



$$x^{(k+1)} = M^{-1}Nx^k + M^{-1}b \quad (1.2)$$

Avec  $x^{(0)} \in \mathbb{K}^n$  (vecteur de départ). L'algorithme est initialisé par un vecteur arbitraire  $x^{(0)} = (x_1^0, x_2^0, x_3^0, \dots, x_n^0)$

Pour un  $\varepsilon$  donné, il s'arrête quand:

$$\forall i \in \mathbb{N}, \left| x_i^{(k+1)} - x_i^{(k)} \right| < \varepsilon \quad (1.3)$$

Selon les choix des matrices  $M$  et  $N$  on a différentes méthodes itératives. Le point de départ de chacune de ces méthodes est l'unique **décomposition** de la matrice  $A = (a_{i,j})_{1 \leq i,j \leq n}$  sous la forme  $A = D - (E + F)$  avec :

- $D = (d_{i,j})_{1 \leq i,j \leq n}$  diagonale, telle que  $d_{i,j} = a_{i,j}$  et  $d_{i,j} = 0$  pour  $i \neq j$  ;
- $-E = (e_{i,j})_{1 \leq i,j \leq n}$  triangulaire inférieure **stricte** telle que  $e_{i,j} = a_{i,j}$  si  $i > j$  et  $e_{i,j} = 0$  si  $i \leq j$  ;
- $-F = (f_{i,j})_{1 \leq i,j \leq n}$  triangulaire supérieure **stricte**, telle que  $f_{i,j} = a_{i,j}$  si  $i < j$  et  $f_{i,j} = 0$  si  $i \geq j$  ;

### Exemple

Considérons la matrice

$$A = \begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}$$

La décomposition de  $A$  sous la forme  $A = D - (E + F)$  décrite ci-dessus s'écrit alors :

$$A = \begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix} = \underbrace{\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}}_D - \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \\ -1 & -1 & 0 \end{pmatrix}}_{-E} - \underbrace{\begin{pmatrix} 0 & -1 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix}}_{-F}$$

### 1.2.1 Méthode de Jacobi

On suppose que  $A$  est une matrice inversible dont aucun élément de la diagonale est nul ( $a_{ii} \neq 0, \forall i$ ). Cette méthode consiste à isoler le coefficient de la diagonale de chaque ligne du système, si l'un des coefficients diagonaux est nul, il est parfois possible de permuter certaines lignes pour éviter cette situation.

**On considère** un système linéaire suivant :

$$Ax = b, \text{ avec } A \text{ inversible.}$$

On pose :

$$A = M - N, \text{ avec } M = D \text{ et } N = (E + F); \text{ sachant que } D \text{ est inversible.}$$

Le schéma itératif s'écrit alors :

$$x^{(k+1)} = D^{-1}(E + F)x^k + D^{-1}b \quad (1.4)$$

avec  $x^{(0)} \in \mathbb{K}^n$

La matrice  $B_J = D^{-1}(E + F)$  est appelée **matrice de Jacobi**.

**Algorithme de Jacobi**

$$\left\{ \begin{array}{l} x^{(0)} \text{ donné} \\ x_i^{(k+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j \neq i, j=1}^n a_{ij} x_j^{(k)} \right) \end{array} \right. \quad \forall i = 1, \dots, n.$$

Cet algorithme nécessite  $a_{ii} \neq 0$  pour  $i = 1, \dots, n$

Explicitement, on obtient :

$$\begin{aligned} a_{11}x_1^{(k+1)} &= -a_{12}x_2^{(k)} - a_{13}x_3^{(k)} \dots - a_{1n}x_n^{(k)} + b_1 \\ a_{22}x_2^{(k+1)} &= -a_{21}x_1^{(k)} - a_{23}x_3^{(k)} \dots - a_{2n}x_n^{(k)} + b_2 \\ &\vdots \\ a_{nn}x_n^{(k+1)} &= -a_{n1}x_1^{(k)} - \dots - a_{nn-1}x_{n-1}^{(k)} + b_n \end{aligned}$$

**Test de convergence**

La méthode ne converge pas toujours. On démontre que si  $A$  est une matrice définie positive, la méthode itérative converge. De même, si  $A$  est une matrice diagonalement dominante, c'est-à-dire si

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}| \quad (1.5)$$

alors la méthode de Jacobi converge. Par conséquent, on peut avoir intérêt à réarranger les termes de  $A$  de façon à mettre  $A$  sous la forme d'une matrice dont les éléments diagonaux sont les plus grands possibles.

**1.2.2 Méthode de Gauss-Seidel**

Comme pour la méthode de Jacobi, on suppose cette fois que la matrice  $D$  est une matrice inversible dont aucun élément de la diagonale est nul ( $a_{ii} \neq 0, \forall i$ ). Cette méthode consiste à isoler le coefficient de la diagonale de chaque ligne du système, si l'un des coefficients diagonaux est nul, comme pour la méthode de Jacobi, il est parfois possible de permuter certaines lignes pour éviter cette situation.

On considère un système linéaire suivant :

$$Ax = b, \text{ avec } A \text{ inversible.}$$

On pose :

$$A = M - N, \text{ avec } M = D - E \text{ et } N = F; \text{ avec } (D - E) \text{ inversible}$$

Le schéma itératif s'écrit alors :

$$x^{(k+1)} = (D - E)^{-1} F x^{(k)} + (D - E)^{-1} b \quad (1.6)$$

avec  $x^{(0)} \in \mathbb{K}^n$  (vecteur de départ).

La matrice  $B_{GS} = (D - E)^{-1} F$  est appelée **matrice de Gauss-Seidel**. A chaque pas, on calcule

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j=1}^i a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right) \quad (1.7)$$

**Algorithme de Gauss-Seidel**

$$\left\{ \begin{array}{l} x^{(0)} \text{ donné} \\ x_i^{(k+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j<i} a_{ij}x_j^{(k+1)} - \sum_{j>i} a_{ij}x_j^{(k)} \right) \quad \forall i = 1, \dots, n \end{array} \right.$$

Cet algorithme nécessite  $a_{ii} \neq 0$  pour  $i = 1, \dots, n$   
 Explicitement, on obtient :

$$\begin{aligned} a_{11}x_1^{(k+1)} &= -a_{12}x_2^{(k)} - a_{13}x_3^{(k)} \dots - a_{1n}x_n^{(k)} + b_1 \\ a_{22}x_2^{(k+1)} &= -a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} \dots - a_{2n}x_n^{(k)} + b_2 \\ &\vdots \\ a_{ii}x_i^{(k+1)} &= -a_{i1}x_1^{(k+1)} - a_{ii-1}x_{i-1}^{(k+1)} - a_{ii+1}x_{i+1}^{(k)} \dots - a_{in}x_n^{(k)} + b_n \\ &\vdots \\ a_{nn}x_n^{(k+1)} &= -a_{n1}x_1^{(k+1)} - \dots - a_{nn-1}x_{n-1}^{(k+1)} + b_n \end{aligned}$$

**1.3 Résolution des systèmes d'équations non linéaires**

**1.3.1 Méthode de Newton-Raphson**

La **méthode de Newton(ou Newton-Raphson)** consiste à prendre pour  $x_{k+1}$  la racine du développement de Taylor de premier ordre autour de  $x^{(k)}$  ; cela revient, pour autant que  $f'(x^{(k)}) \neq 0$ , à déterminer  $x^{(k+1)}$  qui satisfait :

$$f(x^{(k)}) + f'(x^{(k)})(x^{(k+1)} - x^{(k)}) = 0$$

Et donc

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})} \tag{1.8}$$

**Critère de convergence**

Soit une fonction  $f$  définie sur  $[a, b]$  telle que :

- i.  $f(a)f(b) < 0$
- ii.  $f'(x)$  et  $f''(x)$  sont non nulles et gardent un signe constant sur l'intervalle donné.

**Algorithme de Newton**

(entrée :  $f, f', x^{(0)}$ , Sortie :  $x^{(k+1)}$ )  
 Répéter jusqu'à l'arrêt  
 Pour autant que  $f'(x^{(k)}) \neq 0$

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}$$

**Critère d'arrêt**

En général, pour toutes les méthodes étudiées, on peut utiliser deux critères d'arrêt différents : les itérations s'achèvent dès que :

$$|x_i^{(k+1)} - x_i^{(k)}| < \varepsilon \quad (\text{Contrôle de l'incrément})$$

Si non

$$|f(x^{(k)})| < \varepsilon \quad (\text{Contrôle de résidu})$$

où  $\varepsilon$  est une tolérance fixée. On a vu que pour la méthode de Newton le premier critère (contrôle de l'incrément) est optimal au sens qu'il garantit que l'erreur finale soit plus petite que la tolérance fixée.

Par contre, le deuxième critère (contrôle du résidu) est satisfaisant lorsque  $|f'| \simeq 1$  dans un voisinage de la racine  $\alpha$  (un zéro pour la fonction  $f$ ). Sinon il est soit trop restrictif (si  $|f'| \gg 1$ ) soit trop faible (si  $|f'| \ll 1$ ).

**Remarque**

Dans le cas d'un système de deux équations non linéaires à 2 inconnues, par exemple :

$$\begin{cases} f_1(x_1, x_2) = 0 \\ f_2(x_1, x_2) = 0 \end{cases} \quad (1.9)$$

L'algorithme de **Newton-Raphson** devient :

$$\begin{pmatrix} x_1^{k+1} \\ x_2^{k+1} \end{pmatrix} = \begin{pmatrix} x_1^k \\ x_2^k \end{pmatrix} - \begin{pmatrix} \frac{\partial f_1(x_1, x_2)}{\partial x_1} & \frac{\partial f_1(x_1, x_2)}{\partial x_2} \\ \frac{\partial f_2(x_1, x_2)}{\partial x_1} & \frac{\partial f_2(x_1, x_2)}{\partial x_2} \end{pmatrix}^{-1} \begin{pmatrix} f_1(x_1^k, x_2^k) \\ f_2(x_1^k, x_2^k) \end{pmatrix} \quad (1.10)$$

**1.4 Equation Différentielles Ordinaires (EDO)**

De nombreux modèles s'expriment au moyen d'équations différentielles ordinaires (en abrégé EDO). L'adjectif ordinaire est surtout là pour faire la distinction entre les EDO et les EDP (équations aux dérivées partielles). Si toutes les dérivées sont prises par rapport à une seule variable, on parle d'équation différentielle ordinaire (EDO). Une équation mettant en jeu des dérivées partielles est appelée équation aux dérivées partielles (EDP).

Une équation différentielle ordinaire, également notée EDO, d'ordre  $n$  est une relation entre la variable réelle  $t \in I, I$  intervalle réel une fonction inconnu et  $t \rightarrow x(t)$  et ses dérivées par rapport à  $t$  définie par :

$$F(t, x, x'', \dots, x^{(n)}) = 0 \quad (1.11)$$

Une fonction  $x$  qui vérifie  $F(t, x, x'', \dots, x^{(n)}) = 0$  s'appelle solution de l'EDO.

**Remarque** : Une EDO est d'ordre  $n$  si elle contient les dérivées de  $x$  jusqu'à l'ordre  $n$ .

**Exemple1**

Les équations suivantes :

$$y'(t) - t = 0$$

$$y''(t) - y(t) = 0$$

Sont équation différentielles ordinaires.

### Exemple 2

L'EDO d'ordre 2 la plus célèbre est la deuxième *loi de Newton* :

$$m \frac{d^2x}{dt^2} = F(x)$$

qui décrit par exemple la dynamique d'un point matérielle soumis à la résultante des forces  $F$ .

On peut réécrire la loi de Newton sous la forme d'un système d'équations différentielles, en posant

$$v = \frac{dx}{dt}$$

On obtient alors le système du premier ordre autonome :

$$\begin{cases} v = \frac{dx}{dt} \\ \frac{dv}{dt} = \frac{1}{m} F(x) \end{cases}$$

## 1.4.1 Méthode de résolution des EDO

### a) Méthode d'Euler

La **méthode d'Euler** est une procédure numérique qui permet de résoudre de façon approximative des équations différentielles ordinaires du premier ordre avec condition initiale. Elle a le mérite d'être simple à comprendre et à programmer.

On cherche donc une solution approchée d'une équation ordinaire se mettant sous la forme

$$\begin{cases} \frac{dy}{dt} = f(t, y) & 0 \leq t \leq T \\ t(0) = t_0, y(0) = y_0 \end{cases} \quad (1.12)$$

Avec  $T$ : une durée bien déterminée dans le temps.

On peut intégrer cette équation comme suit :

$$y(t_{k+1}) = y(t_k) + \int_{t_k}^{t_{k+1}} f(t, y(t)) dt \quad (1.13)$$

La méthode d'Euler consiste à approcher l'intégrale par la méthode des rectangles à gauche :

$$\int_{t_k}^{t_{k+1}} f(t, y(t)) dt \cong h \times f(t_k, y(t_k))$$

D'où le schéma itératif suivant

$$y_{k+1} = y_k + \underbrace{(t_{k+1} - t_k)}_h f(t_k, y_k)$$

$h$  : Taille de pas

Donc

$$\begin{cases} y_{k+1} = y_k + hf(t_k, y_k) \\ y_0 = y(0), \quad k = 0, 1, \dots, n \end{cases} \quad (1.14)$$

où  $y_k$  désigne l'approximation numérique de  $y(t_k)$ . La mise en œuvre est alors extrêmement simple :

**Algorithme d'Euler**

1. Initialisation du pas  $h$  et de la durée  $T$ .
2. Initialisation des conditions initiales :  $t = 0$  et  $y = y(0)$ .
3. Tant que  $t \leq T$  faire :
  - (a) Calcul de  $k_1 = f(t, y)$ .
  - (b)  $y = y + hk_1$ ;  $t = t + h$ .
  - (c) Enregistrement des données.

**Exemple**

Pour le problème de valeur initiale suivant:

$$\begin{cases} y' + 2y = 2 - e^{-4t} \\ y(0) = 1 \end{cases}$$

Utilisez la méthode d'Euler avec un pas de  $h = 0,1$  pour trouver les valeurs approximatives de la solution à  $t = 0.1, 0.2, 0.3, 0.4$  et  $0.5$ . Comparez-les aux valeurs exactes de la solution à ces points.

Il s'agit d'une équation différentielle linéaire assez simple, nous vous laisserons donc le soin de vérifier que la solution correcte est :

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

De cela nous pouvons constater que :

$$f(t, y) = y' = 2 - e^{-4t} - 2y$$

Notez également que  $t_0 = 0$  et  $y_0 = y(0) = 1$ .

Nous pouvons maintenant commencer à faire des calculs.

On a :

$$y_{k+1} = y_k + h \times f(t_k, y_k)$$

Pour  $k = 0$

$$y_1 = y_0 + h \times f(t_0, y_0) \Rightarrow y_1 = 1 + 0.1f(0,1)$$

$$f_0 = f(0,1) = 2 - e^{-4(0)} - 2(1) = -1$$

$$\Rightarrow y_1 = 0.9$$

Donc, l'approximation de la solution à  $t_1 = 0.1$  est  $y_1 = 0.9$ .

Pour  $k = 1$

$$y_2 = y_1 + h \times f(t_1, y_1) \Rightarrow y_2 = 1 + 0.1f(0.1,0.9)$$

$$f_1 = f(0.1,0.9) = 2 - e^{-4(0.1)} - 2(0.9) = -0.4703$$

$$\Rightarrow y_2 = 0.9 + (0.1)(-0.4703) = 0.85297$$

Par conséquent, l'approximation de la solution à  $t_2 = 0.2$  est  $y_2 = 0.85297$ .

Je vous laisse le soin de vérifier le reste de ces calculs.

$$f_2 = -0.1553$$

$$y_3 = 0.8374$$

$$f_3 = 0.02392 \qquad y_4 = 0.8398$$

$$f_4 = 0.11843 \qquad y_5 = 0.8517$$

Voici un tableau (1.1) rapide qui donne les approximations ainsi que la valeur exacte des solutions aux points donnés. Nous avons également inclus l'erreur en pourcentage. Il est souvent plus facile de voir à quel point une approximation est efficace si vous examinez les pourcentages. La formule pour cela est,

$$\text{Erreur (\%)} = \frac{|\text{valeur exacte} - \text{valeur approximée}|}{\text{valeur exacte}} \times 100$$

**Tableau 1.1** Approximations et valeur exacte des solutions aux points donnés

Temps $t_k$	Approximation	Exacte	Erreur
$t_0 = 0$	$y_0 = 1$	$y(0) = 1$	0%
$t_1 = 0.1$	$y_1 = 0.9$	$y(0.1) = 0.9258$	2.79%
$t_2 = 0.2$	$y_2 = 0.8529$	$y(0.2) = 0.8895$	4.11%
$t_3 = 0.3$	$y_3 = 0.8374$	$y(0.3) = 0.8762$	<b>4.42%</b>
$t_4 = 0.4$	$y_4 = 0.8398$	$y(0.4) = 0.8763$	4.16%
$t_5 = 0.5$	$y_5 = 0.8517$	$y(0.5) = 0.8837$	3.63%

D'après le tableau (1.1), l'erreur maximale dans les approximations du dernier exemple était de 4,42%, ce qui n'est pas si mal, mais qui n'est pas non plus une si grande approximation. Ce type d'erreur est cependant généralement inacceptable dans la plupart des applications réelles. Alors, comment pouvons-nous obtenir de meilleures approximations?

Rappelons que nous obtenons les approximations en utilisant une ligne tangente pour approximer la valeur de la solution et que nous avançons dans le temps par pas de  $h$ . Donc, si nous voulons une approximation plus précise, il semble alors qu'il existe une seule façon d'obtenir une meilleure approximation c'est de ne pas avancer autant à chaque étape. En d'autres termes, il faudrait prendre des  $h$  plus petits.

### Remarque

- La diminution de la taille de pas  $h$  améliore la précision de l'approximation.
- La diminution d'un facteur 10 la taille du pas  $h$ , réduit l'erreur d'un facteur de 10 environ.

### b) Méthode de Runge-Kutta

Les techniques de **Runge-Kutta** sont des schémas numériques à un pas qui permettent de résoudre les équations différentielles ordinaires. Elles font parties des méthodes les plus populaires de par leur facilité de mise en œuvre et leur précision. C'est *Carle Runge* et *Martin Kutta* qui, au début du 20e siècle, ont inventé ces méthodes.

Dans de nombreux cas, les systèmes d'équations différentielles que l'on rencontre en science peuvent se mettre sous la forme d'une équation différentielle ordinaire du premier ordre (Eq. 1.8).

À l'instar de la **méthode d'Euler**, celles de **Runge-Kutta** sont des **schémas numériques** à un pas basée sur la discrétisation de la variable  $t$ . On note  $h$  ce pas et  $y_k$  la valeur approchée de  $y(t_k)$  pour les différents instants



$$t_k = kh. \tag{1.15}$$

En intégrant l'équation différentielle entre  $t_k$  et  $t_k + h$  on a la relation :

$$y(t_{k+1}) = y(t_k) + \int_{t_k}^{t_k+h} f(t, y(t)) dt \tag{1.16}$$

L'idée consiste à approcher cette intégrale de façon plus précise que ne le fait la méthode d'Euler.

Le schéma itératif de **Runge-Kutta d'ordre 2** est (RK2):

$$\begin{cases} y_{k+1} = y_k + \frac{h}{2}(k_1 + k_2), \\ t_{k+1} = t_k + h; \quad k = 0, \dots, N - 1 \end{cases} \quad \text{avec} \quad \begin{cases} k_1 = f(t_k, y_k) \\ k_2 = f(t_k + h, y_k + hk_1) \end{cases}$$

La méthode classique **de Runge-Kutta** est une approximation d'ordre 4. On cherche donc une solution approchée d'une équation ordinaire se mettant sous la forme

$$\begin{cases} \frac{dy}{dt} = f(t, y) & 0 \leq t \leq T \\ t(0) = t_0, \quad y(0) = y_0 \end{cases}$$

Le schéma itératif de Runge-Kutta d'ordre 4 est (RK4):

$$\begin{cases} y_{k+1} = y_k + \frac{h}{6}[k_1 + 2k_2 + 2k_3 + k_4] \\ t_{k+1} = t_k + h, \quad k = 0, \dots, N - 1 \end{cases} \quad \text{avec} \quad \begin{cases} k_1 = f(t_k, y_k) \\ k_2 = f(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_1) \\ k_3 = f(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_2) \\ k_4 = f(t_k + h, y_k + hk_3) \end{cases}$$

### c) Méthode de d'Adams-Bashforth

La méthode **d'Adams- Bashforth** est une méthode à pas multiples qui repose sur une approximation de l'intégrale suivant :

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

Les schémas itératifs de la méthode d'Adams-Bashforth:

On note en abrégé  $f_k = f(t_k, y_k)$ . Voici trois schémas :

- schéma explicite d'Adams-Bashforth d'ordre 2 à 2 pas (**AB2**):

$$y_{k+1} = y_k + \frac{h}{2}(3f_k - f_{k-1}) \tag{1.17}$$

- schéma explicite d'Adams-Bashforth d'ordre 3 à 3 pas(**AB3**) :

$$y_{k+1} = y_k + \frac{h}{12}(23f_k - 16f_{k-1} + 5f_{k-2}) \tag{1.18}$$

- schéma explicite d'Adams-Bashforth d'ordre 4 à 4 pas (**AB4**):

$$y_{k+1} = y_k + \frac{h}{24} (55f_k - 59f_{k-1} + 37f_{k-2} - 9f_{k-3}) \quad (1.19)$$

#### d) Méthode d'Adams-Moulton (AM)

Cette méthode est l'une des méthodes fermées, son algorithme est :

$$y_{k+1} = y_k + \frac{h}{2} (f(t_k, y_k) + f(t_{k+1}, y_{k+1}^{(i)})) \quad (1.20)$$

Avec  $y_{k+1}^{(i)}$  est donnée pour  $i = 0, 1, 2, \dots$

$y_{k+1}^{(0)}$  est dite le **prédicteur** de  $y_{k+1}$ , les  $y_{k+1}^{(i)}$  sont appelés **les correcteurs**, pour les calculés, on utilise la **méthode d'Euler** comme **prédicteur** et la **méthode d'Adams-Moulton** comme **correcteur**. On obtient :

$$\begin{cases} y_{k+1}^{(0)} = y_k + hf(t_k, y_k) & \text{Prédiction} \\ y_{k+1} = y_k + \frac{h}{2} (f(t_k, y_k) + f(t_{k+1}, y_{k+1}^{(i)})) & \text{Correction} \end{cases} \quad (1.21)$$

### 1.5 Intégration numérique

L'intégration numérique est le calcul par des méthodes numériques l'intégrale :

$$I(f) = \int_{x_0}^{x_n} f(x) dx \quad (1.22)$$

où  $f(x)$  est une fonction connue seulement en quelques points ou encore une fonction n'ayant pas de primitive.

#### 1.5.1 Méthodes des trapèzes

On considère une subdivision de  $[a, b]$  en sous intervalles égaux  $[x_i, x_{i+1}]$  de bornes  $x_i = a + ih$ , ( $i = 0, \dots, n$ ),  $a = x_0$ ,  $b = x_n$  et  $h = \frac{b-a}{n}$ . On suppose connues les valeurs  $f(x_i)$  pour  $i = 0, \dots, n$ . Sur chaque intervalle  $[x_i, x_{i+1}]$  on remplace la fonction  $f(x)$  par la droite :

$$y = f(x_i) + (x - x_i) \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} \quad (1.23)$$

L'aire exacte est donc remplacée par l'aire  $T_i$  du trapèze. Par conséquent en sommant les aires des  $n$  trapèzes de base  $|x_i, x_{i+1}|$ , ( $i = 0, \dots, n$ ), on obtient une approximation suivante :

$$\int_a^b f(x) dx = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x) dx \approx \sum_{i=0}^{n-1} \frac{h}{2} [f(x_i) + f(x_{i+1})] \quad (1.24)$$

Après un calcul on déduit la formule de la méthode de Trapèzes suivante :

$$I_{\text{trapèze}} = \int_a^b f(x) dx \approx \frac{h}{2} \left( f(x_0) + f(x_n) + 2 \sum_{i=1}^{n-1} f(x_i) \right) \quad (1.25)$$

**Remarque**

Plus le nombre  $n$  de sous-intervalles est grand, plus la précision sur la valeur approchée de l'intégrale est grande.

L'erreur commise à la méthode de trapèzes est majorée :

$$|I_{\text{trapèze}}(f) - I_{\text{exacte}}(f)| = \left| \frac{(b-a)^3}{12\eta^2} f''(\eta) \right| \leq \frac{(b-a)^3}{12\eta^2} M_2 \quad (1.26)$$

$$M_2 = \max_{[a,b]} |f''(x)| \quad (1.27)$$

Etant donnée la précision  $\varepsilon$  on peut alors déterminer le nombre minimal  $n$  des sous-intervalles par :

$$n \geq \sqrt{\frac{(b-a)^3}{12\varepsilon} M_2} \quad (1.28)$$

**1.5.2 Méthode de Simpson**

Dans cette méthode on suppose que  **$n$  est pair** (soit  $n = 2s$ ), puis on subdivise  $[a, b]$  en  $s$  sous intervalles égaux  $|x_{i-1}, x_{i+1}|$  de longueur  $h$ , ( $i = 1 \dots s - 1$ ), puis on remplace  $f(x)$  sur chaque intervalle  $|x_{i-1}, x_{i+1}|$  non pas par une droite comme dans les trapèzes, mais par une parabole ayant aux abscisses  $x_{i-1}, x_i$  et  $x_{i+1}$  les mêmes valeurs que  $f$ ; c'est à dire par une interpolation quadratique sur ces trois points. Le polynôme  $P_1$  d'interpolation en  $x_{i-1}, x_i$  et  $x_{i+1}$  est :

$$\begin{aligned} P_1(x) &= f(x_{i-1}) \frac{(x - x_{i+1})(x - x_i)}{2h^2} + \\ &= f(x_i) \frac{(x - x_{i-1})(x - x_{i+1})}{-h^2} + \\ &= f(x_{i+1}) \frac{(x - x_i)(x - x_{i-1})}{2h^2} \end{aligned}$$

D'autre part la surface  $S_i$  délimitée par cette parabole, les droites  $x = x_{i-1}, x = x_{i+1}$  et l'axe des abscisses s'obtient en calculant :  $\int_{x_{i-1}}^{x_{i+1}} P_1(x) dx$  d'où ;

$$S_i = \frac{h}{3} (f(x_{i-1}) + 4f(x_i) + f(x_{i+1}))$$

Ceci donne l'approximation de Simpson  $I_{\text{Simpson}}$  de  $I$ , en sommant les aires  $I_{\text{Simpson}}$  pour  $i = 1$  à  $n - 1$ . Finalement :

$$I_{\text{Simpson}} = \frac{h}{3} [[f(x_0) + f(x_{2s}) + 4[f(x_1) + f(x_3) + \dots + f(x_{2s-1})] + 2[f(x_2) + f(x_4) + \dots + f(x_{2s-2})]] \quad (1.29)$$

C'est à dire :

$$I_{\text{Simpson}} = \frac{h}{3} \left[ f(x_0) + f(x_{2s}) + 4 \sum_{k=1}^s f(x_{2k-1}) + 2 \sum_{k=1}^{s-1} f(x_{2k}) \right]$$

L'erreur commise à la méthode de Simpson est majorée :

$$|I_{\text{Simpson}}(f) - I_{\text{exacte}}(f)| = \left| -\frac{h^5(b-a)^3}{180\eta^4} f^{(4)}(\eta) \right| \leq \frac{(b-a)^5}{180\eta^4} M_4 \quad (1.26)$$

$$M_2 = \max_{[a,b]} |f^{(4)}(x)| \quad (1.27)$$

Etant donnée la précision  $\varepsilon$  on peut alors déterminer le nombre minimal  $n$  des sous-intervalles par :

$$n \geq \sqrt[4]{\frac{(b-a)^3}{180\varepsilon} M_4} \quad (1.28)$$

### Remarque

En général la méthode de Simpson donne une meilleure approximation que celle des trapèzes, (sous certaines conditions de régularité de  $f$ ).

## Exercices corrigés Chapitre 1

**Exercice 1 :**

Résoudre le système suivant :

$$\begin{cases} 4x_1 + 2x_2 + x_3 = 4 \\ -x_1 + 2x_2 + 0x_3 = 2 \\ 2x_1 + x_2 + 4x_3 = 9 \end{cases}$$

avec  $x^{(0)} = (0,0,0)^T$  (vecteur de départ) et le critère d'arrêt  $\varepsilon = 10^{-5}$ .

à l'aide de la **méthode de Gauss-Seidel**. Effectuer seulement les trois (3) premières itérations, en partant de point  $x^{(0)} = (0,0,0)^T$  (vecteur de départ).

Déduire la solution exacte ?

**Exercice 2 :**

On considère l'équation différentielle :

$$\begin{cases} \frac{dy}{dt} = f(t, y) = y(t) + t \\ y(0) = 1 \end{cases}$$

a) Vérifier que la solution analytique est  $y(t) = -1 - t + 2e^t$

b) Montrer que la méthode d'Euler s'écrit  $y_{k+1} = y_k(1 + h) + ht_k$

Calculer de la solution au point 0.3.

**Exercice 3 :**

On considère des équations différentielles suivantes :

$$(a) \frac{dy}{dt} = t \sin(y(t)) \quad (y(0) = 2)$$

$$(b) \frac{dy}{dt} = e^t y(t) \quad (y(0) = 2)$$

Utilisez la **méthode d'Euler**, puis la **méthode de Runge-Kutta** d'ordre 4, avec un pas de  $h = 0,1$  pour trouver les valeurs approximatives de la solution à  $t = 0,1, 0,2, 0,3$ .

**Exercice 4 :**

Soit l'équation  $f$  suivante

$$f(x) = \ln(x) - x + 2 = 0$$

Approcher de la racine à  $\varepsilon = 10^{-4}$  près par la **méthode de Newton** avec  $x_0 = 3$ .

**Exercice 5 :**

Déterminer par la méthode des trapèzes puis par celle de Simpson  $\int_0^{\frac{\pi}{2}} f(x) dx$  sur la base du tableau suivant :

$x$	0	$\pi/8$	$\pi/4$	$3\pi/8$	$\pi/2$
$f(x)$	0	0.382683	0.707107	0.923880	1

Ces points d'appui sont ceux donnant  $\sin x$ , comparer alors les résultats obtenus avec la valeur exacte.

**Exercice 6 :**

On lance une fusée verticalement du sol et l'on mesure pendant les premières 80 secondes l'accélération  $\gamma$  :

$t(s)$	0	10	20	30	40	50	60	70	80
$\gamma(m/s^2)$	30	31.63	33.44	35.47	37.75	40.33	43.29	46.70	50.67

Calculer la vitesse  $V$  de la fusée à l'instant  $t = 80$  s, par les Trapèzes puis par Simpson.

**Solutions des Exercices de Chapitre 1**

**Solution Exercice 1 :**

Résolution à l'aide de la **méthode de Gauss-Seidel** du système suivant :

$$\begin{cases} 4x_1 + 2x_2 + x_3 = 4 \\ -x_1 + 2x_2 + 0x_3 = 2 \\ 2x_1 + x_2 + 4x_3 = 9 \end{cases} ; \quad \varepsilon = 10^{-5} \text{ et } x^{(0)} = (0,0,0)^T$$

On a :

$$\begin{cases} x_1^{(k+1)} = \frac{1}{4}(4 - 2x_2^{(k)} - x_3^{(k)}) \\ x_2^{(k+1)} = \frac{1}{2}(2 + x_1^{(k+1)}) \\ x_3^{(k+1)} = \frac{1}{4}(9 - 2x_1^{(k+1)} - x_2^{(k+1)}) \end{cases}$$

**Itération (1) (pour  $k = 0$ )**

En partant de  $x^{(0)} = (0,0,0)^T$

$$\begin{cases} x_1^{(1)} = \frac{1}{4}(4 - 2x_2^{(0)} - x_3^{(0)}) \\ x_2^{(1)} = \frac{1}{2}(2 + x_1^{(1)}) \\ x_3^{(1)} = \frac{1}{4}(9 - 2x_1^{(1)} - x_2^{(1)}) \end{cases} \Rightarrow \begin{cases} x_1^{(1)} = \frac{1}{4}(4 - 2(0) - (0)) = 1 \\ x_2^{(1)} = \frac{1}{2}(2 + 1) = 3/2 \\ x_3^{(1)} = \frac{1}{4}(9 - 2(1) - 3/2) = 11/8 \end{cases}$$

$$x^{(1)} = \left(1, \frac{3}{2}, \frac{11}{8}\right)^T \cong (1, 1.5000, 1.3750)$$

**Itération (2) (pour  $k = 1$ )**

En partant de  $x^{(1)} = \left(1, \frac{3}{2}, \frac{11}{8}\right)^T$

$$\begin{cases} x_1^{(2)} = \frac{1}{4}(4 - 2x_2^{(1)} - x_3^{(1)}) \\ x_2^{(2)} = \frac{1}{2}(2 + x_1^{(2)}) \\ x_3^{(2)} = \frac{1}{4}(9 - 2x_1^{(2)} - x_2^{(2)}) \end{cases} \Rightarrow \begin{cases} x_1^{(2)} = \frac{1}{4}\left(4 - 2\left(\frac{3}{2}\right) - \left(\frac{11}{8}\right)\right) = \frac{-3}{32} \\ x_2^{(2)} = \frac{1}{2}\left(2 + \frac{-3}{32}\right) = \frac{61}{64} \\ x_3^{(2)} = \frac{1}{4}\left(9 - 2\left(\frac{-3}{32}\right) - \frac{61}{64}\right) = \frac{527}{256} \end{cases}$$

$$x^{(2)} = \left(\frac{-3}{32}, \frac{61}{64}, \frac{527}{256}\right)^T \cong (-0.0937, 0.9521, 2.0585)$$

**Itération (3) (pour  $k = 2$ )**

En partant de  $x^{(2)} = \left(\frac{-3}{32}, \frac{61}{64}, \frac{527}{256}\right)^T$

$$\begin{cases} x_1^{(3)} = \frac{1}{4}(4 - 2x_2^{(2)} - x_3^{(2)}) \\ x_2^{(3)} = \frac{1}{2}(2 + x_1^{(3)}) \\ x_3^{(3)} = \frac{1}{4}(9 - 2x_1^{(3)} - x_2^{(3)}) \end{cases} \Rightarrow \begin{cases} x_1^{(3)} = \frac{1}{4}\left(4 - 2\left(\frac{61}{64}\right) - \left(\frac{527}{256}\right)\right) = \frac{9}{1024} \\ x_2^{(3)} = \frac{1}{2}\left(2 + \frac{9}{1024}\right) = \frac{2057}{2048} \\ x_3^{(3)} = \frac{1}{4}\left(9 - 2\left(\frac{9}{1024}\right) - \left(\frac{2057}{2048}\right)\right) = \frac{16339}{8192} \end{cases}$$

$$x^{(3)} = \left(\frac{9}{1024}, \frac{2057}{2048}, \frac{16339}{8192}\right)^T \cong (0.0087, 1.0043, 1.9945)$$

La suite  $x^{(3)}$  converge vers la solution du système  $x = (0, 1, 2)$ .

## Solution exercice 2

1- On considère l'équation différentielle :

$$\begin{cases} \frac{dy}{dt} = f(t, y) = y(t) + t \\ y(0) = 1 \end{cases}$$

a) Vérification de la solution analytique:  $y(t) = -1 - t + 2e^t$

$$\text{On a : } y'(t) = -1 + 2e^t = y(t) + t$$

b) Méthode d'Euler s'écrit:

$$\begin{aligned} y_{k+1} &= y_k + hf(t_k, y_k) \\ &= y_k + h(t_k + y_k) \\ y_{k+1} &= \mathbf{y}_k(\mathbf{1} + \mathbf{h}) + \mathbf{h}t_k \end{aligned}$$

c) Calcule de la solution au point 0.3.

On a  $h = 0.1$ ,  $y_0 = 1$  et  $t_k = t_0 + kh$  et  $\mathbf{y}_{k+1} = \mathbf{1.1y}_k + \mathbf{0.1t}_k$

Donc :

$$\text{Pour } k = 0, y_1 = 1.1y_0 + 0.1t_0 = 1.1 \times (1) + 0.1(0) = 1.1$$

$$\text{Pour } k = 1, y_2 = 1.1y_1 + 0.1t_1 = 1.1 \times (1.1) + 0.1(0.1) = 1.22$$

$$\text{Pour } k = 2, y_3 = 1.1y_2 + 0.1t_2 = 1.1 \times (1.22) + 0.1(0.2) = 1.362$$

C'est à dire que l'approximation en  $t = 0.3$  de  $y(t)$ , est  $y_3 = 1.362$ .

### Solution de l'exercice 3 :

#### 1) Méthode d'Euler

a) On a  $y' = t \sin(y(t))$ ,  $y(0) = 2$  et  $h = 0,1$ . On a donc que  $t_0 = 0$ , que  $y_0 = 2$  et que  $f(t_k, y_k) = t_k \sin(y_k)$ .

$$\text{Euler : } \begin{cases} y_{k+1} = y_k + hf(t_k, y_k) \\ t_{k+1} = t_k + h \end{cases}$$

$$-y_0 = 2$$

$$-y_1 = 2 + 0,1 \times 0 \times \sin 2 = 2$$

$$-y_2 = 2 + 0,1 \times 0,1 \sin 2 = 2,0090929$$

$$-y_3 = 2,0090929 + 0,1 \times 0,2 \times \sin 2,0090929 = 2,02720249$$

$$y_3 = 2,02720249.$$

#### 2) Méthode RK4

$$\begin{cases} y_{k+1} = y_k + \frac{h}{6} [k_1 + 2k_2 + 2k_3 + k_4] \\ t_{k+1} = t_k + h \end{cases}, k = 0, \dots, N-1 \quad \text{avec} \quad \begin{cases} k_1 = f(t_k, y_k) \\ k_2 = f(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_1) \\ k_3 = f(t_k + \frac{1}{2}h, y_k + \frac{1}{2}hk_2) \\ k_4 = f(t_k + h, y_k + hk_3) \end{cases}$$

On a que  $h = 0,1$ ,  $t_0 = 0$ ,  $y_0 = 2$  et que  $f(t_k, y_k) = t_k \sin(y_k)$ .

▪ Pour la première itération, on obtient :

$$k_1 = f(t_0, y_0) = 0,1 \times 0 \times \sin 2 = 0$$

$$k_2 = f\left(0 + 0,05, 2 + \frac{0}{2}\right) = f(0,05, 2) = 0,05 \times \sin 2 = 0,04546487$$

$$k_3 = f(0,05, 2 + 0,004546487/2) = f(0,05, 2,002273244) \\ = 0,05 \times \sin(2,002273244) = 0,04541745$$

$$k_4 = f(0,1, 2,004541745) = 0,0907398$$

$$y_1 = y_0 + \frac{h}{6} [k_1 + 2k_2 + 2k_3 + k_4] = 2,004541741$$

De même, on trouve que:

▪ Pour la deuxième itération, on obtient :

$$k_1 = 0,09074, k_2 = 0,13582, k_3 = 0,13568, k_4 = 0,18032$$

$$y_2 = 2,01810947.$$

▪ Pour la troisième itération, on obtient :

$$k_1 = 0,18032, k_2 = 0,22442, k_3 = 0,22418, k_4 = 0,26751$$

$$y_3 = 2,04052645.$$



b) On a :

$$y'(t) = y(t)e^t, y(0) = 2 \text{ et } h = 0,1.$$

Donc, on a également que  $t_0 = 0, y_0 = 2$  et que  $f(t_k, y_k) = y_k e^{t_k}$ .

1) *Méthode d'Euler* :

$$y_1 = 2,2, \quad y_2 = 2,4431376, \quad y_3 = 2,741543$$

2) *Méthode de Runge-Kutta 4 (RK4)*

• *Première itération* :

$$k_1 = 2, \quad k_2 = 2.20767, \quad k_3 = 2.21859, \quad k_4 = 2.45553$$

$$y_1 = 2.2218007$$

• *Deuxième itération* :

$$k_1 = 2.45547, \quad k_2 = 2.72401, \quad k_3 = 2.73961, \quad k_4 = 3.04833$$

$$y_2 = 2.495651$$

• *Troisième itération* :

$$k_1 = 0,304820 \quad k_2 = 0,340018 \quad k_3 = 0,342278 \quad k_4 = 0,383080$$

$$y_3 = 2,8377328$$

**Solution de l'exercice 4 :**

Approche de la racine à  $\varepsilon = 10^{-4}$  près par la méthode de Newton avec  $x_0 = 3$ .

$$\text{Algorithme de Newton} \quad \begin{cases} x_0 = 3 \\ x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})} \end{cases}$$

On a :

$$f(x) = \ln(x) - x + 2 = 0 \Rightarrow f'(x) = \frac{1}{x} - 1$$

Avec  $f(x^{(k)}) = \ln(x^{(k)}) - x^{(k)} + 2$

$$x^{(1)} = x^{(0)} - \frac{f(x^{(0)})}{f'(x^{(0)})} = 3 - \frac{f(3)}{f'(3)} = 3.1479 \quad |x^{(1)} - x^{(0)}| = |3.1479 - 3| = 0.1479$$

$$x^{(2)} = x^{(1)} - \frac{f(x^{(1)})}{f'(x^{(1)})} = 3.1479 - \frac{f(3.1479)}{f'(3.1479)} = 3.1462 \quad |x^{(2)} - x^{(1)}| = |3.1462 - 3.1479| = 0.0017$$

$$x^{(3)} = x^{(2)} - \frac{f(x^{(2)})}{f'(x^{(2)})} = 3.1462 - \frac{f(3.1462)}{f'(3.1462)} = 3.1462 \quad |x^{(3)} - x^{(2)}| = |3.1462 - 3.1462| = 0.0000$$

Donc la solution approchée est  $x^{(3)} = 3.1462$ .

**Solution de l'exercice 5 :**

On l'intégrale :

$$I = \int_0^{\frac{\pi}{2}} f(x) dx$$

a) Soit  $T$  l'approximation de  $I$  par la méthode des trapèzes, le pas  $T$  est donné par,  $h = \frac{z_s - z_0}{N} = \frac{\pi}{8}$ .

$$\begin{aligned} T &= \frac{h}{2} \left( f(x_0) + f(x_1) + 2 \sum_{i=1}^3 f(x_i) \right) \\ &= \frac{\pi}{16} (0 + 1 + 2(0,382683 + 0.707107 + 0.92388)) \\ &= 0.987116 \end{aligned}$$

b) Soit  $S$  l'approximation de  $I$  par la méthode de Simpson. Celle-ci s'écrit,

$$\begin{aligned} S &= \frac{h}{3} (y_0 + y_4 + 4(y_1 + y_3) + 2y_2) \\ &= \frac{\pi}{8} \cdot \frac{1}{3} [(0 + 1 + 4(0.38 \dots + 0.92 \dots) + 2 \cdot 0.707)] \\ &= 1.000135 \end{aligned}$$

Les points d'appui donnés dans cet exercice correspondent à la fonction  $\sin x$ . Et  $I = \int_0^{\frac{\pi}{2}} \sin x dx = 1$ . On constate donc que l'approximation de  $I$  par Simpson est meilleure que celle par les trapèzes, puisque  $|S - I| = 0.000135$  et  $|T - I| = 0.012884$ .

### Solution de l'exercice 6 :

On sait que l'accélération  $\gamma$  est la dérivée de la vitesse  $V$ , donc,

$$\begin{aligned} V(t) &= V(0) + \int_0^t \gamma(s) ds \\ V(80) &= 0 + \underbrace{\int_0^{80} \gamma(s) ds}_I \end{aligned}$$

a) Calculons  $I$  par la méthode des trapèzes. Ici, d'après le tableau des valeurs,  $h = 10$ .

$$\begin{aligned} I &= \frac{h}{2} \left( \gamma(x_0) + \gamma(x_n) + 2 \sum_{i=1}^{n-1} \gamma(x_i) \right) \\ &= \frac{1}{2} \cdot 10(30 + 50,67 + 2(31,63 + \dots + 46,70)) \\ &= 3089 \text{m}^{-\text{s}^{-1}} \end{aligned}$$

b) Calculons  $I$  par la méthode de Simpson.

$$\begin{aligned} V(80) &= \frac{h}{3} (\gamma(x_0) + \gamma(x_n) + 4(\gamma(x_1) + \gamma(x_3) + \dots) + 2(\gamma(x_2) + \gamma(x_4) + \dots)) \\ &= \frac{10}{3} (30 + 50,67 + 4(31,63 + 35,47 + \dots) + 2(33,44 + 37,75 + \dots)) \\ &= 3087 \text{ m}^{-1} \text{ s}^{-1} \end{aligned}$$

