (on utilise l'abréviation $f_j = f(t_j, y_j)$)

$$p(t_j) = f_j \quad \text{pour} \quad j = n, n-1, \ldots, n-k+1.$$



L'approximation de $y(t_{n+1})$ est alors définie par

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} p(\tau)\, d\tau. \tag{5.2}$$

Si l'on représente le polynôme $p(t)$ par la formule de Newton (voir le paragraphe II.1)

$$p(t) = \sum_{j=0}^{k-1} \Big(\prod_{i=0}^{j-1} (t - t_{n-i}) \Big) \cdot \delta^j f[t_n, t_{n-1}, \ldots, t_{n-j}] \tag{5.3}$$

la méthode (5.2) devient

$$y_{n+1} = y_n + \sum_{j=0}^{k-1} \Big(\int_{t_n}^{t_{n+1}} \prod_{i=0}^{j-1} (t - t_{n-i})\, dt \Big) \cdot \delta^j f[t_n, t_{n-1}, \ldots, t_{n-j}]. \tag{5.4}$$

Cas équidistant. Dans la situation $t_j = t_0 + jh$ les différences divisées peuvent être écrites sous la forme

$$\delta^j f[t_n, t_{n-1}, \ldots, t_{n-j}] = \frac{\nabla^j f_n}{j!\, h^j} \tag{5.5}$$

où $\nabla^0 f_n = f_n$, $\nabla f_n = f_n - f_{n-1}$, $\nabla^2 f_n = \nabla(\nabla f_n) = f_n - 2f_{n-1} + f_{n-2}, \ldots$ sont les ***différences finies régressives*** (à distinguer des différences finies progressives $\Delta f_n = f_{n+1} - f_n$). La formule (5.4) devient alors (poser $t = t_n + sh$)

$$y_{n+1} = y_n + h \sum_{j=0}^{k-1} \gamma_j \nabla^j f_n \tag{5.6}$$

où

$$\gamma_j = \frac{1}{j!} \int_0^1 \prod_{i=0}^{j-1} (i + s)\, ds = \int_0^1 \binom{s+j-1}{j} ds. \tag{5.7}$$

Les premiers coefficients γ_j sont donnés dans le tableau III.2.

TAB. III.2: Coefficients pour les méthodes d'Adams explicites

j	0	1	2	3	4	5	6	7	8
γ_j	1	$\frac{1}{2}$	$\frac{5}{12}$	$\frac{3}{8}$	$\frac{251}{720}$	$\frac{95}{288}$	$\frac{19087}{60480}$	$\frac{5257}{17280}$	$\frac{1070017}{3628800}$

Des cas particuliers sont:

$$\begin{aligned}
k = 1: &\quad y_{n+1} = y_n + h f_n && \text{(méthode d'Euler)} \\
k = 2: &\quad y_{n+1} = y_n + h\Big(\frac{3}{2} f_n - \frac{1}{2} f_{n-1}\Big) \\
k = 3: &\quad y_{n+1} = y_n + h\Big(\frac{23}{12} f_n - \frac{16}{12} f_{n-1} + \frac{5}{12} f_{n-2}\Big) \\
k = 4: &\quad y_{n+1} = y_n + h\Big(\frac{55}{24} f_n - \frac{59}{24} f_{n-1} + \frac{37}{24} f_{n-2} - \frac{9}{24} f_{n-3}\Big).
\end{aligned}$$

Si l'on veut appliquer cette méthode (par exemple avec $k = 3$) à la résolution de $y' = f(t, y)$, $y(t_0) = y_0$, il faut connaître trois approximations initiales y_0, y_1 et y_2. Ensuite, on peut utiliser la formule récursivement pour calculer y_3, y_4, etc. Adams a calculé la série de Taylor de la solution exacte (autour de la valeur initiale) pour déterminer les approximations initiales qui manquent. Evidemment, on peut aussi les obtenir par une méthode à un pas.

Remarque. Dans la construction de la méthode (5.4), on a utilisé le polynôme d'interpolation $p(t)$ en-dehors de l'intervalle $[t_{n-k+1}, t_n]$. On sait bien (voir le chapitre II) que ceci peut provoquer de grandes erreurs. La modification suivante est aussi due à J.C. Adams (1855).

Méthodes d'Adams implicites. L'idée est de considérer le polynôme $p^*(t)$ de degré k qui satisfasse

$$p^*(t_j) = f_j \qquad \text{pour} \qquad j = n+1, n, n-1, \ldots, n-k+1 \tag{5.8}$$

(attention: $f_{n+1} = f(t_{n+1}, y_{n+1})$ est encore inconnu) et de définir l'approximation numérique par

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} p^*(\tau)\, d\tau. \tag{5.9}$$

Comme précédemment, la formule de Newton donne

$$p^*(t) = \sum_{j=0}^{k} \Big(\prod_{i=0}^{j-1} (t - t_{n-i+1}) \Big) \cdot \delta^j f[t_{n+1}, t_n, \ldots, t_{n-j+1}] \tag{5.10}$$

et la méthode devient

$$y_{n+1} = y_n + \sum_{j=0}^{k} \Big(\int_{t_n}^{t_{n+1}} \prod_{i=0}^{j-1} (t - t_{n-i+1})\, dt \Big) \cdot \delta^j f[t_{n+1}, t_n, \ldots, t_{n-j+1}]. \tag{5.11}$$

Pour le cas équidistant, on a

$$y_{n+1} = y_n + h \sum_{j=0}^{k} \gamma_j^* \nabla^j f_{n+1} \tag{5.12}$$

où les coefficients γ_j^* sont donnés par (voir tableau III.3)

$$\gamma_j^* = \frac{1}{j!} \int_0^1 \prod_{i=0}^{j-1} (i - 1 + s)\, ds = \int_0^1 \binom{s+j-2}{j}\, ds. \tag{5.13}$$

TAB. III.3: Coefficients pour les méthodes d'Adams implicites

j	0	1	2	3	4	5	6	7	8
γ_j^*	1	$-\frac{1}{2}$	$-\frac{1}{12}$	$-\frac{1}{24}$	$-\frac{19}{720}$	$-\frac{3}{160}$	$-\frac{863}{60480}$	$-\frac{275}{24192}$	$-\frac{33953}{3628800}$

Des cas particuliers sont:

$$\begin{aligned}
k = 0: &\quad y_{n+1} = y_n + h f_{n+1} = y_n + h f(t_{n+1}, y_{n+1}) \\
k = 1: &\quad y_{n+1} = y_n + h\Big(\frac{1}{2} f_{n+1} + \frac{1}{2} f_n\Big) \\
k = 2: &\quad y_{n+1} = y_n + h\Big(\frac{5}{12} f_{n+1} + \frac{8}{12} f_n - \frac{1}{12} f_{n-1}\Big) \\
k = 3: &\quad y_{n+1} = y_n + h\Big(\frac{9}{24} f_{n+1} + \frac{19}{24} f_n - \frac{5}{24} f_{n-1} + \frac{1}{24} f_{n-2}\Big).
\end{aligned}$$

Chacune de ces formules représente une équation non linéaire pour y_{n+1}, de la forme

$$y_{n+1} = \eta_n + h\beta f(t_{n+1}, y_{n+1}) \tag{5.14}$$

(par exemple, pour $k = 2$ on a $\beta = 5/12$ et $\eta_n = y_n + h(8f_n - f_{n-1})/12$). On peut résoudre ce système par les méthodes du chapitre VI (méthode de Newton) ou simplement par la méthode des approximations successives.

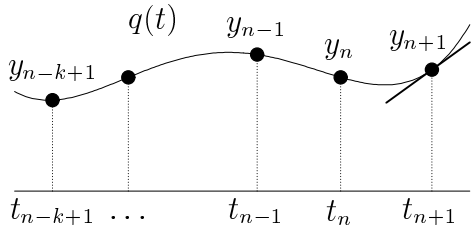
Méthodes prédicteur-correcteur. La solution de (5.14) est elle-même seulement une approximation de $y(t_{n+1})$. Ainsi, il n'est pas important de résoudre (5.14) à une très grande précision. L'idée est de calculer une première approximation par une méthode explicite et de corriger cette valeur (une ou plusieurs fois) par la formule (5.14). Avec cet algorithme, un pas de la méthode prend la forme suivante:

P: on calcule le prédicteur $\widehat{y}_{n+1} = y_n + h\sum_{j=0}^{k-1} \gamma_j \nabla^j f_n$ par la méthode d'Adams explicite; $\widehat{y}_{n+1}$ est déjà une approximation de $y(t_{n+1})$;
E: evaluation de la fonction: on calcule $\widehat{f}_{n+1} = f(t_{n+1}, \widehat{y}_{n+1})$;
C: l'approximation corrigée est alors donnée par $y_{n+1} = \eta_n + h\beta\widehat{f}_{n+1}$;
E: calculer $f_{n+1} = f(t_{n+1}, y_{n+1})$.

Cette procédure, qu'on dénote PECE, est la plus utilisée. D'autres possibilités sont: de faire plusieurs itérations, par exemple PECECE, ou d'omettre la dernière évaluation de f (c.-à-d. PEC) et de prendre $\widehat{f}_{n+1}$ à la place de f_{n+1} pour le pas suivant.

Méthodes BDF (backward differentiation formulas). Au lieu de travailler avec un polynôme qui passe par les f_j, on considère le polynôme $q(t)$ de degré k, défini par

$$q(t_j) = y_j \quad \text{pour} \quad j = n+1, n, \ldots, n-k+1$$



et on détermine y_{n+1} de façon telle que

$$q'(t_{n+1}) = f(t_{n+1}, q(t_{n+1})). \tag{5.15}$$

Comme dans (5.10), la formule de Newton donne

$$q(t) = \sum_{j=0}^{k}\Big(\prod_{i=0}^{j-1}(t - t_{n-i+1})\Big) \cdot \delta^j y[t_{n+1}, t_n, \ldots, t_{n-j+1}]. \tag{5.16}$$

Chaque terme de cette somme contient le facteur $(t - t_{n+1})$. Alors, on calcule facilement $q'(t_{n+1})$ et on obtient

$$\sum_{j=1}^{k}\Big(\prod_{i=1}^{j-1}(t_{n+1} - t_{n-i+1})\Big) \cdot \delta^j y[t_{n+1}, t_n, \ldots, t_{n-j+1}] = f(t_{n+1}, y_{n+1}). \tag{5.17}$$

Pour le cas équidistant, cette formule devient (utiliser (5.5))

$$\sum_{j=1}^{k} \frac{1}{j}\nabla^j y_{n+1} = h f_{n+1}. \tag{5.18}$$

Des cas particuliers sont:

$$\begin{aligned}
k=1: &\quad y_{n+1} - y_n = h f_{n+1} \\
k=2: &\quad \frac{3}{2}y_{n+1} - 2y_n + \frac{1}{2}y_{n-1} = h f_{n+1} \\
k=3: &\quad \frac{11}{6}y_{n+1} - 3y_n + \frac{3}{2}y_{n-1} - \frac{1}{3}y_{n-2} = h f_{n+1} \\
k=4: &\quad \frac{25}{12}y_{n+1} - 4y_n + 3y_{n-1} - \frac{4}{3}y_{n-2} + \frac{1}{4}y_{n-3} = h f_{n+1}
\end{aligned}$$

De nouveau, chaque formule définit implicitement l'approximation numérique y_{n+1} (les méthodes BDF sont très importantes pour la résolution de problèmes dits "raides", voir le paragraphe III.9).

Expérience numérique. Pour plusieurs valeurs de k, nous avons appliqué la méthode d'Adams explicite (en pointillés dans la figure III.7, $k = 1, 2, 3, 4$) ainsi que la méthode d'Adams implicite (sous la forme PECE, trait continu, $k = 0, 1, 2, 3, 4$) au problème (2.9). Comme dans la figure III.4, le travail numérique est dessiné en fonction de l'erreur globale à la fin de l'intervalle.

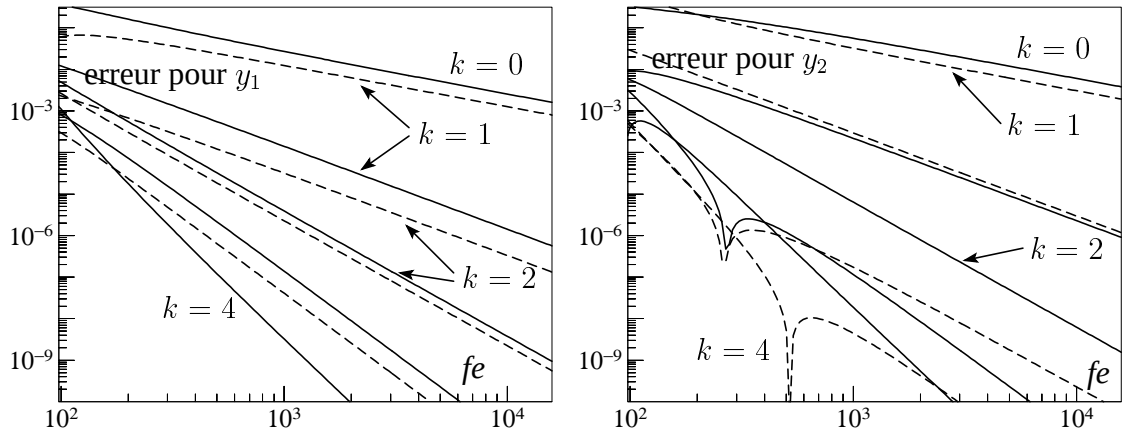


FIG. III.7: Erreur globale par rapport au travail numérique

On constate que cette erreur se comporte comme h^k pour les méthodes explicites et comme h^{k+1} pour les méthodes implicites. Pour pouvoir expliquer ce comportement, nous allons étudier l'erreur locale (ordre), la stabilité et la convergence des méthodes multipas.

III.6 Étude de l'erreur locale

Toutes les méthodes du paragraphe précédent sont de la forme

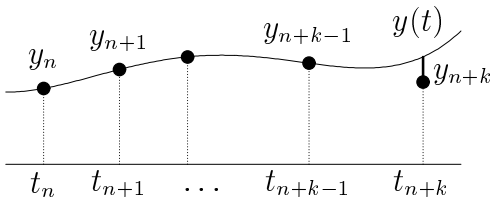
$$\sum_{i=0}^{k} \alpha_i y_{n+i} = h \sum_{i=0}^{k} \beta_i f_{n+i} \tag{6.1}$$

où $\alpha_k \neq 0$ et $|\alpha_0| + |\beta_0| > 0$. Une méthode est explicite si $\beta_k = 0$, sinon elle est implicite.

Définition 6.1 *Soit $y(t)$ une solution de $y' = f(t,y)$ et soit y_{n+k} la valeur obtenue par la méthode (6.1) en utilisant $y_i = y(t_i)$ pour $i = n, \ldots, n+k-1$ (valeurs sur la solution exacte, voir la figure). Alors,*

erreur locale $:= y(t_{n+k}) - y_{n+k}$.

On dit que la méthode (6.1) a l'ordre p si l'erreur locale est $\mathcal{O}(h^{p+1})$.



Théorème 6.2 *Une méthode multipas a l'ordre p, si et seulement si ses coefficients satisfont*

$$\sum_{i=0}^{k} \alpha_i = 0 \qquad \textit{et} \qquad \sum_{i=0}^{k} \alpha_i i^q = q \sum_{i=0}^{k} \beta_i i^{q-1} \qquad \textit{pour} \quad q = 1, \ldots, p. \tag{6.2}$$

Démonstration. Le développement en série de Taylor du défaut de (6.1) donne

$$\sum_{i=0}^{k} \alpha_i y(t+ih) - h \sum_{i=0}^{k} \beta_i y'(t+ih) \; = \sum_{q \geq 0} d_q \, y^{(q)}(t) \, \frac{h^q}{q!} \tag{6.3}$$

où $d_0 = \sum_i \alpha_i$ et $d_q = \sum_i \alpha_i i^q - q \sum_i \beta_i i^{q-1}$. Comme $y_i = y(t_i)$ pour $i = n, \ldots, n+k-1$ dans la définition 6.1, on a $f_i = f(t_i, y(t_i)) = y'(t_i)$ pour $i = n, \ldots, n+k-1$, et en soustrayant la formule (6.1) de (6.3) on obtient

$$\alpha_k \Big(y(t_{n+k}) - y_{n+k} \Big) - h\beta_k \Big(f(t_{n+k}, y(t_{n+k})) - f(t_{n+k}, y_{n+k}) \Big) \; = \sum_{q \geq 0} d_q \, y^{(q)}(t_n) \, \frac{h^q}{q!}.$$

et la condition que l'erreur locale soit $\mathcal{O}(h^{p+1})$ devient $d_0 = d_1 = \ldots = d_p = 0$. □

Exemple (méthode d'Adams explicite à k pas). Pour $q \leq k$, considérons l'équation différentielle $y' = qt^{q-1}$ avec comme solution $y(t) = t^q$. Dans cette situation, le polynôme $p(t)$ de (5.2) est égal à $f(t, y(t))$ et la méthode d'Adams explicite donne le résultat exact, c.-à-d. le défaut (6.3) est zéro. Ceci implique $d_q = 0$. Ainsi, l'ordre de cette méthode est $\geq k$ (on peut en fait montrer qu'il est égal à k).

De la même manière, on montre que la méthode d'Adams implicite a l'ordre $p = k + 1$ et la méthode BDF l'ordre $p = k$.

III.7 Stabilité

La structure simple des conditions d'ordre pour les méthodes multipas (voir (6.2)) permet de construire des méthodes avec un ordre maximal. Mais, ces méthodes sont-elles utiles?

Exemple (Dahlquist 1956). Posons $k = 2$ et construisons une méthode explicite ($\beta_2 = 0$; normalisation $\alpha_2 = 1$) avec un ordre maximal. Les conditions (6.2) avec $p = 3$ nous donnent la méthode d'ordre 3 suivante

$$y_{n+2} + 4y_{n+1} - 5y_n = h(4f_{n+1} + 2f_n). \tag{7.1}$$

Une application à l'équation différentielle $y' = y$, $y(0) = 1$ donne la formule de récurrence

$$y_{n+2} + 4(1-h)y_{n+1} - (5+2h)y_n = 0. \tag{7.2}$$

Pour résoudre (7.2), nous insérons $y_n = \zeta^n$ et obtenons l'équation caractéristique

$$\zeta^2 + 4(1-h)\zeta - (5+2h) = 0 \tag{7.3}$$

avec comme solution $\zeta_1 = 1 + h + \mathcal{O}(h^2)$, $\zeta_2 = -5 + \mathcal{O}(h)$. La solution de (7.2) est alors

$$y_n = C_1\zeta_1^n + C_2\zeta_2^n \tag{7.4}$$

où les constantes C_1 et C_2 sont déterminées par $y_0 = 1$ et $y_1 = e^h$ (on a choisi la valeur y_1 sur la solution exacte). Pour n grand, le terme $C_2\zeta_2^n \approx C_2(-5)^n$ est dominant et on n'a aucun espoir que la solution numérique converge vers la solution exacte e^x (figure III.8).

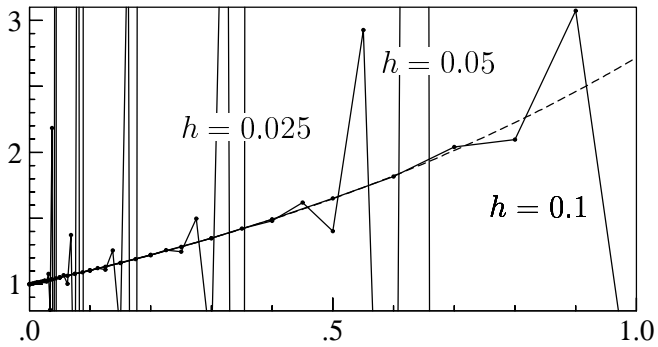


FIG. III.8: Instabilité de la méthode (7.1)

La raison de la divergence de la solution numérique dans l'exemple précédent est que le polynôme

$$\rho(\zeta) := \sum_{i=0}^{k} \alpha_i \zeta^i \tag{7.5}$$

possède une racine qui est plus grande que 1 en valeur absolue.

Pour trouver une condition nécessaire pour la convergence, considérons le problème $y' = 0$ avec des valeurs initiales $y_0, y_1, \ldots, y_{k-1}$ perturbées. La solution numérique y_n satisfait

$$\alpha_k y_{n+k} + \ldots + \alpha_0 y_n = 0 \tag{7.6}$$

et est donnée par une combinaison linéaire de

$$\begin{array}{ll} \zeta^n \ldots\ldots\ldots\ldots\ldots & \text{si } \zeta \text{ est une racine simple de } \rho(\zeta) = 0, \\ \zeta^n,\ n\zeta^n \ldots\ldots\ldots\ldots\ldots & \text{si } \zeta \text{ est une racine double de } \rho(\zeta) = 0, \\ \zeta^n,\ n\zeta^n, \ldots, n^{\ell-1}\zeta^n \ldots\ldots & \text{si } \zeta \text{ est une racine de multiplicité } \ell. \end{array}$$

Pour que la solution numérique reste bornée, il faut que les conditions de la définition suivante soient remplies.

Définition 7.1 *Une méthode multipas est stable, si les racines du polynôme $\rho(\zeta)$ satisfont*
i) si $\rho(\widehat{\zeta}) = 0$ alors $|\widehat{\zeta}| \leq 1$,
ii) si $\rho(\widehat{\zeta}) = 0$ et $|\widehat{\zeta}| = 1$ alors $\widehat{\zeta}$ est une racine simple de $\rho(\zeta)$.

Pour les méthodes d'Adams, on a

$$\rho(\zeta) = \zeta^{k-1}(\zeta - 1). \tag{7.7}$$

Elles sont donc stables. Les méthodes BDF sont stables seulement pour $k \leq 6$ (voir les exercices 17 et 18).

Remarque. Donnons encore sans démonstration un résultat intéressant qui s'appelle "la première barrière de Dahlquist". Pour une méthode stable, l'ordre p satisfait $p \leq k + 2$ (si k est pair), $p \leq k + 1$ (si k est impair) et $p \leq k$ (si la méthode est explicite).

III.8 Convergence des méthodes multipas

Pour l'étude de la convergence des méthodes multipas, nous nous contentons du cas équidistant, le cas général étant trop technique.

Théorème 8.1 *Supposons que les k valeurs de départ satisfassent $\|y(t_i) - y_i\| \leq C_0 h^p$ pour $i = 0, 1, \ldots, k-1$. Si la méthode multipas (6.1) est d'ordre p et stable, alors elle est convergente d'ordre p, c.-à-d. que l'erreur globale satisfait*

$$\|y(t_n) - y_n\| \leq Ch^p \qquad \textit{pour} \qquad x_n - x_0 = nh \leq \textit{Const}. \tag{8.1}$$

Démonstration. Le point essentiel de la démonstration est le suivant: on écrit formellement la méthode multipas (6.1) sous la forme d'une méthode à un pas et on applique les idées de la démonstration du paragraphe III.3.

Formulation comme une méthode à un pas. Avec $\alpha_k = 1$, la méthode multipas (6.1) devient

$$y_{n+k} = -\sum_{i=0}^{k-1} \alpha_i y_{n+i} + h\Psi(t_n, y_n, \ldots, y_{n+k-1}, h). \tag{8.2}$$

Pour une méthode explicite ($\beta_k = 0$), l'expression Ψ est donné par

$$\Psi(t_n, y_n, \ldots, y_{n+k-1}, h) = \sum_{i=0}^{k-1} \beta_i f(t_{n+i}, y_{n+i}),$$

sinon elle est définie implicitement. Considérons maintenant les super-vecteurs

$$Y_n := (y_{n+k-1}, \dots, y_{n+1}, y_n)^T$$

et écrivons la méthode (8.2) sous la forme

$$Y_{n+1} = AY_n + h\Phi(t_n, Y_n, h) \qquad \text{où} \tag{8.3}$$

$$A = \begin{pmatrix} -\alpha_{k-1} & -\alpha_{k-2} & \dots & -\alpha_1 & -\alpha_0 \\ 1 & 0 & \dots & 0 & 0 \\ & 1 & \ddots & & 0 \\ & & \ddots & \ddots & \vdots \\ & & & 1 & 0 \end{pmatrix} \quad \text{et} \quad \Phi(t, Y, h) = \begin{pmatrix} \Psi(t, Y, h) \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{8.4}$$

(si les y_j étaient déjà des vecteurs, il faudrait remplacer A dans (8.3) par $A \otimes I$, etc; cependant nous n'utiliserons pas cette notation jugée trop lourde).

Erreur locale. Considérons des valeurs $y_n, \dots, y_{n+k-1}$ sur la solution exacte, notons

$$Y(t_n) := (y(t_{n+k-1}), \dots, y(t_{n+1}), y(t_n))^T \tag{8.5}$$

et appliquons une fois la méthode multipas. Ceci donne

$$\widehat{Y}_{n+1} = AY(t_n) + h\Phi(t_n, Y(t_n), h).$$

La première composante de $\widehat{Y}_{n+1} - Y(t_{n+1})$ est exactement l'erreur locale (définition 6.1), tandis que les autres composantes sont égales à zéro. Comme la méthode a l'ordre p par hypothèse, nous avons

$$\|\widehat{Y}_{n+1} - Y(t_{n+1})\| \le C_1 h^{p+1} \qquad \text{pour} \qquad t_{n+1} - t_0 = (n+1)h \le \textit{Const}. \tag{8.6}$$

Propagation de l'erreur (stabilité). Considérons une deuxième solution numérique, définie par

$$Z_{n+1} = AZ_n + h\Phi(t_n, Z_n, h)$$

et estimons la différence $Y_{n+1} - Z_{n+1}$. Pour les méthodes d'Adams, on a

$$\begin{pmatrix} y_{n+k} - z_{n+k} \\ y_{n+k-1} - z_{n+k-1} \\ \vdots \\ y_{n+1} - z_{n+1} \end{pmatrix} = \begin{pmatrix} y_{n+k-1} - z_{n+k-1} \\ y_{n+k-1} - z_{n+k-1} \\ \vdots \\ y_{n+1} - z_{n+1} \end{pmatrix} + h \begin{pmatrix} \Psi(t_n, Y_n, h) - \Psi(t_n, Z_n, h) \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

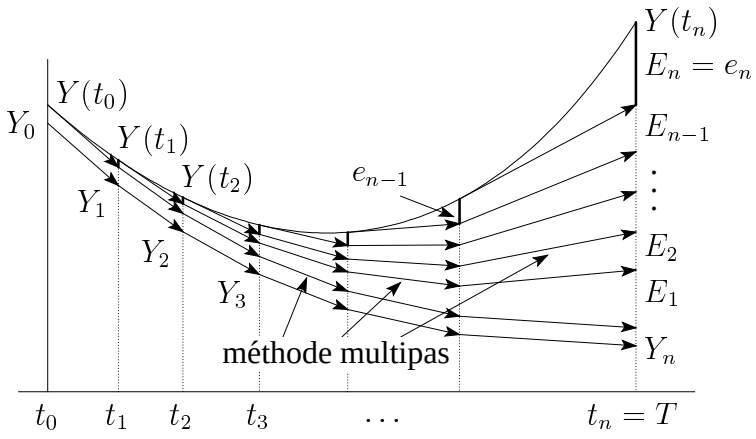


FIG. III.9: Estimation de l'erreur globale pour des méthodes multipas

En utilisant la norme infinie et une condition de Lipschitz pour Ψ (qui est une conséquence de celle de $f(t,y)$), on obtient

$$\|Y_{n+1} - Z_{n+1}\| \leq (1 + h\Lambda)\|Y_n - Z_n\|. \tag{8.7}$$

Pour une méthode générale, on est obligé de choisir une autre norme pour arriver à (8.7). La stabilité de la méthode implique que ceci est possible (voir Hairer, Nørsett & Wanner (1993), paragraphe III.4).

Accumulation des erreurs propagées. Cette partie de la démonstration est exactement la même que pour les méthodes à un pas (voir le paragraphe III.3 et la figure III.9). Au lieu de (3.2) et (3.5), on utilise (8.6) et (8.7). □

III.9 Equations différentielles raides (stiff)

> The most pragmatical opinion is also historically the first one (Curtiss & Hirschfelder 1952): *stiff equations are equations where certain implicit methods, in particular BDF, perform better, usually tremendously better, than explicit ones.* (Hairer & Wanner 1991)

L'exemple 1.2 d'une équation différentielle, modélisant une réaction chimique, a montré qu'une méthode de Runge–Kutta explicite est obligée de prendre des pas d'intégration très petits pour obtenir une approximation acceptable. Une caractéristique de cette équation différentielle est que la solution cherchée est très lisse et les autres solutions s'approchent rapidement de celle-ci.

Pour mieux comprendre le phénomène de l'exemple 1.2, considérons le problème plus simple

$$\varepsilon y' = -y + \cos t, \qquad 0 < \varepsilon \ll 1 \tag{9.1}$$

qui possède les mêmes caractéristiques (voir la figure III.10).

Solution exacte. L'équation différentielle (9.1) est linéaire inhomogène. Cherchons une solution particulière de la forme $y(t) = A\cos t + B\sin t$. En introduisant cette fonction dans (9.1)

$$-\varepsilon A\sin t + \varepsilon B\cos t = -A\cos t - B\sin t + \cos t,$$

une comparaison des coefficients donne $A = 1/(1+\varepsilon^2)$ et $B = \varepsilon/(1+\varepsilon^2)$. Comme la solution générale de (9.1) est la somme de la solution générale de l'équation homogène et d'une solution particulière, nous obtenons

$$y(t) = e^{-t/\varepsilon}C + \cos t + \varepsilon\sin t + \mathcal{O}(\varepsilon^2). \tag{9.2}$$

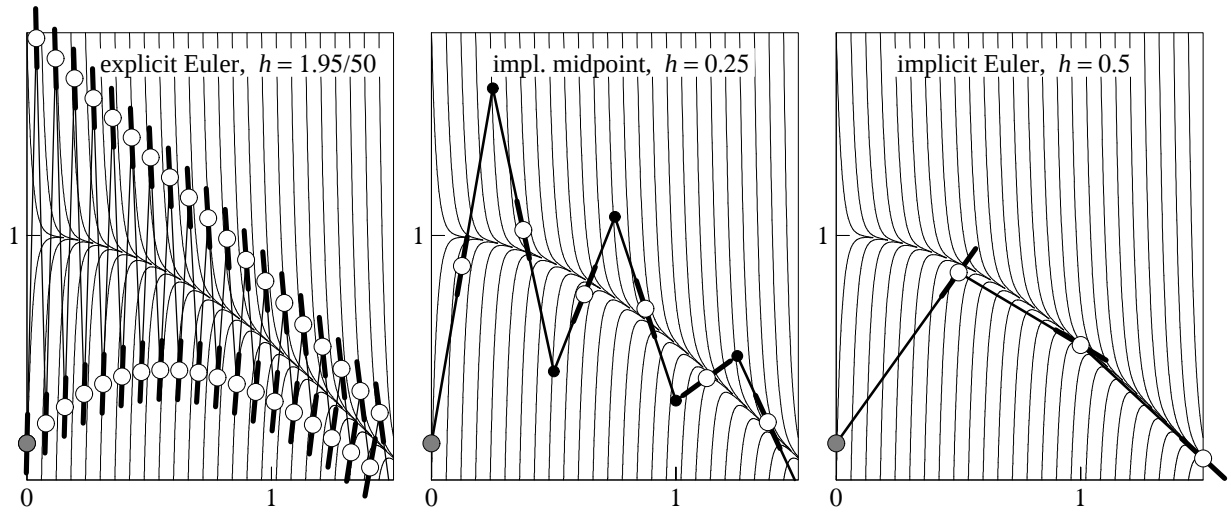


FIG. III.10: Solutions exactes et numériques pour le problème (9.1) de Curtiss & Hirschfelder, $\varepsilon = 1/50$.

Solution numérique (Euler explicite). La méthode d'Euler explicite $y_{n+1} = y_n + hf(t_n, y_n)$, appliquée au problème (9.1) avec des pas constants, donne avec $t_n = nh$

$$y_{n+1} = \Big(1 - \frac{h}{\varepsilon}\Big) y_n + \frac{h}{\varepsilon} \cos t_n. \tag{9.3}$$

Ceci est une équation aux différences finies qui est linéaire et inhomogène. La solution est obtenue comme pour une équation différentielle. On cherche d'abord une solution particulière de la forme $y_n = A \cos t_n + B \sin t_n$. On l'introduit dans (9.3) et, en utilisant $t_{n+1} = t_n + h$ et les théorèmes d'addition pour $\sin(t_n + h)$ et $\cos(t_n + h)$, on obtient ainsi

$$\begin{aligned} A\Big(\cos t_n \cdot \cos h - \sin t_n \cdot \sin h\Big) + B\Big(\sin t_n \cdot \cos h + \cos t_n \cdot \sin h\Big) \\ = \Big(1 - \frac{h}{\varepsilon}\Big)\Big(A \cos t_n + B \sin t_n\Big) + \frac{h}{\varepsilon} \cos t_n. \end{aligned}$$

En comparant les coefficients de $\cos t_n$ et $\sin t_n$, on obtient deux équations linéaires pour A et B dont la solution est $A = 1 + \mathcal{O}(h\varepsilon)$ et $B = \varepsilon + \mathcal{O}(h^2\varepsilon)$. En ajoutant la solution générale de l'équation homogène à la solution particulière, on obtient (dessin à gauche de la figure III.10)

$$y_n = \Big(1 - \frac{h}{\varepsilon}\Big)^n C + \cos t_n + \varepsilon \sin t_n + \mathcal{O}(h\varepsilon). \tag{9.4}$$

On voit que la solution numérique y_n est proche de la solution exacte seulement si $|1 - h/\varepsilon| < 1$, c.-à-d. si $h < 2\varepsilon$. Si ε est très petit (par exemple $\varepsilon = 10^{-6}$) une telle restriction est inacceptable.

Solution numérique (Euler implicite). Pour la méthode d'Euler implicite, le même calcul donne

$$\Big(1 + \frac{h}{\varepsilon}\Big) y_{n+1} = y_n + \frac{h}{\varepsilon} \cos t_{n+1}, \tag{9.5}$$

dont la solution peut être écrite sous la forme

$$y_n = \Big(1 + \frac{h}{\varepsilon}\Big)^{-n} C + \cos t_n + \varepsilon \sin t_n + \mathcal{O}(h\varepsilon). \tag{9.6}$$

Cette fois-ci nous n'avons pas de restriction sur la longueur du pas, car $|(1 + h/\varepsilon)^{-1}| < 1$ pour tout $h > 0$. Le dessin à droite de la figure III.10 illustre bien la bonne approximation même si h est très grand.

Le calcul précédent a montré que ce n'est pas la solution particulière qui pose des difficultés à la méthode explicite, mais c'est l'approximation de la solution de l'équation homogène $\varepsilon y' = -y$. Nous considérons donc le problème un peu plus général

$$y' = \lambda y \tag{9.7}$$

comme *équation de test* (Dahlquist 1963). Sa solution exacte est $y(t) = e^{\lambda t} C$ et elle reste bornée pour $t \geq 0$ si $\Re\lambda \leq 0$. La solution numérique d'une méthode de Runge–Kutta ou d'une méthode multipas, appliquée avec des pas constants au problème (9.7), ne dépend que du produit $h\lambda$. Il est alors intéressant d'étudier pour quelle valeur de $h\lambda$ la solution numérique reste bornée.

Définition 9.1 (A-stabilité) *Considérons une méthode dont la solution numérique $\{y_n\}_{n\geq 0}$ pour l'équation de test (9.7) est une fonction de $z = h\lambda$. Alors, l'ensemble*

$$S := \{z \in \mathbb{C}\,;\ \{y_n\}_{n\geq 0} \ \textit{est bornée}\} \tag{9.8}$$

s'appelle domaine de stabilité *de la méthode. On dit que la méthode est* A-stable *si*

$$S \supset \mathbb{C}^- \qquad \textit{où} \qquad \mathbb{C}^- = \{z \in \mathbb{C}\,;\ \Re z \leq 0\}.$$

Pour la méthode d'Euler explicite le domaine de stabilité est $S = \{z\,;\ |1 + z| \leq 1\}$, le disque de rayon 1 et de centre -1. Pour la méthode d'Euler implicite il est $S = \{z\,;\ |1 - z| \geq 1\}$, l'extérieur du disque de rayon 1 et de centre $+1$. Seulement la méthode implicite est A-stable.

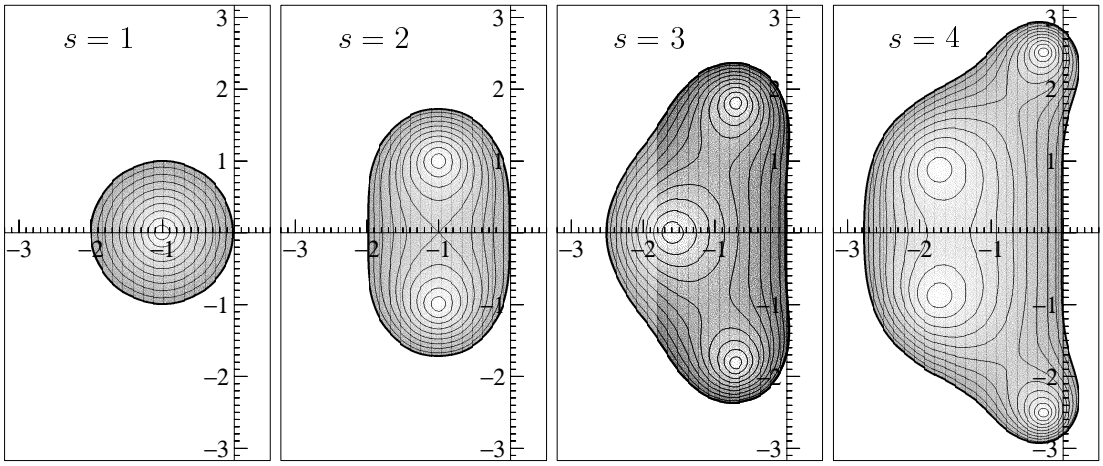


FIG. III.11: Domaines de stabilité pour les méthodes de Runge–Kutta avec s étages et d'ordre $p = s$

Domaine de stabilité des méthodes à un pas. Pour une méthode de Runge–Kutta, la solution numérique est de la forme $y_{n+1} = R(h\lambda)y_n$, où la fonction $R(z)$ s'appelle *fonction de stabilité* (voir l'exercice 2). Le domaine de stabilité est alors

$$S = \{z \; ; \; |R(z)| \leq 1\}.$$

Si la méthode explicite à s étages (définition 2.1) a l'ordre $p = s$, $R(z)$ est le polynôme de degré s obtenu par troncature de la série $e^z = 1 + z + \frac{1}{2!}z^2 + \dots$. Les domaines de stabilité pour ces méthodes d'ordre 1 jusqu'à 4 sont dessinés dans la figure III.11. Comme $R(z)$ est un polynôme, S est borné et la condition de stabilité $h\lambda \in S$ impose une restriction sévère à h. Ces méthodes ne sont donc pas recommandées pour la résolution des équations différentielles raides.

Domaine de stabilité des méthodes multipas. Si l'on applique une méthode multipas (6.1) au problème (9.7), on obtient l'équation aux différences finies

$$\sum_{j=0}^{k}(\alpha_j - h\lambda\beta_j)y_{n+j} = 0. \tag{9.9}$$

La solution numérique est une combinaison linéaire de $\zeta_j(h\lambda)^n$, $j = 1, \dots, k$, où $\zeta_j(z)$ est un zéro du polynôme caractéristique (en supposant que les zéros soient distincts)

$$\rho(\zeta) - z\sigma(\zeta) = 0, \qquad \rho(\zeta) = \sum_{j=0}^{k}\alpha_j\zeta^j, \qquad \sigma(\zeta) = \sum_{j=0}^{k}\beta_j\zeta^j. \tag{9.10}$$

Le domaine de stabilité S d'une méthode multipas est alors l'ensemble des $z \in \mathbb{C}$, tel que toutes les racines de (9.10) sont majorées par 1. Pour étudier et dessiner S, il est plus simple de considérer le bord ∂S, car (par continuité de $\zeta_j(z)$)

$$z \in \partial S \qquad \Rightarrow \qquad \rho(e^{i\theta}) - z\sigma(e^{i\theta}) = 0 \quad \text{pour un } \theta \in [0, 2\pi).$$

Il suffit alors de dessiner la courbe $\theta \mapsto \rho(e^{i\theta})/\sigma(e^{i\theta})$ (la "root locus curve") qui sépare l'ensemble S du reste. Pour savoir, quelle composante connexe appartient à S, il suffit de le vérifier pour un seul point.

Les méthodes d'Adams explicites et implicites (à l'exception de la méthode d'Euler implicite et de la règle du trapèze) ont toutes un domaine de stabilité borné et petit. Ces méthodes ne sont donc pas utilisables pour des problèmes raides. Pour les méthods BDF, par contre, le domaine de

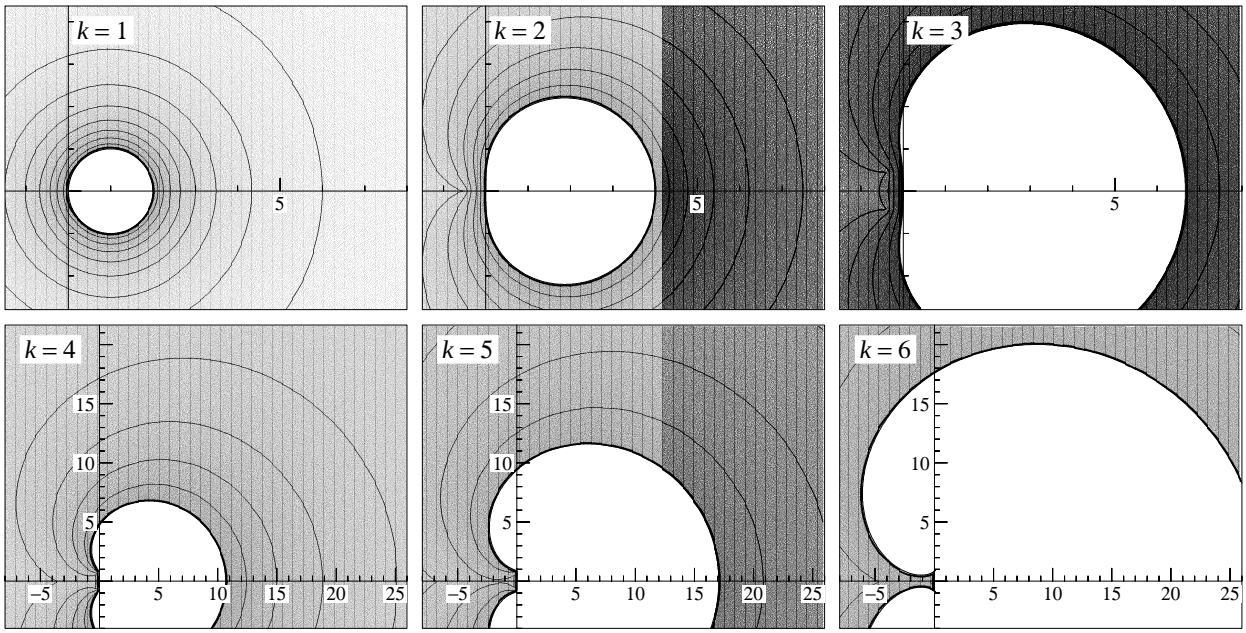


FIG. III.12: Domaines de stabilité pour les méthodes BDF à k pas.

stabilité contient une grande partie du demi-plan gauche (voir la figure III.12). La méthode BDF avec $k = 2$ (ordre $p = 2$) est même A-stable. Ceci est une conséquence du fait que la "root locus curve"

$$z = \rho(e^{i\theta})/\sigma(e^{i\theta}) = \frac{3}{2} - 2e^{-i\theta} + \frac{1}{2}e^{-2i\theta}$$

satisfait

$$\Re z = \frac{3}{2} - 2\cos\theta + \frac{1}{2}\cos 2\theta = (1-\cos\theta)^2 \geq 0.$$

Les méthodes BDF sont beaucoup utilisées pour résoudre des équations différentielles raides même si elles sont A-stables seulement pour $k \leq 2$ (voir la figure III.12). La célèbre *barrière de Dahlquist* (1963) dit que l'ordre d'une méthode multipas A-stable ne peut pas être plus grand que 2. Ce résultat négatif a motivé la recherche d'autres méthodes d'intégration qui permettent de combiner A-stabilité avec un ordre élevé.

Méthodes de Runge–Kutta implicites (Radau IIA)

Comme pour la dérivation des méthodes de Runge–Kutta explicites, nous partons de la formule intégrée (2.3) de l'équation différentielle. Nous appliquons une formule de quadrature avec $c_s = 1$ ayant l'ordre maximal $2s-1$ (formules de Radau, à comparer avec les exercices I.10 et I.16). Par exemple, pour $s = 2$ on a

$$y(t_0+h) = y(t_0) + \frac{h}{4}\Big(3f\Big(t_0+\frac{h}{3}, y(t_0+\frac{h}{3})\Big) + f\Big(t_0+h, y(t_0+h)\Big)\Big) + \mathcal{O}(h^4). \qquad (9.11)$$

Pour approximer la valeur $y(t_0+h/3)$, nous intégrons l'équation différentielle de t_0 à $t_0+h/3$ et nous appliquons une formule de quadrature qui utilise les mêmes évaluations de f que dans (9.11):

$$y(t_0+\frac{h}{3}) = y(t_0) + \frac{h}{12}\Big(5f\Big(t_0+\frac{h}{3}, y(t_0+\frac{h}{3})\Big) - f\Big(t_0+h, y(t_0+h)\Big)\Big) + \mathcal{O}(h^3). \qquad (9.12)$$

En supprimant les termes du reste et en notant k_1 et k_2 les deux évaluations de f, nous arrivons à

$$\begin{aligned} k_1 &= f\Big(t_0+\frac{h}{3}, y_0+\frac{h}{12}(5k_1-k_2)\Big) \\ k_2 &= f\Big(t_0+h, y_0+\frac{h}{4}(3k_1+k_2)\Big) \\ y_1 &= y_0+\frac{h}{4}\Big(3k_1+k_2\Big) \end{aligned} \qquad (9.13)$$

TAB. III.4: Coefficients c_i, a_{ij} et b_i pour les méthodes Radau IIA avec $s = 2$ et $s = 3$, ordre $p = 2s - 1$

$$\begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & \frac{-1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array} \qquad \begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{4+\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ 1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \\ \hline & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{array}$$

qui est une méthode de Runge–Kutta comme dans la définition 2.1, mais où la matrice (a_{ij}) n'est plus triangulaire inférieure (voir aussi le tableau III.4). Les deux premières équations de (9.13) constituent un système non linéaire pour k_1 et k_2, qu'il faut résoudre avec les techniques du chapitre VI (méthode de Newton simplifiée).

Lemme 9.2 *La méthode (9.13) a l'ordre* 3 *et elle est A-stable.*

Démonstration. L'ordre 3 est une conséquence des formules (9.11) et (9.12). Il suffit que la formule (9.12) soit d'ordre 2 car le terme correspondant dans (9.11) est mulitplié par h.

A-stabilité. Si l'on applique la méthode à $y' = \lambda y$, on obtient avec $z = h\lambda$

$$hk_1 = zy_0 + \frac{z}{12}(5hk_1 - hk_1), \qquad hk_2 = zy_0 + \frac{z}{4}(3hk_1 + hk_2).$$

On résoud ce système linéaire pour hk_1 et hk_2, et on introduit la solution dans la troisième formule de (9.13). Ceci donne $y_1 = R(z)y_0$ avec comme fonction de stabilité

$$R(z) = \frac{P(z)}{Q(z)} = \frac{1 + z/3}{1 - 2z/3 + z^2/6}. \tag{9.14}$$

Sur l'axe imaginaire, on a

$$|Q(iy)|^2 - |P(iy)|^2 = \left(1 - \frac{y^2}{6}\right)^2 + \frac{4}{9}\,y^2 - \left(1 + \frac{y^2}{9}\right) = \frac{1}{36}\,y^4 \geq 0.$$

Ceci implique $|Q(iy)| \geq |P(iy)|$ et aussi $|R(iy)| \leq 1$. Les singularités de $R(z)$ (les zéros de $Q(z)$) sont $z_{12} = 2 \pm i\sqrt{2}$ dans le demi-plan droit. Donc $R(z)$ est analytique dans le demi-plan gauche et, par le principe du maximum, $R(z)$ est majoré par 1 pour $\Re z \leq 0$. □

Cette contruction peut être généralisée pour obtenir des méthodes de Runge-Kutta implicites qui sont A-stables et d'ordre $2s - 1$. Elles s'appellent méthodes Radau IIA et sont, comme les méthodes BDF, souvent employées pour résoudre des équations différentielles raides. La méthode RADAU5, utilisée pour le calcul de la figure III.2, est celle avec $s = 3$ et ordre $p = 5$. Les coefficients de la méthode (9.13) et de celle avec $s = 3$ sont données dans le tableau III.4.

Remarque. Après la publication remarquable de Dahlquist en 1963, la recherche sur la résolution des équations différentielles raides a rapidement pris une place importante en analyse numérique. Des nouvelles méthodes d'intégration, des nouvelles théories (par exemple, étoiles d'ordre) et des programmes informatiques performants ont été dévelopé. Plus de détails peuvent être trouvés dans le livre *"Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems"* par Hairer & Wanner (1996).

Beaucoup plus récente est l'étude sur les propriétés géométriques des intégrateurs numériques. Ca sera le contenu du paragraphe suivant. Le livre *"Geometric Numerical Integration"* par Hairer, Lubich & Wanner (2002) est consacré à ce sujet.

III.10 Intégration géométrique

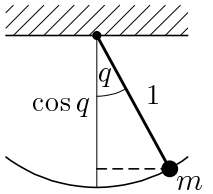
On cherche des méthodes d'intégration pour des problèmes similaires à l'exemple 1.3 qui donnent des bonnes approximations pour un calcul à long terme. Avec q_i à la place de y_i et avec $p_i = m_i q_i'$ l'équation différentielle (1.3) peut être écrite sous la forme (système Hamiltonien)

$$p' = -\nabla U(q), \qquad q' = \nabla T(p), \tag{10.1}$$

où $T(p) = \frac{1}{2}\sum_i m_i^{-1} p_i^T p_i$ est l'énergie cinétique et $U(q) = -G\sum_{i>j} m_i m_j / \|q_i - q_j\|$ est l'énergie potentielle du système. Pour simplifier la présentation, nous supposons par la suite que p et q sont des fonctions scalaires (par conséquent, $\nabla U(q) = U'(q)$ et $\nabla T(p) = T'(p)$).

Un exemple typique et intéressant est l'équation du pendule mathématique où l'énergie cinétique est $T(p) = p^2/2$, l'énergie potentielle $U(q) = -\cos q$ et l'énergie totale

$$H(p,q) = \frac{1}{2}\,p^2 - \cos q.$$



Le système (10.1) possède deux propriétés très intéressantes:

Conservation de l'énergie. Un calcul direct montre que l'énergie totale $H(p,q) = T(p) + U(q)$ reste constante le long des solutions de (10.1); figure III.13 (gauche). Ceci découle du fait que

$$\frac{d}{dt}\,H(p(t),q(t)) = T'(p(t))p'(t) + U'(q(t))q'(t) = 0.$$

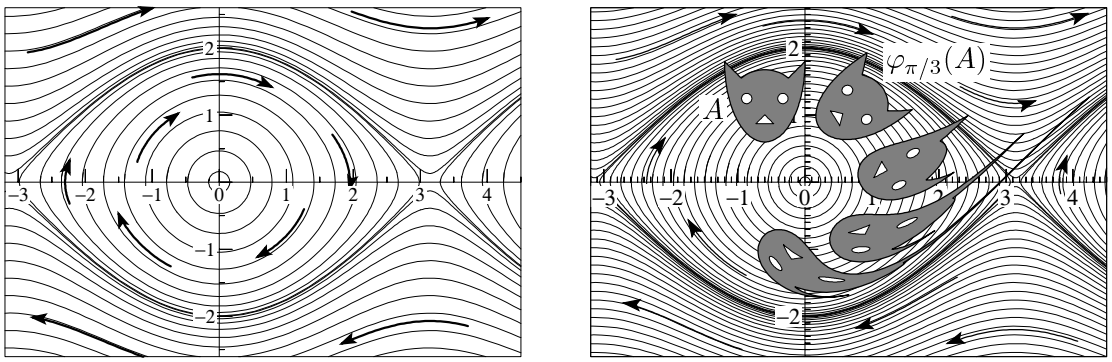


FIG. III.13: Conservation de l'énergie (gauche) et symplecticité (droite) du flot pour l'équation du pendule.

Symplecticité (préservation de l'aire). Nous utilisons la notation $\varphi_t(p_0,q_0) := (p(t),q(t))^T$ pour le flot de l'équation différentielle (10.1) et nous allons démontrer que, pour un sous-ensemble A de l'espace (p,q), l'aire de A reste invariante par rapport à l'application φ_t, c.-à-d.

$$\text{aire}(\varphi_t(A)) = \text{aire}(A) \qquad \text{pour tout } t \tag{10.2}$$

(voir la figure III.13, droite). Comme $\text{aire}(A) = \iint_A d(p_0,q_0)$ et[2]

$$\text{aire}(\varphi_t(A)) = \iint_{\varphi_t(A)} d(p,q) = \iint_A \left|\det\Big(\frac{\partial \varphi_t(p_0,q_0)}{\partial(p_0,q_0)}\Big)\right| d(p_0,q_0),$$

[2]La formule de changement de variables pour les intégrales doubles peut être trouvée dans le livre "Analyse au fil de l'histoire" de Hairer & Wanner à la page 338.

il suffit de démontrer que le déterminant de la matrice Jacobienne de φ_t est égal à 1 en valeur absolue. En dérivant l'équation différentielle (10.1) par rapport à la valeur initiale p_0, on obtient

$$\frac{\partial p'(t)}{\partial p_0} = -U''(q(t))\,\frac{\partial q(t)}{\partial p_0}, \qquad \frac{\partial q'(t)}{\partial p_0} = T''(p(t))\,\frac{\partial p(t)}{\partial p_0}.$$

Une différentiation par rapport à q_0 donne des formules analogues. En changeant l'ordre de la différentiation, on en déduit que

$$\frac{d}{dt}\det\Big(\frac{\partial\varphi_t(p_0,q_0)}{\partial(p_0,q_0)}\Big) = \frac{d}{dt}\Big(\frac{\partial p(t)}{\partial p_0}\frac{\partial q(t)}{\partial q_0} - \frac{\partial p(t)}{\partial q_0}\frac{\partial q(t)}{\partial p_0}\Big) = \ldots = 0.$$

Il suit de $\varphi_0(p_0,q_0) = (p_0,q_0)^T$ que le déterminant de la Jacobienne de φ_0 vaut 1. Par conséquent, il vaut 1 pour tout t, ce qui démontre (10.2).

Dans une simulation numérique d'un système (10.1) on aimerait que les propriétés géométriques du flot exact soient préservées aussi bien que possible. Regardons ce qui se passe avec des méthodes classiques. Les figures III.14 et III.15 montrent le résultat pour la méthode d'Euler explicite et pour la méthode d'Euler implicite. Pour la première, la solution numérique tourne vers l'extérieur, l'énergie augmente et l'aire d'un ensemble croît. Pour la méthode implicite, c'est exactement l'inverse. Aucune de ces deux méthodes donne une approximation de la solution qui est qualitativement acceptable.

Méthode d'Euler symplectique. Nous traitons une équation de (10.1) par la méthode d'Euler explicite et l'autre par la méthode implicite. Ceci donne

$$\begin{aligned} p_{n+1} &= p_n - h\nabla U(q_n) \\ q_{n+1} &= q_n + h\nabla T(p_{n+1}) \end{aligned} \qquad \text{ou} \qquad \begin{aligned} p_{n+1} &= p_n - h\nabla U(q_{n+1}) \\ q_{n+1} &= q_n + h\nabla T(p_n). \end{aligned} \tag{10.3}$$

La variante (A) est la méthode de gauche, la variante (B) celle de droite.

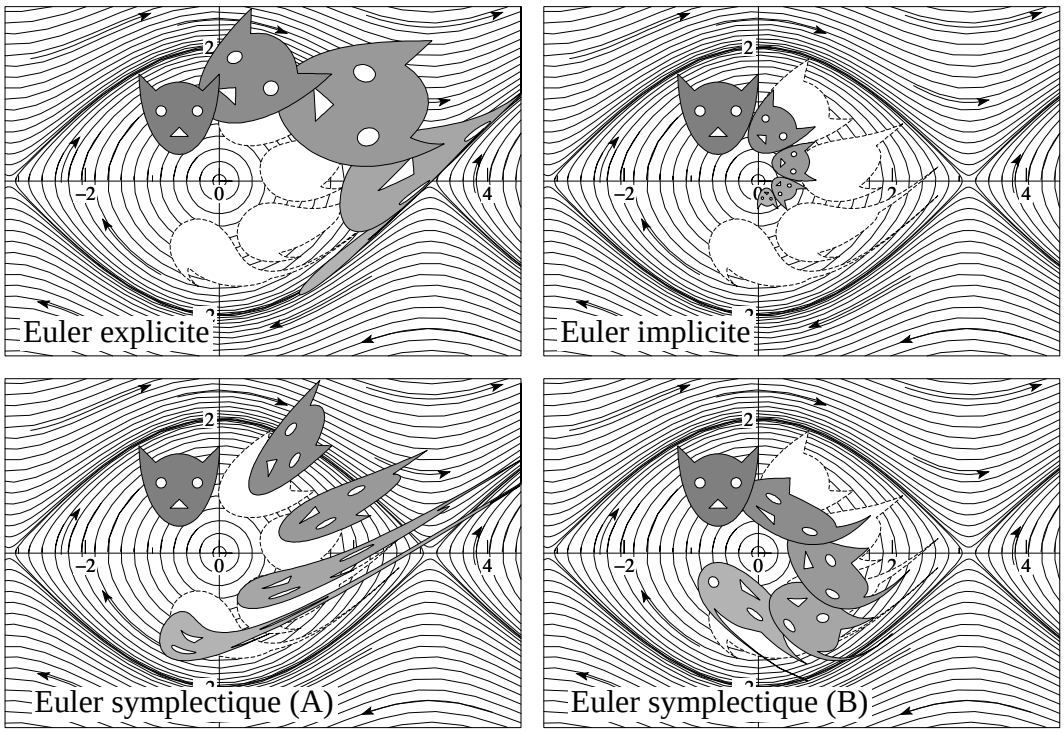


FIG. III.14: Illustration du flot numérique pour l'équation du pendule, $h = \pi/4$.

Les deux méthodes sont parfaitement explicites. Pour la première on calcule d'abord p_{n+1} et ensuite q_{n+1}; pour la deuxième dans l'ordre inverse. La figure III.14 montre une nette amélioration pour ce qui concerne la conservation de l'aire.

Lemme 10.1 *Si $\Phi_h : (p_n, q_n) \mapsto (p_{n+1}, q_{n+1})$ représente une des méthodes de (10.3), alors Φ_h préserve l'aire, c.-à-d.* aire$(\Phi_h(A)) =$ aire(A). *On dit que cette méthode est symplectique.*

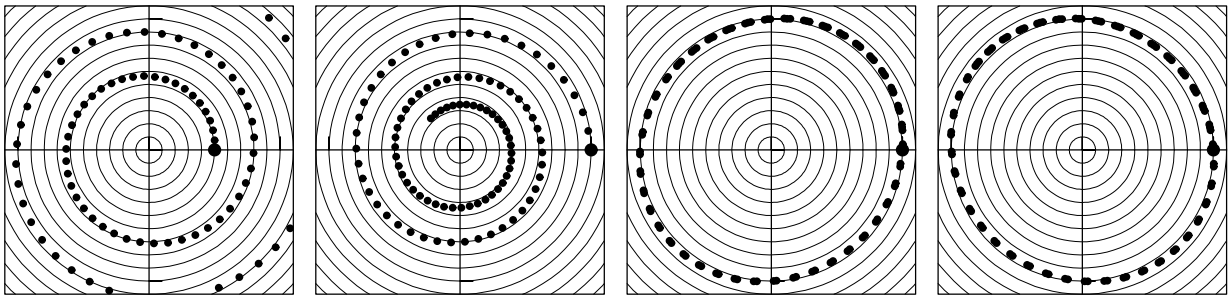
Démonstration. Nous partageons le membre droit du système (10.1) comme

$$\begin{array}{ll} p' = 0 \\ q' = \nabla T(p) \end{array} \qquad \text{et} \qquad \begin{array}{ll} p' = -\nabla U(q) \\ q' = 0. \end{array} \tag{10.4}$$

Les deux systèmes peuvent être résolus de manière exacte. Leurs solutions sont

$$\begin{array}{ll} p(t) = p_0 \\ q(t) = q_0 + t\nabla T(p_0) \end{array} \qquad \text{et} \qquad \begin{array}{ll} p(t) = p_0 - t\nabla U(q_0) \\ q(t) = q_0. \end{array}$$

Les flots $\varphi_t^{(T)}$ et $\varphi_t^{(U)}$ des deux systèmes (10.4) préservent l'aire car ils sont les deux sous la forme (10.1). L'affirmation du lemme est alors une conséquence du fait que les deux méthodes (10.3) sont respectivement $\varphi_h^{(U)} \circ \varphi_h^{(T)}$ et $\varphi_h^{(T)} \circ \varphi_h^{(U)}$. □



explicite implicite symplectique (A) symplectique (B)

FIG. III.15: Solution numérique de différentes méthodes d'Euler appliquées à l'oscillateur harmonique (10.5) avec $h = 0.15$. La valeur initiale est le grand point noir.

On voit par des exemples très simples qu'on n'a pas de *conservation de l'énergie totale*, ni pour les méthodes classiques (Euler explicite ou implicite) ni pour la méthode d'Euler symplectique. L'expérience de la figure III.15 montre la solution numérique pour l'*oscillateur harmonique* qui est un système linéaire $p' = -q$, $q' = p$ possèdant

$$H(p, q) = \frac{1}{2}\Big(p^2 + q^2\Big) \tag{10.5}$$

comme énergie totale. Seulement les deux variantes de la méthode d'Euler symplectique donnent une solution numérique qui reste proche du cercle $H(p, q) = Const$ (solution exacte).

Lemme 10.2 *Pour l'oscillateur harmonique (10.5), la solution numérique (p_n, q_n) de la méthode d'Euler symplectique reste sur une courbe fermée (l'ellipse $p^2 + q^2 \pm hpq = Const$) pour tout n.*

Démonstration. Considérons, par exemple, la première méthode dans (10.3) et appliquons-la à l'oscillateur harmonique (10.5). On obtient ainsi

$$\begin{pmatrix} 1 & 0 \\ -h & 1 \end{pmatrix} \begin{pmatrix} p_{n+1} \\ q_{n+1} \end{pmatrix} = \begin{pmatrix} 1 & -h \\ 0 & 1 \end{pmatrix} \begin{pmatrix} p_n \\ q_n \end{pmatrix}. \tag{10.6}$$

Une multiplication par (p_{n+1}, q_{n+1}) donne la première égalité de

$$p_{n+1}^2 + q_{n+1}^2 - hp_{n+1}q_{n+1} = p_{n+1}p_n + q_{n+1}q_n - hp_{n+1}q_n = p_n^2 + q_n^2 - hp_nq_n.$$

et une multiplication par (p_n, q_n) la deuxième. On en déduit que $p_n^2 + q_n^2 - hp_nq_n = Const$, ce qui signifie que la solution numérique (p_n, q_n) reste sur une ellipse pour tout n. □

La méthode d'Euler symplectique est malheureusement seulement une méthode d'ordre $p = 1$. Peut-on trouver des méthodes avec un ordre plus élevé qui possèdent le même comportement concernant la conservation de l'aire et de l'énergie totale? Voici une méthode d'ordre deux qui est la plus importante dans le contexte de l'intégration géométrique. Elle est beaucoup utilisée dans des simulations en dynamique moléculaire et elle est la base pour plusieurs généralisations.

La méthode "Störmer–Verlet". Pour introduire plus de symétrie dans la discrétisation, nous considérons la composition d'un demi-pas d'une des variantes de la méthode d'Euler symplectique avec un demi-pas de l'autre. Ceci conduit à

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2}\nabla U(q_n) \\ q_{n+1} &= q_n + h\nabla T(p_{n+1/2}) \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2}\nabla U(q_{n+1}) \end{aligned} \qquad \text{ou} \qquad \begin{aligned} q_{n+1/2} &= q_n + \frac{h}{2}\nabla T(p_n) \\ p_{n+1} &= p_n - h\nabla U(q_{n+1/2}) \\ q_{n+1} &= q_{n+1/2} + \frac{h}{2}\nabla T(p_{n+1}). \end{aligned} \tag{10.7}$$

De nouveau on appelle variante (A) la méthode à gauche et variante (B) celle de droite.

Lemme 10.3 *Les deux variantes (10.7) de la méthode "Störmer–Verlet" sont symplectiques, c.-à-d., considérées comme application $(p_n, q_n) \mapsto (p_{n+1}, q_{n+1})$ elles préservent l'aire.*

Pour l'oscillateur harmonique (10.5), leur solution numérique (p_n, q_n) reste sur une courbe fermée (l'ellipse $p^2 + (1 - h^2/4)q^2 = Const$) pour tout n.

Démonstration. Avec la notation de la démonstration du lemme 10.1, nous observons que la variante (A) correspond à l'application $\varphi_{h/2}^{(U)} \circ \varphi_h^{(T)} \circ \varphi_{h/2}^{(U)}$ et la variante (B) à $\varphi_{h/2}^{(T)} \circ \varphi_h^{(U)} \circ \varphi_{h/2}^{(T)}$. Comme les flots des deux systèmes (10.4) préservent l'aire, c'est vrai aussi pour la composition.

Pour le problème $p' = -q$, $q' = p$, la variante (A) de (10.7) devient

$$\begin{pmatrix} p_{n+1} \\ q_{n+1} \end{pmatrix} = \begin{pmatrix} 1 - h^2/2 & -h + h^3/4 \\ h & 1 - h^2/2 \end{pmatrix} \begin{pmatrix} p_n \\ q_n \end{pmatrix}. \tag{10.8}$$

Un calcul direct montre que $p_{n+1}^2 + (1 - h^2/4)q_{n+1}^2 = p_n^2 + (1 - h^2/4)q_n^2$. □

Méthodes de "splitting". Si l'on a besoin d'une très grande précision, l'ordre deux de la méthode "Störmer–Verlet" ne suffit pas. Une idée élégante pour obtenir des méthodes avec un ordre élevé est de considérer $(p_{n+1}, q_{n+1}) = \Phi_h(p_n, q_n)$, où Φ_h est une composition de flots (comme dans la démonstration du lemme 10.1)

$$\Phi_h = \varphi_{b_m}^{(T)} \circ \varphi_{a_m}^{(U)} \circ \varphi_{b_{m-1}}^{(T)} \circ \ldots \circ \varphi_{a_2}^{(U)} \circ \varphi_{b_1}^{(T)} \circ \varphi_{a_1}^{(U)}. \tag{10.9}$$

Pour $m = 1$ et $a_1 = b_1 = 1$ on obtient la variante (B) de la méthode d'Euler symplectique; pour $m = 2$, $a_1 = 1/2$, $b_1 = 1$, $a_2 = 1/2$, $b_2 = 0$ on obtient une des méthodes Störmer–Verlet. De cette manière on peut construire des méthodes avec un ordre arbitrairement grand et pour lesquelles les affirmations du lemme 10.3 restent vraies.

III.11 Exercices

1. Trouver une méthode numérique pour le problème $y' = f(t, y)$, $y(t_0) = y_0$ dans l'esprit de celle de Runge (voir le paragraphe III.2), mais basée sur la règle du trapèze.

2. Appliquer la méthode d'Euler, de Runge et de Heun au problème

$$y' = Ay, \quad y(0) = y_0. \tag{11.1}$$

   Montrer que la solution numérique est donnée par

$$y_n = R(hA)^n y_0,$$

   et calculer $R(hA)$ pour les trois méthodes.

3. Ecrire l'équation différentielle

$$z'' + z = 0,\ z(0) = 1,\ z'(0) = 1,$$

   sous la forme (11.1). Calculer la solution exacte et la solution numérique avec la méthode de Runge sur $[0, 1]$ avec $h = 1/2$.

4. Montrer que l'ordre d'une méthode de Runge-Kutta explicite ne peut pas être plus grand que le nombre d'étages, c.-à-d. $p \leq s$.

   Indication. Appliquer la méthode à $y' = y$, $y(0) = 1$ et observer que y_1 est un polynôme en h de degré s.

5. Donner la famille à un paramètre des méthodes de RK explicites, d'ordre $p = 2$ à $s = 2$ étages (avec comme paramètre libre c_2). Etudier le comportement de la solution numérique pour cette famille quand $c_2 \to 0$.

6. Pour le probléme de Van der Pol

$$\begin{aligned} y_1' &= y_2, & y_1(0) &= 2, \\ y_2' &= (1 - y_1^2)y_2 - y_1, & y_2(0) &= 1/2, \end{aligned}$$

   calculer le terme dominant de l'erreur locale (***i.e.*** le coefficient du terme h^{p+1}) pour la méthode de Runge d'ordre 2.

7. Calculer l'erreur locale d'une méthode de Runge-Kutta pour l'équation différentielle

$$\begin{aligned} y_1' &= x^{r-1} & y_1(0) &= 0 \\ y_2' &= x^{q-1} y_1 & y_2(0) &= 0 \end{aligned}$$

   avec r et q des entiers positifs. En déduire la condition

$$\sum_{i=1}^{s} b_i c_i^{q-1} \sum_{j=1}^{i-1} a_{ij} c_j^{r-1} = \frac{1}{r(q+r)}\,, \qquad r + q \leq p$$

   pour une méthode d'ordre p.

8. (Runge 1905). Considérons une méthode de Runge-Kutta à s étages avec l'ordre $p = s \leq 4$ et supposons que tous les coefficients a_{ij} et b_j soient non-négatifs. Montrer que la constante de Lipschitz Λ de $\Phi(x, y, h)$ (voir le lemme du paragraphe III.3) satisfait

$$(1 + h\Lambda) \leq e^{hL}$$

   où L est la constante de Lipschitz pour la fonction $f(x, y)$. En déduire que l'estimation (3.4) reste vraie si l'on remplace Λ par L.

9. Soit err_i une estimation de l'erreur au i-ème pas et définissons φ_i par

$$err_i = \varphi_i \cdot h_i^r.$$

Si on a fait le calcul jusqu'au n-ème pas, c.-à-d., si on connaît les valeurs de h_i et err_i pour $i \leq n$, il faut trouver une valeur raisonable pour h_{n+1}.

   (a) L'hypothèse $\varphi_{n+1} = \varphi_n$ nous conduit à la formule courante du cours.

   (b) (Gustafsson 1992). Montrer que l'hypothèse $\Delta \ln \varphi_n = \Delta \ln \varphi_{n-1}$ (c.-à-d., $\varphi_{n+1}/\varphi_n = \varphi_n/\varphi_{n-1}$) nous conduit à la formule

$$h_{n+1} = 0.9 \cdot \frac{h_n^2}{h_{n-1}} \Big(\frac{Tol}{err_n} \cdot \frac{err_{n-1}}{err_n} \Big)^{1/r}.$$

10. ("Dense output"). Soit $\{y_n\}$ la solution numérique obtenue par une méthode de Runge-Kutta d'ordre 4 (avec des pas constants). Pour un $x \in (x_n, x_{n+1})$, nous considérons le polynôme $u(x)$ de degré 3 qui satisfait

$$u(x_n) = y_n, \qquad u'(x_n) = f(x_n, y_n), \qquad u(x_{n+1}) = y_{n+1}, \qquad u'(x_{n+1}) = f(x_{n+1}, y_{n+1})$$

(interpolation d'Hermite, voir II.7). Montrer que, pour tout $x \in (x_n, x_{n+1})$, on a

$$u(x) - y(x) = \mathcal{O}(h^4).$$

11. Montrer que la formule de *Milne*

$$y_{n+1} = y_{n-1} + \frac{h}{3}\Big(f_{n+1} + 4f_n + f_{n-1} \Big)$$

est une méthode multipas d'ordre 4 qui est stable. Expliquer pourquoi ses coefficients sont les mêmes que pour la formule de quadrature de Simpson.

12. Calculer la solution générale de

$$y_{n+3} - 5y_{n+2} + 8y_{n+1} - 4y_n = 0,$$

puis donner une formule pour la solution particulière qui satisfait $y_0 = -1$, $y_1 = 0$, $y_2 = 4$.

13. (a) Appliquer la méthode d'Adams explicite

$$y_{n+1} = y_n + h\Big(\frac{3}{2} f_n - \frac{1}{2} f_{n-1} \Big)$$

avec $h = 1/8$ au problème $y' = y^2$, $y(0) = 1$, $y(1/2) =$?
Pour y_1 utiliser la valeur obtenue par la méthode d'Euler explicite.

   (b) Appliquer la méthode d'Euler explicite au même problème, également avec $h = 1/8$.

   (c) Comparer les deux résultats numériques avec la solution exacte $y(1/2) = 2$.

14. En utilisant le théorème binomial généralisé (Analyse I), montrer que pour les coefficients γ_j des méthodes d'Adams explicites, la *fonction génératrice* $G(t) := \sum_{j=0}^{\infty} \gamma_j t^j$ devient
$G(t) = -t/((1-t)\log(1-t))$.

En déduire la formule

$$\gamma_m + \frac{1}{2}\gamma_{m-1} + \frac{1}{3}\gamma_{m-2} + \ldots + \frac{1}{m+1}\gamma_0 = 1. \tag{1}$$

Calculer à l'aide de (1) les γ_j pour $j = 1, 2, 3, 4$.

Indication. Utiliser l'égalité $\binom{s+j-1}{j} = (-1)^j \binom{-s}{j}$.

15. Vérifier que la méthode BDF à k pas est d'ordre $p = k$.

16. Quel est le polynôme $\varrho(\zeta)$ de la méthode d'Adams explicite à k pas?

17. Montrer que le polynôme $\rho(\zeta)$ de la méthode BDF à k pas est donné par

$$\rho(\zeta) = \sum_{j=1}^{k} \frac{1}{j} \zeta^{k-j} (\zeta - 1)^j \ .$$

En utilisant la transformation $\zeta = 1/(1 - z)$, montrer que la méthode est stable si toutes les racines de

$$p(z) = \sum_{j=1}^{k} \frac{1}{j} z^j \tag{11.2}$$

sont en dehors du disque $|z - 1| \leq 1$; elle est instable si au moins une racine de (11.2) se trouve dans ce disque.

Remarque. Le polynôme (11.2) est une somme partielle de la série pour $-\log(1 - z)$.

18. En calculant numériquement les racines du polynôme (11.2), montrer que la méthode BDF est stable pour $1 \leq k \leq 6$, mais instable pour $k = 7$.

19. Une méthode multipas est dite symétrique si

$$\alpha_{k-i} = -\alpha_i, \quad \beta_{k-i} = \beta_i, \quad \text{pour } i = 0, 1, \ldots, k.$$

Démontrer que l'ordre (maximal) d'une méthode symétrique est toujours pair.

20. Soit $\varrho(\zeta)$ un polynôme quelconque de degré k satisfaisant $\varrho(1) = 0$ et $\varrho(1)' \neq 0$.

    (a) Montrer qu'il existe une unique méthode multipas implicite d'ordre $p = k+1$ dont le polynôme caractéristique est $\varrho(\zeta)$.

    (b) Montrer qu'il existe une unique méthode multipas explicite d'ordre $p = k$ dont le polynme caractéristique est $\varrho(\zeta)$.

21. Le polynme caractéristique

$$\varrho(\zeta) = \zeta^{k-2}(\zeta^2 - 1),$$

définit les méthodes de Nystrm ou de Milne-Simpson. Calculer pour $k = 2$ la méthode explicite et implicite l'aide de l'exercice précédent.