2- Méthodes à pas multiples

Il existe une autre approche de résolution des EDO qui a donné naissance à une famille de méthode dites à pas multiples. Le principe à la base de ces méthodes consiste à intégrer l'équation différentielle :

$$y^{'}(t) = f(t, y(t))$$

Dans l'intervalle $[t_n, t_{n+1}]$, ce qui donne :

$$\int_{t_n}^{t_{n+1}} y^{'}(u)du = \int_{t_n}^{t_{n+1}} f(u, y(u))du$$

Ou encore :

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(u, y(u))du$$

Cela nous ramène à un algorithme de la forme :

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f(u, y(u))du$$

On doit maintenant trouver une approximation de l'intégrale présente dans le membre de droite. On utilise une interpolation de la fonction $f(u, y(u))$ à partir des valeurs de $y(u)$ calculées aux itérations précédentes. Il est alors possible de construire une table des différences divisées pour cette fonction et d'effectuer l'interpolation par la méthode de Newton.

t_n	f_n			
		$f[t_n, t_{n-1}]$		
t_{n-1}	f_{n-1}		$f[t_n, t_{n-1}, t_{n-2}]$	
		$f[t_{n-1}, t_{n-2}]$		$f[t_n, t_{n-1}, t_{n-2}, t_{n-3}]$
t_{n-2}	f_{n-2}		$f[t_{n-1}, t_{n-2}, t_{n-3}]$	
		$f[t_{n-2}, t_{n-3}]$		
t_{n-3}	f_{n-3}			

Le polynôme d'interpolation s'écrit :

$$\begin{aligned} p_n(t) = fn + f[t_n, t_{n-1}](t - t_n) + f[t_n, t_{n-1}, t_{n-2}](t - t_n)(t - t_{n-1}) \\ + f[t_n, t_{n-1}, t_{n-2}, t_{n-3}](t - t_n)(t - t_{n-1})(t - t_{n-2}) + \cdots \end{aligned}$$

On peut évaluer la fonction $f(t, y(t))$ au moyen de ce polynôme dans l'intervalle $[t_n, t_{n+1}]$

On utilise aussi la table des différences divisées suivante :

t_{n+1}	f_{n+1}			
		$f[t_{n+1}, t_n]$		
t_n	f_n		$f[t_{n+1}, t_n, t_{n-1}]$	
		$f[t_n, t_{n-1}]$		$f[t_{n+1}, t_n, t_{n-1}, t_{n-2}]$
t_{n-1}	f_{n-1}		$f[t_n, t_{n-1}, t_{n-2}]$	
		$f[t_{n-1}, t_{n-2}]$		
t_{n-2}	f_{n-2}			

Le polynôme correspondant s'écrit :

$$p_n{}^*(t) = f_{n+1} + f[t_{n+1}, t_n](t - t_{n+1}) + f[t_{n+1}, t_n, t_{n-1}](t - t_{n+1})(t - t_n) \\ + f[t_{n+1}, t_n, t_{n-1}, t_{n-2}](t - t_{n+1})(t - t_n)(t - t_{n-1}) + \cdots$$

On constate que l'évaluation de $p_n{}^*(t)$ requit les connaissances préalable de $f_{n+1} = f(t_{n+1}, y_{n+1})$. Or, on ne connaît pas encore y_{n+1}.

Considérons d'abord le polynôme $p_n(t)$. En augmentant successivement le degré du polynôme, on obtient des approximations de plus en plus précises que l'on peut insérer dans la relation :

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f(u, y(u))du$$

Par exemple, si on utilise un polynôme de degré 0, on a l'approximation :

$$f(t, y(t)) \cong p_0(t) = f_n$$

Alors :

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f_n du = y_n + (t_{n+1} - t_n)f_n = y_n + hf(t_n, y_n)$$

C'est l'expression de la méthode d'Euler.

En utilisant maintenant un polynôme de degré 1 ; on a l'approximation :

$$f(t, y(t)) \cong p_1(t) = f_n + f[t_n, t_{n-1}](t - t_n)$$

On obtient :

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} \big(f_n + f[t_n, t_{n-1}](u - t_n)\big)du$$

$$= y_n + (t_{n+1} - t_n)f_n + \frac{(f_n - f_{n-1})}{(t_n - t_{n-1})}\frac{(t_{n+1} - t_n)^2}{2} = y_n + \frac{h}{2}(3f_n - f_{n-1})$$

Ou encore

$$y_{n+1} = y_n + \frac{h}{2}\big(3f(t_n, y_n) - f(t_{n-1}, y_{n-1})\big)$$

Il s'agit d'une méthode à *deux pas* en ce sens que pour obtenir y_{n+1} on doit utiliser y_n et y_{n-1}

Et ainsi de suite pour les polynômes de degré 2, 3, etc. On obtient les formules d'Adams-Bashforth

Formules d'Adams-Bashforth
$y_{n+1} = y_n + hf_n$ (ordre 1)
$y_{n+1} = y_n + \frac{h}{2}(3f_n - f_{n-1})$ (ordre 2)
$y_{n+1} = y_n + \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2})$ (ordre 3)
$y_{n+1} = y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$ (ordre 4)

Passons maintenant au polynôme $p_n{}^*(t)$ et d e la même manière on obtient les formules d'Adams-Moulton.

Formules d'Adams-Moulton
$y_{n+1} = y_n + hf_{n+1}$ (ordre 1)
$y_{n+1} = y_n + \frac{h}{2}(f_{n+1} + f_n)$ (ordre 2)
$y_{n+1} = y_n + \frac{h}{12}(5f_{n+1} + 8f_n - f_{n-1})$ (ordre 3)
$y_{n+1} = y_n + \frac{h}{24}(9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2})$ (ordre 4)

Les formules d'Adams-Moulton sont dites implicites en ce sens que les relations qui permettent d'évaluer y_{n+1} dépendent de y_{n+1} lui-même. On combine les formules d'Adams-Bashforth et d'Adams-Moulton en schémas dites *prédicteurs-correcteurs* dans le but de contourner cette difficulté. On utilise le schéma d'Adams- Bashforth pour obtenir une première approximation y_{n+1}^p de y_{n+1} qui est l'étape de prédiction. On fait appel ensuite aux formules d'Adams-Moulton pour corriger et améliorer cette approximation.

On obtient ainsi les schémas suivants.

Schémas de prédiction-correction
$y_{n+1}^p = y_n + hf_n$ $y_{n+1} = y_n + hf_{n+1}^p$ (ordre 1)
$y_{n+1}^p = y_n + \frac{h}{2}(3f_n - f_{n-1})$ $y_{n+1} = y_n + \frac{h}{2}(f_{n+1}^p + f_n)$ (ordre 2)
$y_{n+1}^p = y_n + \frac{h}{12}(23f_n - 16f_{n-1} + 5f_{n-2})$ $y_{n+1} = y_n + \frac{h}{12}(5f_{n+1}^p + 8f_n - f_{n-1})$ (ordre 3)
$y_{n+1}^p = y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$ $y_{n+1} = y_n + \frac{h}{24}(9f_{n+1}^p + 19f_n - 5f_{n-1} + f_{n-2})$ (ordre 4)

Exemple :

Soit l'équation différentielle :

$$y'(t) = -y(t) + t + 1 \qquad (y(0) = 1)$$

On fait appel aux méthodes de prédiction-correction d'ordre 2 et 4. Les premiers pas de temps sont calculés par la méthode de RK4

La méthode de prédiction-correction d'ordre 2 nécessite la connaissance de y_0 qui vaut 1 et y_1 qui vaut 1.0048375 (calculé par la méthode du RK2).

La première itération donne d'abord une prédiction

$$y_2^p = y_1 + \frac{h}{2}\big(3f(t_1, y_1) - 3f(t_0, y_0)\big) = 1.019111875$$

Et ensuite une correction

$$y_2 = y_1 + \frac{h}{2}\left(f\left(t_2, y_2^p\right) + f(t_1, y_1)\right) = 1.018640031$$

Les autres itérations sont résumées dans le tableau suivant.

t	y_n^p	y_n	e_n
0,0	—	1,000 000 000	0
0,1	—	1,004 837 500	$0{,}819\,640\,40 \times 10^{-7}$
0,2	1,019 111 875	1,018 640 031	$0{,}907\,218\,28 \times 10^{-4}$
0,3	1,041 085 901	1,040 653 734	$0{,}164\,486\,07 \times 10^{-3}$
0,4	1,070 487 675	1,070 096 664	$0{,}223\,381\,96 \times 10^{-3}$
0,5	1,106 614 851	1,106 261 088	$0{,}269\,571\,40 \times 10^{-3}$
0,5	1,148 826 758	1,148 506 695	$0{,}304\,940\,11 \times 10^{-3}$
0,7	1,196 543 746	1,196 254 173	$0{,}331\,129\,90 \times 10^{-3}$
0,8	1,249 241 382	1,248 979 396	$0{,}361\,493\,25 \times 10^{-3}$
0,9	1,306 445 195	1,306 208 166	$0{,}367\,978\,57 \times 10^{-3}$
1,0	1,367 725 911	1,367 511 462	$0{,}367\,978\,57 \times 10^{-3}$

De manière similaire la méthode de prédiction-correction d'ordre 4 nécessite le calcul de y_1, y_2 et de y_3 à l'aide de la méthode de RK4. Par suite l'utilisation de l'algorithme suivant.

$$y_{n+1}^p = y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$$

$$y_{n+1} = y_n + \frac{h}{24}\left(9f_{n+1}^p + 19f_n - 5f_{n-1} + f_{n-2}\right)$$

On obtient les résultats suivants.

t	y_n^p	y_n	e_n
0,0	—	1,000 0000	0
0,1	—	1,004 8375	$0{,}819\,640 \times 10^{-7}$
0,2	—	1,018 7309	$0{,}148\,328 \times 10^{-6}$
0,3	—	1,040 8184	$0{,}201\,319 \times 10^{-6}$
0,4	1,107 0323	1,070 3199	$0{,}127\,791 \times 10^{-6}$
0,5	1,106 5332	1,106 5303	$0{,}391\,302 \times 10^{-6}$
0,6	1,148 8136	1,148 8110	$0{,}603\,539 \times 10^{-6}$
0,7	1,196 5869	1,196 5845	$0{,}772\,415 \times 10^{-6}$
0,8	1,249 3302	1,249 3281	$0{,}903\,669 \times 10^{-6}$
0,9	1,306 5706	1,306 5687	$0{,}100\,294 \times 10^{-5}$
1,0	1,367 8801	1,367 8784	$0{,}107\,514 \times 10^{-5}$

L’erreur est beaucoup plus faible qu’avec la méthode d’ordre 2.