Programme méthode numérique appliquée et optimisation

Chapitre01 : Rappels sur quelque méthode numérique

1. Résolution des systèmes d'équation linéaire et non linéaire par les méthodes itératives (Jacobi, Gauss-Seidel, Newton)
2. Résolution d'équation différentielle ordinaire
   (Euler, Taylor, Runge-Kutta)
3. Résolution des systèmes des équations différentielles ordinaires
   (Euler, Taylor, Runge-Kutta)
4. Intégration (Trapèze, Simpson) et différentiation (Différence finies)

Chapitre02 : Equation aux dérivées partielles

1. Méthode des différences finies (MDF)
2. Méthode des éléments finis (MEF)

Chapitre03 : Technique d'optimisation

Définition et formulation : Problème d'optimisation, technique d'optimisation

Algorithme d'optimisation

1 .Méthodes itératives pour les systèmes d'équations linéaires

On va traiter dans ce chapitre les systèmes d'équations linéaires dont le nombre d'équations est égal à ceux d'inconnues et dont le déterminant est non nul. C'est-à-dire les systèmes qui ont une solution unique. Un système d'équations linéaire de n équations avec n inconnues s'écrit alors :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots a_{2n}x_n = b_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots . \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots a_{nn}x_n = b_n \end{cases} \qquad (1)$$

Ce système peut être écrit sous forme matricielle :

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{1n} \\ a_{21} & a_{22} \cdots & a_{2n} \\ & \vdots & \\ a_{n1} & a_{n2} \ldots & a_{nn} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{X} = \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}}_{B} \Leftrightarrow AX = B$$

Ce système peut être résolu directement par différentes méthodes, on peut citer celle de Kramer (des déterminants) où celle d'élimination.

1) Méthode de Jacobi

Elle utilise le principe du point fixe pour un système d'équations linéaires,

$$\begin{cases} x_1 = (b_1 - a_{12}x_2 - a_{13}x_3 - \cdots - a_{1n}x_n)/a_{11} \\ x_2 = (b_2 - a_{21}x_1 - a_{23}x_3 - \cdots - a_{2n}x_n)/a_{22} \\ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots. \\ x_n = (b_n - a_{n2}x_2 - a_{n3}x_3 - \cdots - a_{nn-1}x_{n-1})/a_{nn} \end{cases} \qquad (2)$$

Pour résoudre le système (1) on utilise l'écriture (2) en portant les termes de droite à l'itération (k) et ceux à gauche à l'intération (k).

$$\begin{cases} x_1^{(k+1)} = (b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)})/a_{11} \\ x_2^{(k+1)} = (b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)})/a_{22} \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ x_n^{(k+1)} = (b_n - a_{n2}x_2^{(k)} - a_{n3}x_3^{(k)} - \cdots - a_{nn-1}x_{n-1}^{(k)})/a_{nn} \end{cases} \qquad (3)$$

En prenant une estimation initiale $X^{(0)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})$ et en utilisant le système (3) on calcule $X^{(1)} = (x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)})$ ensuite on remplace le vecteur $X^{(1)}$ dans le système (3) avec k=1 on calcule $X^{(2)}$ et continue de la même façon de calculer les vecteurs $X^{(3)}$, $X^{(4)}$, $X^{(5)}$,… jusqu'à la convergence.

2) Méthode de Gauss-Seidel

meilleure que $x_1^{(1)}$. D'ici vient le principe de la méthode de Gauss-Seidel, on utilise chaque composante des quelle sera calculée. Ainsi, pour calculer la composante $x_i^{(k+1)}$, on utilise toutes les composantes de $x_1^{(k+1)}$ à $x_{i-1}^{(k+1)}$ déjà calculées à l'itération *(k+1)* en plus de celles $x_{i+1}^{(k)}$ à $x_n^{(k)}$ qui ne sont qu'à l'itération *(k)*. On écrit donc :

$$\begin{cases} x_1^{(k+1)} = (b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)})/a_{11} \\ x_2^{(k+1)} = (b_2 - a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)})/a_{22} \\ x_3^{(k+1)} = (b_3 - a_{31}x_1^{(k+1)} - a_{32}x_3^{(k+1)} - \cdots - a_{3n}x_n^{(k)})/a_{33} \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ x_n^{(k+1)} = (b_n - a_{n2}x_2^{(k+1)} - a_{n3}x_3^{(k+1)} - \cdots - a_{nn-1}x_{n-1}^{(k+1)})/a_{nn} \end{cases} \qquad (4)$$

En prenant une estimation initiale $X^{(0)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})$ et en utilisant le système (4) on calcule $X^{(1)} = (x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)})$ ensuite on remplace le vecteur $X^{(1)}$ dans le système (4) avec *k=1* on calcule $X^{(2)}$ et continue de la même façon de calculer les vecteurs $X^{(3)}$, $X^{(4)}$, $X^{(5)}$,… jusqu'à la convergence.

On arrête les calculs pour cette méthode lorsque la différence absolue entre deux itérations successives soit inférieure à une certaine précision ε donnée.

$$\left|X^{(n+1)} - X^{(n)}\right| < \boldsymbol{\varepsilon}$$

Ici, il faut vérifier la différence pour toutes les composantes une par une.

$$\left|x_1^{(n+1)} - X_1^{(n)}\right| < \boldsymbol{\varepsilon}, \quad \left|x_2^{(n+1)} - X_2^{(n)}\right| < \boldsymbol{\varepsilon}, \ldots, \left|x_n^{(n+1)} - X_n^{(n)}\right| < \boldsymbol{\varepsilon}$$

2. Méthode itérative pour les systèmes d'équation non linéaire :

La méthode de Newton :

Soit les systèmes d'équation NL suivante :

$$\begin{cases} f_1(x_1, x_2, \ldots\ldots\ldots, x_n) = 0 \\ f_2(x_1, x_2, \ldots\ldots\ldots, x_n) = 0 \\ \qquad \vdots \\ f_n(x_1, x_2, \ldots\ldots\ldots, x_n) = 0 \end{cases}$$

On peut étendre la méthode de newton au cas vectoriel

$$\textit{poser} \quad X^{k+1} = X^k + \delta X^k$$

$$\textit{résoudre} \quad \delta X^k = -\left[J_F\left(X^k\right)\right]^{-1} . F\left(X^k\right)$$

$J_F\left(X^k\right)$ La matrice Jacobienne :

$$J_F\left(X^k\right) = \begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 & \ldots\ldots & \partial f_1/\partial x_n \\ \partial f_2/\partial x_1 & \partial f_2/\partial x_2 & \ldots\ldots & \partial f_2/\partial x_n \\ \vdots & \vdots & \ldots\ldots & \vdots \\ \partial f_n/\partial x_1 & \partial f_n/\partial x_2 & \ldots\ldots & \partial f_n/\partial x_n \end{bmatrix}$$

Exemple : Soit le système suivant :

$$\begin{cases} x_1 + x_1 x_2 - 4 = 0 \\ x_1 + x_2 - 3 = 0 \end{cases} \textit{Avec} \quad x_1^0 = 1.98 \quad , x_2^0 = 1{,}02$$

3. Résolution des équations différentielles

1) Méthode d'Euler

Soit l'équation différentielle du premier ordre avec la condition de Cauchy :

$$\begin{cases} y'(t) = f(t, y(t)) \\ y(t_0) = y_0 \end{cases} \quad t \in [t_0, T] \quad (1)$$

Divisons l'intervalle $[t_0, T]$ en n parties égales, *c.à.d.* le pas $h = \frac{T - t_0}{n}$

Ecrivons $t_k = t_0 + kh$ $(k=0,1,\ldots,n)$

La tangente à la courbe $y=y(t)$ en $t=t_0$ a pour équation :

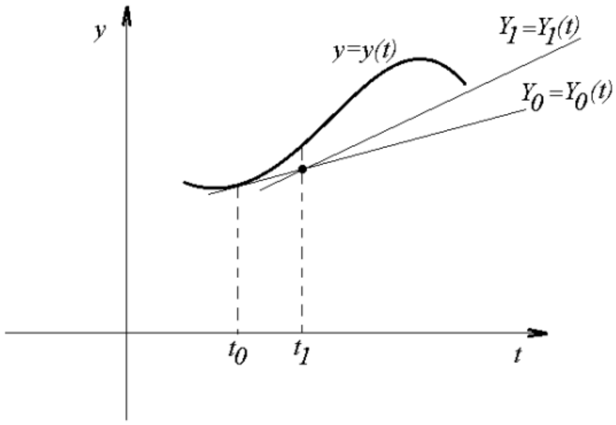
$$Y_0(t) = y(t_0) + y'(t_0)(t - t_0) \qquad (2)$$

Selon *(1)*, l'équation *(2)* peut s'écrire :

$$Y_0(t) = y(t_0) + f(t_0, y(t_0))(t - t_0)$$

Encore :

$$Y_0(t) = y(t_0) + f(t_0, y_0)(t - t_0)$$



Au point $t=t_1$:

$$Y_0(t_1) = y(t_0) + f(t_0, y_0)(t_1 - t_0)$$

Or $\qquad h=t_1 - t_0$

$$Y_0(t_1) = y_0 + hf(t_0, y_0)$$

En approximant $Y_0(t_1) \approx y_1$, on peut écrire :

$$y_1 = y_0 + hf(t_0, y_0)$$

De même considérons la droite d'équation $Y_1(t) = y(t_1) + y'(t_1)(t - t_1)$

Au point $t=t_2$

$Y_1(t_2) = y_1 + y'(t_1)(t - t_1)$ en faisant de même nous aurons :

$$y_2 = y_1 + hf(t_1, y_1)$$

Au point $t=t_{n+1}$

$$y_{n+1} = y_n + hf(t_n, y_n)$$

2) Méthode de Taylor

Le développement de Taylor permet de diminuer l'erreur d'approximation

$$y(t_{i+1}) = y(t_i + h)$$

$$= y(t_i) + h.y'(t_i) + \frac{h^2}{2}.y''(t_i) + O(h^2)$$

$$= y(t_i) + h.f\left(t_i, y(t_i)\right) + \frac{h^2}{2}.f'\left(t_i, y(t_i)\right)$$

$$et \ \ on \ \ a \ \ f'\left(t_i, y(t_i)\right) = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}.\frac{\partial y}{\partial t}$$

$$\Rightarrow f'\left(t_i, y(t_i)\right) = \frac{\partial f\left(t_i, y(t_i)\right)}{\partial t} + \frac{\partial f\left(t_i, y(t_i)\right)}{\partial y}.f\left(t_i, y(t_i)\right)$$

On obtient :

$$y(t_{i+1}) \simeq y_{i+1} = y_i + h.f\left(t_i, y(t_i)\right) + \frac{h^2}{2}.\left[\frac{\partial f\left(t_i, y(t_i)\right)}{\partial t} + \frac{\partial f\left(t_i, y(t_i)\right)}{\partial y}.f\left(t_i, y(t_i)\right)\right]$$

3) Méthode de Runge Kutta

<u>RK2</u>

$$y_{i+1} = y_i + \frac{h}{2}.(k_1 + k_2)$$

$$\begin{cases} k_1 = f\left(t_i, y_i\right) \\ k_2 = f\left(t_i + h, y_i + h.k_1\right) \end{cases}$$

<u>RK4</u>

$$y_{i+1} = y_i + \frac{1}{6}.(k_1 + 2k_2 + 2k_3 + k_4)$$

$$\begin{cases} k_1 = h.f\left(t_i, y_i\right) \\ k_2 = h.f\left(t_i + h/2, y_i + k_1/2\right) \\ k_3 = h.f\left(t_i + h/2, y_i + k_2/2\right) \\ k_4 = h.f\left(t_i + h, y_i + k_3\right) \end{cases}$$

<u>Exemple d'application :</u>

4. *Résolution des systèmes des équations différentielle ordinaire :*

$Soit\ \ y^{(p)} = f\left(t, y, y', y'', ..., y^{(p)}\right)$

Equation différentielle d'ordre supérieur

On peut ramener le problème à celui d'un système d'équation du 1er ordre

$Z = \left(z_1, z_2, z_3, ..., z_p\right) \quad et \quad y = z_1$

On obtient le système suivant :

$$\begin{cases} z_1' = z_2 \\ z_2' = z_3 \\ \vdots \\ z_{p-1}' = z_p \\ z_p' = f\left(t, z_1, z_2, .., z_p\right) \end{cases}$$

1) Euler

$$Z_{i+1} = Z_i + h.F\left(t_i, Z_i\right)$$

$$avec \quad Z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{pmatrix}, \quad F = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_p \end{pmatrix}$$

2) Taylor

$$Y' = \begin{bmatrix} y_1'(t) \\ y_2'(t) \end{bmatrix} = \begin{bmatrix} f_1\left(t, y_1(t), y_2(t)\right) \\ f_2\left(t, y_1(t), y_2(t)\right) \end{bmatrix} \quad Avec \begin{bmatrix} y_1(t_0) \\ y_2(t_0) \end{bmatrix} = \begin{bmatrix} y_{10} \\ y_{2à} \end{bmatrix}$$

$$Y_{k+1} = \begin{bmatrix} y_{1(k+1)} \\ y_{2(k+1)} \end{bmatrix} = \begin{bmatrix} y_{1k} + h.f_1\left(t_k, y_{1k}, y_{2k}\right) \\ y_{2k} + h.f_2\left(t_k, y_{1k}, y_{2k}\right) \end{bmatrix} + \frac{h^2}{2}. \begin{bmatrix} \dfrac{\partial f_1}{\partial t} + \dfrac{\partial f_1}{\partial y_1}.f_1 + \dfrac{\partial f_1}{\partial y_2}.f_2 \\ \dfrac{\partial f_2}{\partial t} + \dfrac{\partial f_2}{\partial y_1}.f_1 + \dfrac{\partial f_2}{\partial y_2}.f_2 \end{bmatrix}_{\left(t_k, y_{1k}, y_{2k}\right)}$$

Pour la première itération k=0

$$Y_{11}=\begin{bmatrix} y_{11}\\ y_{21}\end{bmatrix}=\begin{bmatrix} y_{10}+h.f_1\left(t_0,y_{10},y_{20}\right)\\ y_{20}+h.f_2\left(t_0,y_{10},y_{20}\right)\end{bmatrix}+\frac{h^2}{2}.\begin{bmatrix} \frac{\partial f_1}{\partial t}+\frac{\partial f_1}{\partial y_1}.f_1+\frac{\partial f_1}{\partial y_2}.f_2\\ \frac{\partial f_2}{\partial t}+\frac{\partial f_2}{\partial y_1}.f_1+\frac{\partial f_2}{\partial y_2}.f_2\end{bmatrix}_{\left(t_0,y_{10},y_{20}\right)}$$

3) Méthode de Runge Kutta

<u>RK2</u>

$$Y'=\begin{bmatrix} y_1'(t)\\ y_2'(t)\end{bmatrix}=\begin{bmatrix} f_1\left(t,y_1(t),y_2(t)\right)\\ f_2\left(t,y_1(t),y_2(t)\right)\end{bmatrix}\quad Avec\ \begin{bmatrix} y_1(t_0)\\ y_2(t_0)\end{bmatrix}=\begin{bmatrix} y_{10}\\ y_{20}\end{bmatrix}$$

$$Y_{k+1}=Y_k+\frac{h}{2}\left(K_1+K_2\right)$$

$$K_1=\begin{pmatrix} k_{11}\\ k_{12}\end{pmatrix}=\begin{pmatrix} f_1\left(t_k,y_{1k},y_{2k}\right)\\ f_2\left(t_k,y_{1k},y_{2k}\right)\end{pmatrix}$$

$$K_2=\begin{pmatrix} k_{21}\\ k_{22}\end{pmatrix}=\begin{pmatrix} f_1\left(t_k+h,y_{1k}+h.k_{11},y_{2k}+h.k_{12}\right)\\ f_2\left(t_k+h,y_{1k}+h.k_{11},y_{2k}+h.k_{12}\right)\end{pmatrix}$$

Pour la première itération k=0

$$Y_1=Y_0+\frac{h}{2}\left(K_1+K_2\right)$$

$$K_1=\begin{pmatrix} k_{11}\\ k_{12}\end{pmatrix}=\begin{pmatrix} f_1\left(t_0,y_{10},y_{20}\right)\\ f_2\left(t_0,y_{10},y_{2}\right)\end{pmatrix}$$

$$K_2=\begin{pmatrix} k_{21}\\ k_{22}\end{pmatrix}=\begin{pmatrix} f_1\left(t_0+h,y_{10}+h.k_{11},y_{20}+h.k_{12}\right)\\ f_2\left(t_0+h,y_{10}+h.k_{11},y_{20}+h.k_{12}\right)\end{pmatrix}$$

RK4

$$Y_{k+1}=Y_k+\frac{h}{6}\left(K_1+2K_2+2K_3+K_4\right)$$

$$K_1=\begin{pmatrix} k_{11}\\ k_{12}\end{pmatrix}=\begin{pmatrix} f_1\left(t_k,y_{1k},y_{2k}\right)\\ f_2\left(t_k,y_{1k},y_{2k}\right)\end{pmatrix}$$

$$K_2=\begin{pmatrix} k_{21}\\ k_{22}\end{pmatrix}=\begin{pmatrix} f_1\left(t_k+h/2,y_{1k}+h/2.k_{11},y_{2k}+h/2.k_{12}\right)\\ f_2\left(t_k+h/2,y_{1k}+h/2.k_{11},y_{2k}+h/2.k_{12}\right)\end{pmatrix}$$

$$K_3=\begin{pmatrix} k_{31}\\ k_{32}\end{pmatrix}=\begin{pmatrix} f_1\left(t_k+h/2,y_{1k}+h/2.k_{21},y_{2k}+h/2.k_{22}\right)\\ f_2\left(t_k+h/2,y_{1k}+h/2.k_{21},y_{2k}+h/2.k_{22}\right)\end{pmatrix}$$

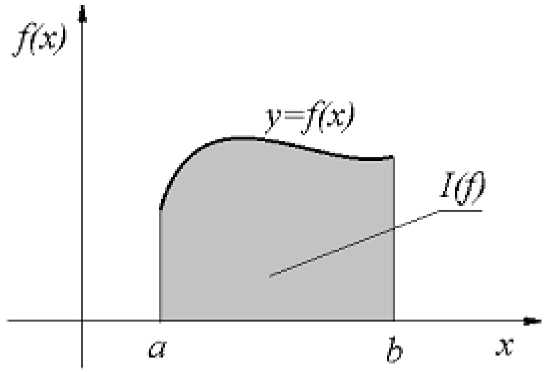
$$K_4=\begin{pmatrix} k_{41}\\ k_{42}\end{pmatrix}=\begin{pmatrix} f_1\left(t_k+h,y_{1k}+h.k_{31},y_{2k}+h.k_{32}\right)\\ f_2\left(t_k+h,y_{1k}+h.k_{31},y_{2k}+h.k_{32}\right)\end{pmatrix}$$

5 *Intégration et dérivation numérique :*

5.1 Intégration numérique

Introduction

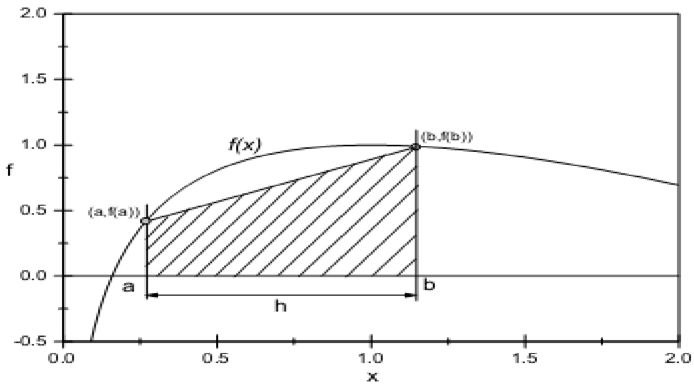
Soit une fonction *y=f(x)* définie et continue sur un intervalle *[a,b]*



Trouver l'intégrale définie $\int_a^b f(x)dx$ veut dire calculer l'aire $I(f)$ délimitée par les droites *y=0 ; x=a ; x=b* et la courbe *y=f(x)*.

1.Formule du trapéze

Cette formule est très simple, elle permet de remplacer la courbe ***f(x)*** de la fonction à intégrer par une ligne droite qui relie les points ***(a,f(a))*** et ***(b,f(b))*** ce qui donne un trapèze (Fig. 3.1).



L'intégrale est donc remplacée par la surface du trapèze :

$$s = \int_a^b f(x) = \frac{h}{2}(f(a) + f(b))$$

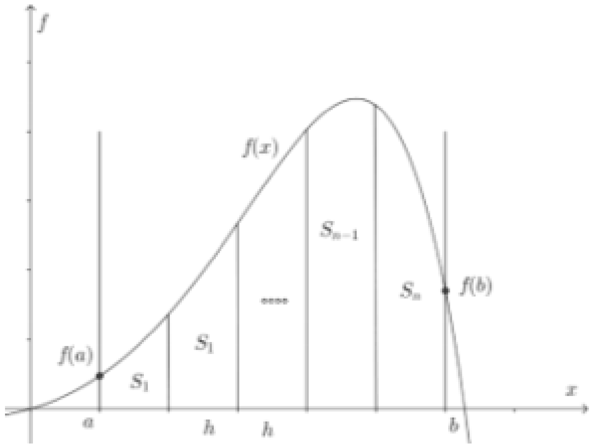
Avec h=b-a est dit pas d'intégration

2. Formule générale du trapèze

On divise l'intervalle ***[a,b]*** en plusieurs sous intervalles égaux et on applique la formule du trapèze à chaque sous intervalle (Fig. 3.2). On a donc les sous intervalles $[\boldsymbol{a = x_0, x_1}] \cup [\boldsymbol{x_1, x_2}] \cup \ldots\ldots \cup [\boldsymbol{x_{n-1}, x_n = b}]$, l'application de la formule du trapèze donne :

$$\int_a^b f(x) = \frac{h}{2}(f(x_0) + f(x_1)) + \frac{h}{2}(f(x_1) + f(x_2)) + \frac{h}{2}(f(x_2) + f(x_3)) + \cdots + \frac{h}{2}(f(x_{n-1}) + f(x_n))$$

$$\int_a^b f(x) = \frac{h}{2}\left(f(x_0) + 2\sum_{i=1}^{n-1} f(x_i) + f(x_n)\right) = \frac{h}{2}\left(f_0 + 2\sum_{i=1}^{n-1} f_i + f_n\right)$$



3. Formule générale du Simpson

Dans cette formule on ne remplace pas la fonction par une droite mais par une parabole de degré $\boldsymbol{n}$ inférieure ou égale à deux. Cette dernière doit passer par trois points $(\boldsymbol{x_0, y_0}), (\boldsymbol{x_1, y_1})$ **et** $(\boldsymbol{x_2, y_2})$ ce qui fait que cette méthode n'est applicable que pour un nombre pair de tranches (une tranche c'est l'intervalle entre deux points) (Fig. 3.4.). La formule de Simpson s'écrit :

$$\int_a^b \boldsymbol{f(x)} \cong \frac{\boldsymbol{h}}{\boldsymbol{3}}(\boldsymbol{f(x_0)} + \boldsymbol{4f(x_1)} + \boldsymbol{f(x_2)})$$

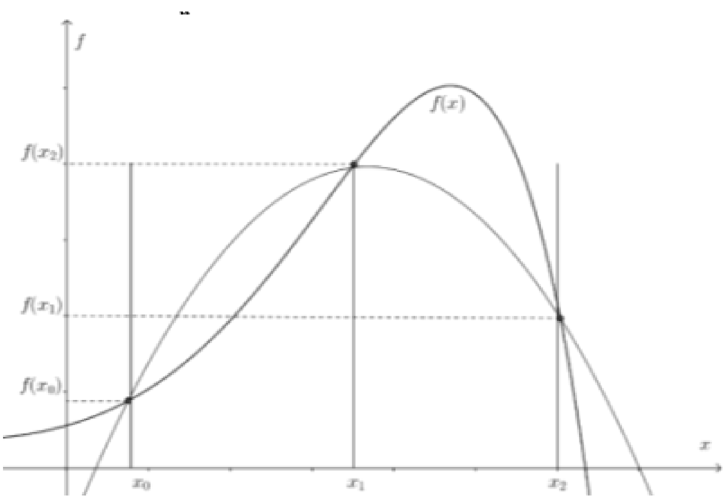


Fig. 3.4. Méthode de Simpson

Si on généralise la formule de Simpson pour $\boldsymbol{2n}$ sous intervalles avec un pas d'intégration $\boldsymbol{h = \frac{b-a}{2n}}$, $\boldsymbol{a = x_0 < x_1 < \cdots \ldots < x_{2n} = b}$ et $\boldsymbol{x_k = a + hk}$ pour $\boldsymbol{k=0,1,2,\ldots\ldots,2n}$.

La formule de Simpson généralisée s'écrit :

$$\int_a^b f(x) \cong \frac{h}{3}\left(f(x_0) + 2\sum_{i\,pair} f(x_i) + 4\sum_{i\,impair} f(x_i) + f(x_{2n})\right)$$