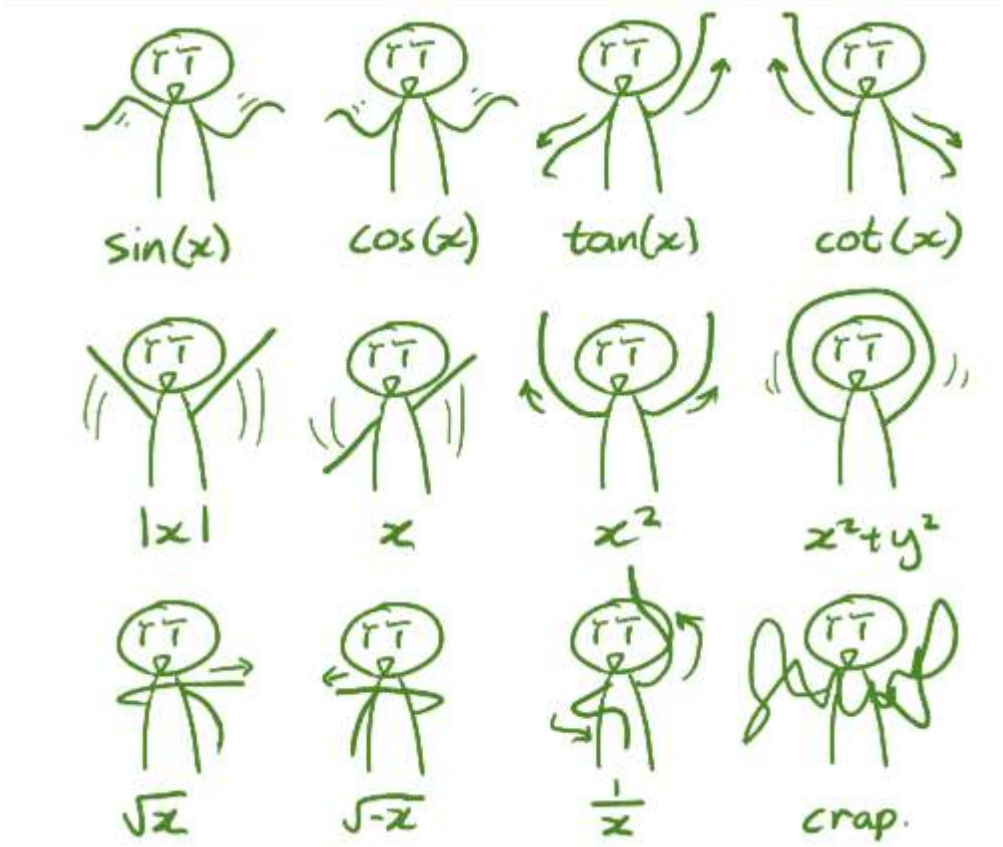
GRO 305 : Mathématiques pour l'ingénieur

Programme de Génie Robotique



Nicolas Quaegebeur

Adapté des notes de cours d'**Alain Berry** (dept. Génie mécanique) et de **Jean de La Fontaine** (dept. Génie électrique)

UNIVERSITÉ DE SHERBROOKE

Version 3

Été 2019 - Nicolas Quaegebeur

Mise à jour été 2023 – Abdelaziz Ramzi

Table des matières

Chapitre 1 : Rappels - calcul différentiel et intégral

1.1 Introduction

Le calcul différentiel et intégral est une composante fondamentale du Calcul, et une application importante du calcul infinitésimal, développé à peu près simultanément par Newton (1642-1727) et Leibniz (1646-1716).

Ces notions mathématiques ont été inventées dans le cadre de l'étude du mouvement : la question très simple de savoir quelle est la vitesse atteinte après *t* secondes par une bille lâchée sans vitesse initiale à *t*=0 a débouché sur la notion de vitesse *instantanée* et de *dérivée* d'une fonction (la vitesse instantanée est la dérivée de la fonction-position), des notions inconnues jusqu'alors. La notion de dérivée a à son tour ouvert la voie à des développements cruciaux en mathématiques et en physique : recherche d'extremums, optimisation, théorie des équations différentielles.

La notion d'*intégrale* est apparue presque simultanément à celle de dérivée pour répondre à la question inverse : comment déterminer la position de la bille connaissant sa vitesse en tout temps ? La position de la bille à un instant donné est l'*intégrale* de la vitesse depuis *t*=0 jusqu'à cet instant. Cette notion d'intégrale a rapidement trouvé des applications importantes dans le calcul d'aires de surfaces planes, de longueurs de courbes, de volumes et de masses (ces calculs étaient faits jusqu'alors à partir de la géométrie euclidienne).

Ce chapitre expose les techniques élémentaires du calcul différentiel et intégral. La section 2 présente les principaux systèmes de coordonnées d'espace utilisés en Calcul. Les sections 3 (Dérivée), 4 (Intégrale) et 5 (Séries de Taylor et de Maclaurin) sont principalement des rappels de notions vues au Collégial.

1.2 Systèmes de coordonnées dans l'espace

Dans le but par exemple d'étudier le mouvement d'un objet dans l'espace, ou de caractériser la géométrie d'un objet en trois dimensions, il est nécessaire de se munir d'un système de repérage approprié de cet objet, c'est-à-dire d'un système de coordonnées dans l'espace. Suivant la nature du problème, il existe différents systèmes de coordonnées qui peuvent être utilisés plus ou moins avantageusement ; cette section présente les trois systèmes de coordonnées d'espace les plus utilisés : les coordonnées *cartésiennes*, les coordonnées *cylindriques* et les coordonnées *sphériques*.

1.2.1 Coordonnées cartésiennes

Un point *M* dans l'espace est repéré par ses coordonnées cartésiennes (x, y, z) dans le repère cartésien orthogonal (O, x, y, z) (Figure 1.1). Si $(\hat{\imath},\ \hat{\jmath},\ \hat{k})$ sont les trois vecteurs unitaires portés par les axes x, y, z respectivement, alors le vecteur position du point M par rapport au point O s'exprime

$$\overrightarrow{r^{M/O}} = x\,\hat{\imath} + y\,\hat{\jmath} + z\,\hat{k}$$

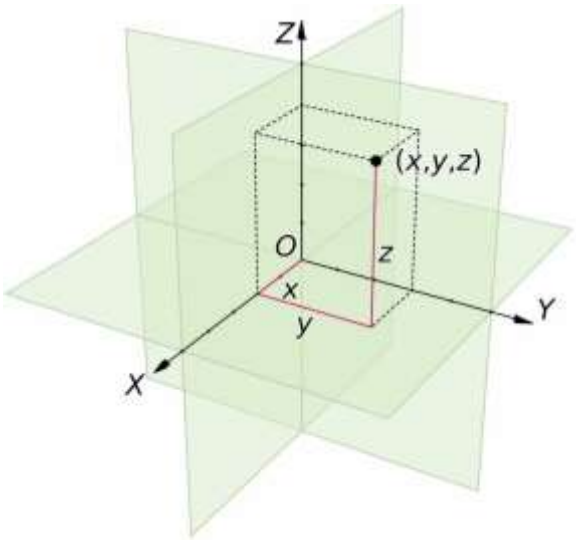


Figure 1.1 : Coordonnées cartésiennes d'espace

Exemple :
Écrivons en coordonnées cartésiennes l'équation de la sphère de rayon 2 centrée au point (1, 0, - 2). Cette sphère est la surface formée par l'ensemble des points (x, y, z) dont la distance au point (1, 0, - 2) est égale à 2. L'équation de cette sphère en coordonnés cartésiennes est donc

$$\sqrt{(x-1)^2 + (y-0)^2 + (z-(-2)^2} = 2$$

ou encore, $\quad (x-1)^2 + y^2 + (z+2)^2 = 4$

1.2.2 Coordonnées cylindriques

Le système de coordonnées cylindriques est bien adapté pour décrire par exemple l'écoulement d'un fluide dans une conduite, ou pour définir la géométrie de surfaces coniques ou cylindriques.

Définition :
Les coordonnées cylindriques (r, θ, z) d'un point M dans l'espace sont définies de la manière suivante (figure 1.2):

- r et θ sont les coordonnées polaires de la projection de M dans le plan (O, x, y) $(r \geq 0,\ 0 \leq \theta \leq 2\pi)$
- z *est* la troisième coordonnée cartésienne de M.

Relations entre les coordonnées cartésiennes et cylindriques :

$$\begin{cases} x = r\cos\theta \\ y = r\sin\theta \\ z = z \end{cases} \qquad \begin{cases} r = \sqrt{x^2 + y^2} \\ \theta = \tan^{-1}\dfrac{y}{x} \quad 0 \leq \theta \leq 2\pi \\ z = z \end{cases}$$

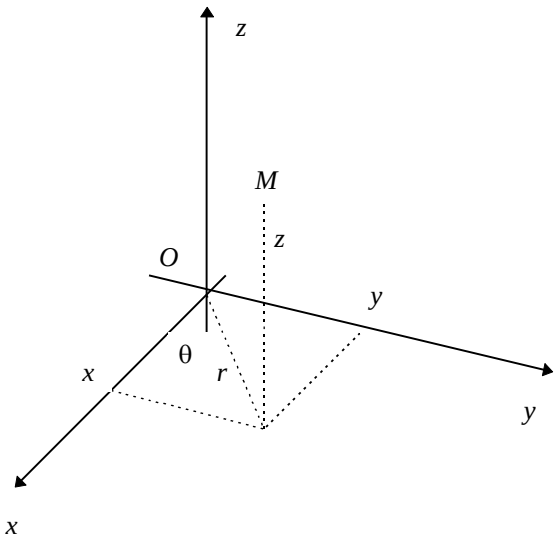


Figure 1.2 : Coordonnées cylindriques d'espace

Exemple :
Écrivons en coordonnées cylindriques l'équation de la sphère de rayon 2 centrée au point (1, 0, - 2). Cette sphère a comme équation en coordonnées cartésiennes $(x-1)^2 + y^2 + (z+2)^2 = 4$. En substituant dans cette équation $x = rcos\theta, y = rsin\theta$, on obtient

$$(r\cos\theta - 1)^2 + (r\sin\theta)^2 + (z+2)^2 = 4,$$

c'est-à-dire

$$r^2\cos^2\theta - 2r\cos\theta + 1 + (r\sin\theta)^2 + (z+2)^2 = 4$$

ou encore

$$r^2 - 2r\cos\theta + z^2 + 4z + 4 = 0$$

1.2.3 Coordonnées sphériques

Le système de coordonnées sphériques est bien approprié pour décrire par exemple la position d'un point sur la surface terrestre, ou pour représenter le mouvement des planètes autour du soleil.

Définition :
Les coordonnées sphériques (R,θ,ϕ) d'un point *M* dans l'espace sont définies de la manière suivante (figure 1.3):

- *R* est la distance de O à *M* $(r \geq 0)$;
- θ est l'angle polaire de la projection de *M* dans le plan (O,x,y) $(0 \leq \theta \leq 2\pi)$;
- ϕ est l'angle entre le vecteur **OM** et l'axe orienté *z* $(0 \leq \phi \leq \pi)$.

Dans l'exemple des coordonnées géographiques d'un point sur la surface terrestre, l'angle θ représente la longitude et l'angle ϕ représente la latitude du point.

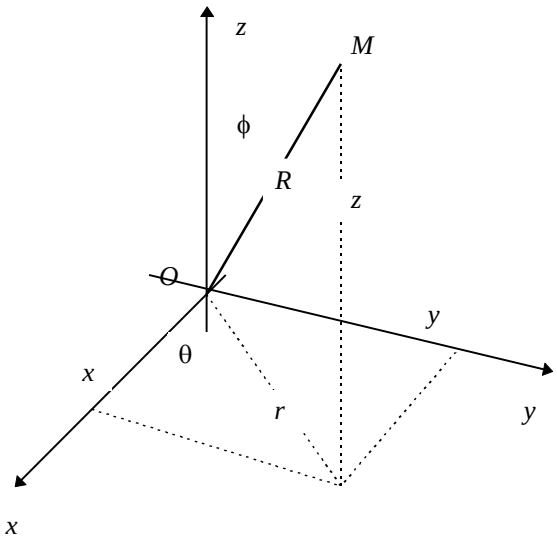


Figure 1.3 : Coordonnées sphériques d'espace

Relations entre les coordonnées cartésiennes, cylindriques et sphériques :

Cartésien ⇔ **Cylindrique**

$$\begin{cases} x = r\cos\theta \\ y = r\sin\theta \\ z = z \end{cases} \qquad \begin{cases} r = \sqrt{x^2 + y^2} \\ \theta = \tan^{-1}\dfrac{y}{x} \quad 0 \le \theta \le 2\pi \\ z = z \end{cases}$$

Cartésien ⇔ **Sphérique**

$$\begin{cases} x = R\cos\theta\sin\phi \\ y = R\sin\theta\sin\phi \\ z = R\cos\phi \end{cases} \qquad \begin{cases} R = \sqrt{x^2 + y^2 + z^2} \\ \theta = \tan^{-1}\dfrac{y}{x} \quad 0 \le \theta \le 2\pi \\ \phi = \cos^{-1}\dfrac{z}{R} \quad 0 \le \phi \le \pi \end{cases}$$

Cylindrique ⇔ **Sphérique**

$$\begin{cases} r = R\sin\phi \\ \theta = \theta \\ z = R\cos\phi \end{cases} \qquad \begin{cases} R = \sqrt{r^2 + z^2} \\ \theta = \theta \\ \phi = \tan^{-1}\dfrac{r}{z} \quad 0 \le \phi \le \pi \end{cases}$$

Exemple :

Écrivons en coordonnées sphériques l'équation de la sphère de rayon 2 centrée au point (1, 0, - 2). Cette sphère a comme équation en coordonnées cartésiennes $(x-1)^2 + y^2 + (z+2)^2 = 4$. En substituant dans cette équation

$\begin{cases} x = R\cos\theta\sin\phi \\ y = R\sin\theta\sin\phi \\ z = R\cos\phi \end{cases}$, on obtient : $(R\cos\theta\sin\phi - 1)^2 + (R\sin\theta\sin\phi)^2 + (R\cos\phi + 2)^2 = 4$,

c'est-à-dire : $R^2\cos^2\theta\sin^2\phi - 2R\cos\theta\sin\phi + 1 + R^2\sin^2\theta\sin^2\phi + R^2\cos^2\phi + 4R\cos\phi + 4 = 4$

ou encore : $R^2 - 2R\cos\theta\sin\phi + 4R\cos\phi + 1 = 0$

1.3 Dérivée

La notion de dérivée d'une fonction a été inventée à peu près simultanément par Leibniz (1646-1716) et Newton (1642-1727), dans le cadre de l'étude du mouvement et du calcul infinitésimal. Cette notion, l'une des plus importantes en mathématiques, a permis notamment de définir correctement la *vitesse instantanée* d'un objet en mouvement. On se contentera ici de simplement rappeler les définitions et les résultats importants

Définition :

La dérivée de la fonction $f : x \mapsto f(x)$ est la fonction $x \mapsto f'(x) = \frac{df}{dx}(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$.

La dérivée $f'(x)$ représente le taux d'accroissement instantané de la fonction *f* en *x*, ou encore la pente de la droite tangente à la courbe $y = f(x)$ en *x*.

Règles de la dérivation :

- $\frac{d}{dx}(c) = 0$ où *c* est une constante
- $\frac{d}{dx}(x^n) = nx^{n-1}$ où *n* est un nombre rationnel
- $\frac{d}{dx}(cf) = c\frac{df}{dx}$
- $\frac{d}{dx}(f + g) = \frac{df}{dx} + \frac{dg}{dx}$
- $\frac{d}{dx}(fg) = f\frac{dg}{dx} + g\frac{df}{dx}$
- $\frac{d}{dx}(\frac{f}{g}) = \frac{g\frac{df}{dx} - f\frac{dg}{dx}}{g^2}$

- Règle d'enchainement : $(f \circ g)' = (f' \circ g)g'$, ou encore $\frac{dy}{dt} = \frac{dy}{dx}\frac{dx}{dt}$
- $\frac{d}{dx}(f^n) = nf^{n-1}\frac{df}{dx}$ où *n* est un nombre rationnel
- $(f^{-1})' = \left(\frac{1}{f}\right)'$ où f^{-1} désigne la fonction inverse de *f*

Résultats :

- $\frac{d}{dx}(\sin x) = \cos x$
- $\frac{d}{dx}(\cos x) = -\sin x$
- $\frac{d}{dx}(\tan x) = \frac{1}{\cos^2 x} = \sec^2 x$
- $\frac{d}{dx}(\cot x) = -\frac{1}{\sin^2 x} = -\csc^2 x$
- $\frac{d}{dx}(\sec x) = \sec x \tan x$
- $\frac{d}{dx}(\csc x) = -\csc x \cot x$
- $\frac{d}{dx}(\ln x) = \frac{1}{x}$
- $\frac{d}{dx}(e^x) = e^x$
- $\frac{d}{dx}(\sin^{-1} x) = \frac{1}{\sqrt{1-x^2}}$
- $\frac{d}{dx}(\cos^{-1} x) = -\frac{1}{\sqrt{1-x^2}}$
- $\frac{d}{dx}(\tan^{-1} x) = \frac{1}{1+x^2}$
- $\frac{d}{dx}(\cot^{-1} x) = -\frac{1}{1+x^2}$
- $\frac{d}{dx}(\sec^{-1} x) = \frac{1}{|x|\sqrt{1-x^2}}$
- $\frac{d}{dx}(\csc^{-1} x) = -\frac{1}{|x|\sqrt{1-x^2}}$
- $\frac{d}{dx}(\sinh x) = \cosh x$
- $\frac{d}{dx}(\cosh x) = \sinh x$
- $\frac{d}{dx}(\tanh x) = \frac{1}{\cosh^2 x}$
- $\frac{d}{dx}(\coth x) = -\frac{1}{\sinh^2 x}$.

1.4 Intégrale

Comme pour la dérivée, ce sont Newton et Leibniz qui ont historiquement été les premiers à proposer le concept mathématique d'intégrale, toujours dans le cadre de l'étude du mouvement. Ils ont défini l'intégrale d'une fonction comme la limite de la somme de Riemann de cette fonction et ont démontré que le résultat obtenu représente l'aire inscrite sous la courbe représentative de cette fonction. La notion d'intégrale a révolutionné le calcul dans son entier, en particulier les calculs d'aires ou de volumes, qui étaient effectués jusque là à partir de la géométrie euclidienne.

On se contente encore une fois de rappeler les définitions et les résultats importants.

Définitions :

L'intégrale définie $\int_a^b f(x)dx$ est définie comme $\int_a^b f(x)dx = \lim_{\max(x_i - x_{i-1})\to 0} \sum_{i=1}^{n} f(c_i)(x_i - x_{i-1})$

où $x_0 = a < x_1 < x_2 < \ldots < x_{i-1} < x_i < \ldots < x_{n-1} < x_n = b$, et c_i est un point arbitraire de l'intervalle $[x_{i-1},\ x_i]$. Une primitive de la fonction $f : x \mapsto f(x)$ est une fonction $F(x)$ telle que $F'(x) = f(x)$. On note $F(x) = \int f(x)dx$.

Si *f* est continue sur $[a,\ b]$ et si *F* est une primitive de *f* sur $[a,\ b]$, alors $\int_a^b f(x)dx = F(b) - F(a)$

Règles :

- $\int kf(x)dx = k\int f(x)dx$ où *k* est une constante
- $\int (f(x) + g(x))dx = \int f(x)dx + \int g(x)dx$
- $\int f^n \left(\frac{df}{dx}\right) dx = \frac{f^{n+1}}{n+1} + C$ où *n* est un nombre rationnel, $n \neq 1$
- Substitution : $\int_a^b f(g(x))g'(x)dx = \int_{g(a)}^{g(b)} f(u)du$
- Intégration par parties $\int udv = uv - \int vdu$

Résultats : (Se reporter à des tables d'intégrales[1] pour d'autres résultats)

- $\int x^n dx = \frac{x^{n+1}}{n+1} + C$ où *n* est un nombre rationnel, $n \neq 1$
- $\int \sin(kx)dx = -\frac{\cos(kx)}{k} + C$
- $\int \cos(kx)dx = \frac{\sin(kx)}{k} + C$
- $\int \sec^2 x dx = \tan x + C$
- $\int \csc^2 x dx = -\cot x + C$
- $\int \sec x \tan x dx = \sec x + C$
- $\int \csc x \cot x dx = -\csc x + C$
- $\int \frac{1}{x} dx = \ln|x| + C$
- $\int \tan x dx = -\ln|\cos x| + C$
- $\int \cot x dx = \ln|\sin x| + C$
- $\int e^x dx = e^x dx$
- $\int \frac{dx}{\sqrt{a^2 - x^2}} = \sin^{-1}\left(\frac{x}{a}\right) + C$
- $\int \frac{dx}{a^2 + x^2} = \frac{1}{a}\tan^{-1}\left(\frac{x}{a}\right) + C$
- $\int \frac{dx}{x\sqrt{x^2 - a^2}} = \frac{1}{a}\sec^{-1}\left|\frac{x}{a}\right| + C$
- $\int \sinh x dx = \cosh x + C$
- $\int \cosh x dx = \sinh x + C$
- $\int \frac{1}{\cosh^2 x} dx = \tanh x + C$
- $\int \frac{1}{\sinh^2 x} dx = -\coth x + C$
- $\int \frac{dx}{\sqrt{a^2 + x^2}} = \sinh^{-1}(\frac{x}{a}) + C$

[1] Consulter par exemple « Standard Mathematical Tables and Formulae, 30th Edition », CRC Press (1995)

1.5 Séries de Taylor

Les fonctions $f: x \mapsto f(x)$ peuvent être approchées par des séries de puissances de *x* appelées séries de Taylor et de Maclaurin. Ces développements en séries de puissances sont très utiles en mathématiques pour déterminer des approximations polynomiales de fonctions, et en physique pour linéariser des phénomènes complexes, par exemple. De nouveau, cette section se veut simplement un rappel des définitions et des résultats importants.

Série de Taylor :
Soit $f: x \mapsto f(x)$ une fonction qui admet des dérivées de tous ordres dans un intervalle contenant le point a comme point intérieur. La série de Taylor de *f* en $x = a$ est

$$f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \ldots + \frac{f^{(n)}(a)}{n!}(x-a)^n + \ldots = \sum_{i=0}^{\infty} \frac{f^{(i)}(a)}{i!}(x-a)^i$$

Série de Maclaurin :
La série de Maclaurin de *f* correspond à la série de Taylor dans le cas particulier $a = 0$:

$$f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \ldots + \frac{f^{(n)}(0)}{n!}x^n + \ldots = \sum_{i=0}^{\infty} \frac{f^{(i)}(0)}{i!}x^i$$

Polynôme de Taylor :
Soit $f: x \mapsto f(x)$ une fonction qui admet des dérivées jusqu'à l'ordre *n* dans un intervalle contenant le point a comme point intérieur. Le polynôme de Taylor d'ordre *n* de *f* en $x = a$ est

$$P_n(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \ldots + \frac{f^{(n)}(a)}{n!}(x-a)^n$$

Le polynôme de Taylor d'ordre *n* est la série de Taylor tronquée à l'ordre *n*. Ce polynôme de Taylor représente la « meilleure » approximation de la fonction *f* en $x = a$ par un polynôme de degré *n*. En particulier, la **linéarisation** de *f* en $x = a$ est

$$P_1(x) = f(a) + f'(a)(x-a)$$

Théorème : Convergence de la série de Taylor :
Si $f: x \mapsto f(x)$ admet des dérivées de tous ordres dans un intervalle contenant le point a comme point intérieur, alors pour tout *x* dans cet intervalle et pour tout entier *n*,

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \ldots + \frac{f^{(n)}(a)}{n!}(x-a)^n + R_n(x)$$

où $R_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!}(x-a)^{n+1}$, *c* étant un certain point compris entre *a* et *x*. On peut donc aussi écrire

$$f(x) = P_n(x) + R_n(x)$$

Ce résultat remarquable permet donc non seulement de générer autour de $x = a$ la « meilleure » approximation polynomiale possible de la fonction *f*, mais aussi de garantir l'erreur $R_n(x)$ associée à cette approximation : cette erreur est de l'ordre de $(x-a)^{n+1}$.

Exemple :
Soit la fonction $f(x) = \sin x$. Déterminons la série de Taylor de *f* en *x*=0 (série de Maclaurin). Nous avons

$$f'(x) = \cos x$$
$$f''(x) = -\sin x$$
$$f'''(x) = -\cos x$$
$$\dots$$
$$f^{(2i)}(x) = (-1)^i \sin x$$
$$f^{(2i+1)}(x) = (-1)^i \cos x$$

Donc :

$$f^{(2i)}(0) = 0$$
$$f^{(2i+1)}(0) = (-1)^i$$

La série de Taylor de $f(x) = \sin x$ en *x*=0 ne contient donc que des puissances impaires ; cette série est

$$x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + \frac{(-1)^n x^{2n+1}}{(2n+1)!} + \dots$$

La figure 1.4 ci-dessous illustre la fonction $f(x) = \sin x$ et sa série de Taylor en *x*=0 tronquée à différents ordres, c'est-à-dire les polynômes de Taylor associés. On peut constater que les polynômes de Taylor fournissent des approximations de plus en plus précises de $f(x) = \sin x$ à mesure que l'ordre de ces polynômes augmente.

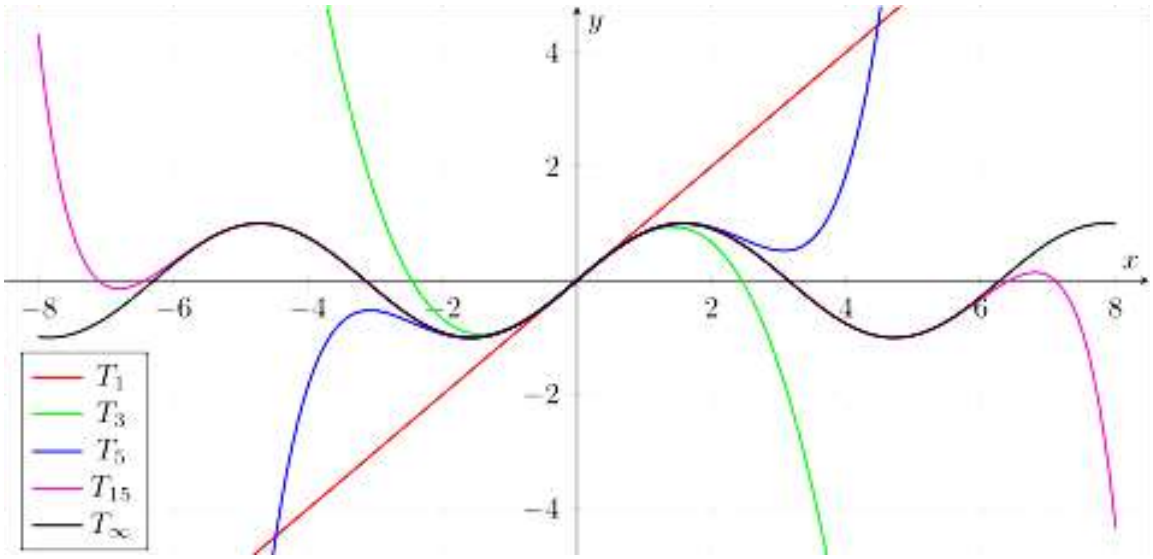


Figure 1.4 : Fonction $f(x) = \sin x$ et divers polynômes de Taylor associés.

Chapitre 2 : Calcul vectoriel intégral

2.1 Introduction

Dans cette section, nous abordons le calcul *intégral* en lien avec les fonctions vectorielles. Ces notions trouvent des applications importantes en Géométrie, en Dynamique énergétique et en Physique en général.

2.2 Fonction vectorielle

2.2.1 Définition

Une fonction vectorielle $\overrightarrow{f(t)}$ d'une seule variable réelle t est définie comme un vecteur de l'espace dont les composantes dépendent de la variable t. La variable de départ est notée ici t, car dans beaucoup de situations, notamment en dynamique, il s'agira du temps. Il faut cependant garder à l'esprit que cette variable est *a priori* quelconque.

En utilisant les composantes de $\overrightarrow{f(t)}$ sur la base orthonormale $(\hat{\boldsymbol{\imath}},\ \hat{\boldsymbol{\jmath}},\ \hat{\boldsymbol{k}})$ du repère cartésien, on définit:

$$\overrightarrow{f(t)} = f_x(t)\hat{\imath} + f_y(t)\hat{\jmath} + f_z(t)\hat{k}$$

$f_x(t)$, $f_y(t)$ et $f_z(t)$ étant 3 fonctions scalaires de *la* variable t.

À titre d'exemple, la position $\overrightarrow{r(t)}$ d'une particule en mouvement dans l'espace est une fonction vectorielle du temps (Figure 2.1)

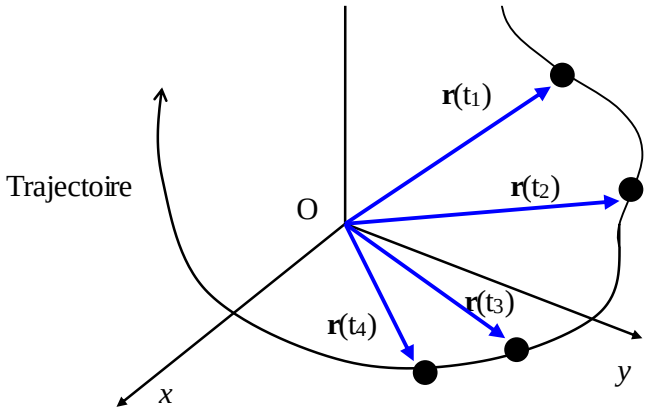


Figure 2.1 : Exemple de fonction vectorielle du temps : le vecteur position $\overrightarrow{r(t)}$ d'une particule

2.2.2 Dérivation

Les règles habituelles du calcul différentiel pour les fonctions *réelles* d'une variable réelle s'étendent aux fonctions *vectorielles* d'une variable réelle, en particulier les règles de dérivation. La fonction vectorielle $\overrightarrow{f(t)}$ est dite **différentiable** au point *t* lorsque la limite suivante existe :

$$\overrightarrow{f'(t)} = \lim_{dt \to 0} \left(\frac{\vec{f}(t+dt) - \vec{f}(t)}{dt} \right)$$

Le vecteur $\overrightarrow{f'}(t)$ est alors appelé la **dérivée** de $\vec{f}(t)$. En pratique, $\overrightarrow{f'}(t)$ est obtenu en dérivant les composantes de $\vec{f}(t)$ dans le repère cartésien:

$$\overrightarrow{f'}(t) = f_x'(t)\hat{\imath} + f_y'(t)\hat{\jmath} + f_z'(t)\hat{k}$$

La dérivée vectorielle est à la base de la définition des vecteurs vitesse et accélération à partir du vecteur *position* d'une particule.

2.3 Courbes dans l'espace et mesures

2.3.1 Courbes dans l'espace

Considérons pour commencer la fonction vectorielle $\overrightarrow{r(t)}$ représentant la position d'un point (Fig. 2.1) dans l'espace et t le temps. En utilisant les composantes du vecteur position $\overrightarrow{r(t)}$ sur la base orthonormale $(\hat{\imath}, \hat{\jmath}, \hat{k})$ du repère cartésien :

$$\vec{r}(t) = x(t)\hat{\imath} + y(t)\hat{\jmath} + z(t)\hat{k}$$

$x(t)$, $y(t)$ et $z(t)$ étant 3 fonctions scalaires de la variable t. Cette expression représente l'équation paramétrée de la trajectoire (*C*) de la particule.

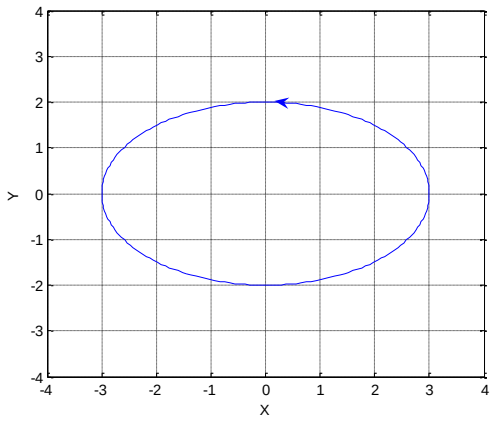
Exemples :

- **Ellipse :** $x(t) = a\cos t; \quad y(t) = b\sin(t), \quad z(t) = 0$

(*b*<*a*) représente une ellipse dans le plan (*x y*), centrée à l'origine, de demi grand axe *a*, demi petit axe *b*.

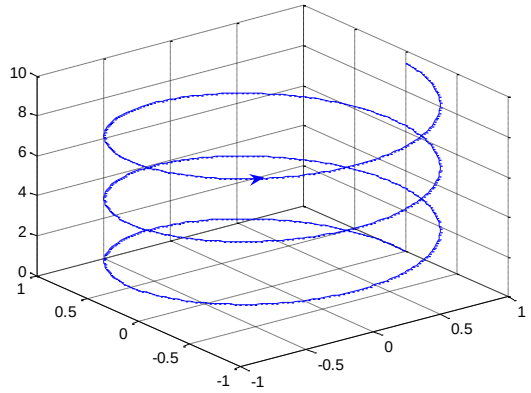
- **Hélice circulaire :** $x(t) = a\cos t; \quad y(t) = a\sin(t); \quad z(t) = ct$

Représente une hélice circulaire de rayon *a*, de pas *c*



Ellipse

$x(t) = 3\cos t; \quad y(t) = 2\sin(t), \quad z(t) = 0$

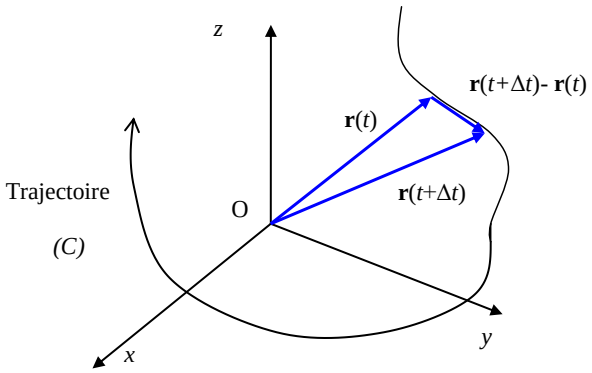


Hélice circulaire

$x(t) = \cos t; \quad y(t) = \sin(t); \quad z(t) = 0.5t$

2.3.2 Tangente à une courbe

La notion de tangente à une courbe dans l'espace est une généralisation directe de celle de la tangente à une courbe dans le plan.



En référence à la figure ci-dessus, le vecteur tangent à la courbe à l'instant t est donné par :

$$\overrightarrow{r'}(t) = x'(t)\hat{\imath} + y'(t)\hat{\jmath} + z'(t)\hat{k}$$

Le vecteur $\hat{u}(t) = \frac{\overrightarrow{r'}(t)}{|\overrightarrow{r'}(t)|}$ est ainsi défini comme le vecteur **unitaire** tangent à la courbe que l'on note $\overrightarrow{dr}$

2.3.3 Longueur d'une courbe

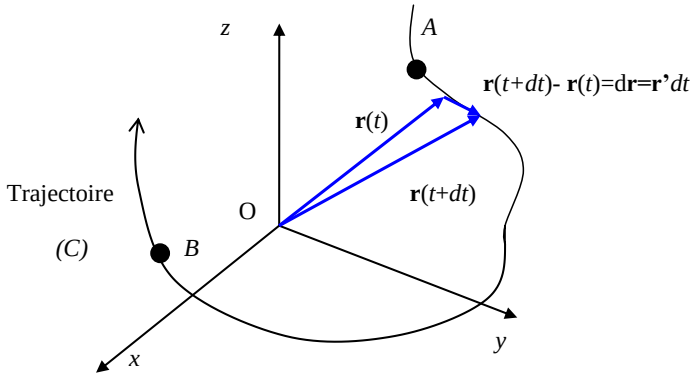
En référence à la figure ci-dessous, la longueur *infinitésimale* de l'élément de courbe (*C*) entre les instants *t* et *t + dt* est :

$$dl = |\vec{r}(t+dt) - \vec{r}(t)| = \left|\overrightarrow{r'(t)}\right| dt$$

Par conséquent, la longueur le (*C*) entre 2 points *A (t = a)* et *B (t=b)* est

$$l = \int_A^B dl = \int_a^b \left|\overrightarrow{r'(t)}\right| dt$$

$$l = \int_a^b \sqrt{x'(t)^2 + y'(t)^2 + z'(t)^2}\, dt$$



Il existe un cas particulier usuel, celui où la courbe est le graphe 2D d'une fonction $g(x)$ sur l'intervalle $x = [a,b]$. Cela revient dans ce cas, à représenter la courbe à l'aide de la fonction vectorielle suivante :

$$\overrightarrow{f(x)} = x\,\hat{\imath} + g(x)\hat{\jmath} + 0\hat{k}$$

En supposant toujours que $x = [a,b]$ désigne l'intervalle sur lequel on recherche la longueur de la courbe, cela revient donc à appliquer la formule précédente (on dérive les trois composantes de la fonction vectorielle par rapport à la variable x) afin d'obtenir :

$$l = \int_A^B dl = \int_a^b \left|\overrightarrow{f'(x)}\right| dx$$

$$l = \int_a^b \sqrt{1 + g'(x)^2}\, dx$$

2.4 Intégrale de ligne d'un champ vectoriel

2.4.1 Définition

Après avoir introduit les notions de champ vectoriel et de courbe dans l'espace, nous allons définir la notion d'*intégrale d'un champ vectoriel le long d'une courbe*. Nous devons pour cela, préalablement nous donner :

- Un champ vectoriel : $\vec{F}(x,y,z) = F_x(x,y,z)\hat{\imath} + F_y(x,y,z)\hat{\jmath} + F_z(x,y,z)\hat{k}$

- Une courbe paramétrée dans l'espace : $\vec{r}(t) = x(t)\hat{\imath} + y(t)\hat{\jmath} + z(t)\hat{k}$

En dynamique de la particule, la courbe représentera la trajectoire d'une particule dans l'espace, et le champ vectoriel représentera la résultante des forces agissant sur cette particule. La figure 2.2 illustre cette situation.

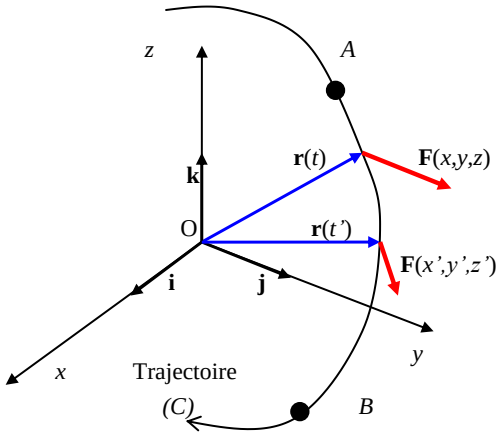


Figure 2.2 : Notion d'intégrale de ligne d'un champ vectoriel

On définit l'intégrale de ligne du champ vectoriel **F** le long de la courbe paramétrée (*C*) entre les points *A (t = a)* et *B (t=b)* :

$$\int_A^B \vec{F}(x,y,z).\ \overrightarrow{dr} = = \int_a^b F_x(t)\,x'(t) + F_y(t)\,y'(t) + F_z(t)\,z'(t)$$

On note aussi $\int_C \vec{F}(x,y,z).\ \overrightarrow{dr}$ pour indiquer que l'intégration se fait sur la courbe (*C*)

C'est la généralisation de l'intégrale $\int_a^b f(x)dx$ pour une fonction réelle d'une variable réelle.

Remarque : On peut obtenir une expression un peu différente de la définition précédente, en remarquant que $\overrightarrow{dr}$ est un vecteur tangent à la courbe (*C*) (voir figure 2.3)

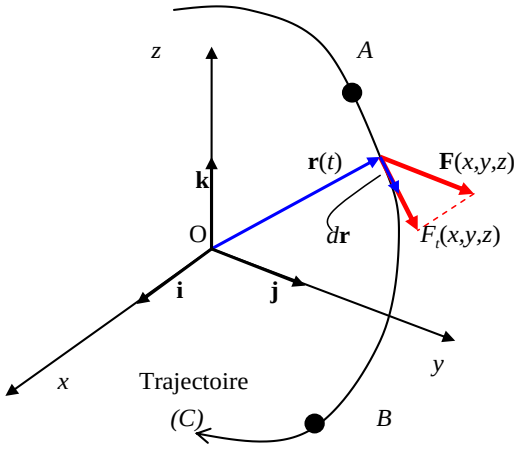


Figure 2.3 : Interprétation de l'intégrale de ligne via la composante tangentielle du champ vectoriel

Alors, le produit scalaire $\mathbf{F}(x,y,z).d\mathbf{r}$ dans l'expression de l'intégrale peut se réécrire

$$\mathbf{F}(x,y,z).d\mathbf{r} = F_t(x,y,z)dr$$

Où $F_t(x,y,z)$ est la composante tangentielle du champ vectoriel **F.** L'intégrale de ligne peut donc s'exprimer sous la forme

$$\int_A^B \mathbf{F}(x,y,z).d\mathbf{r} = \int_A^B F_t(x,y,z).dr$$

2.4.2 Une application importante : le travail d'une force

Dans le cas le plus général, le travail d'une force $\mathbf{F}$ agissant sur une particule qui se déplace dans l'espace le long d'une trajectoire (*C*) entre les points *A (t = a)* et *B (t=b)* correspond à l'intégrale

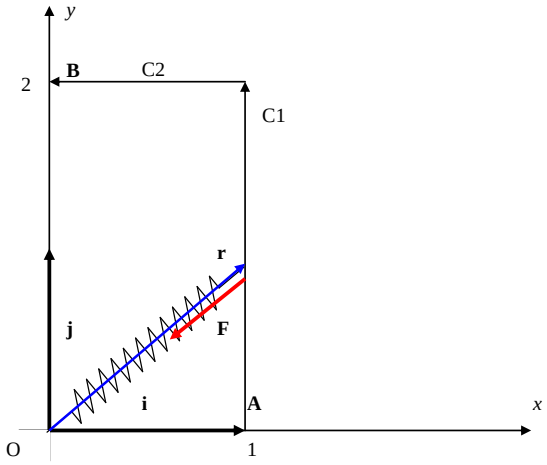
$$\int_A^B \mathbf{F}(x,y,z).d\mathbf{r} \qquad \text{[N.m]}$$

Exemple : Soit un ressort de raideur *k* dont une extrémité est fixée au point O et l'autre extrémité se déplace entre les points $x=1, y=0$ et $x=1, y=2$ en suivant la trajectoire illustrée sur le schéma ci-dessous. On suppose en outre que la longueur libre du ressort est nulle, de sorte que la force de rappel exercée par le ressort est $\mathbf{F} = -k\mathbf{r}$. Calculons le travail de cette force le long du chemin $(C_1)+(C_2)$. L'équation paramétrée des chemins (C_1) et (C_2) est :

$$\begin{cases} (C_1): \boldsymbol{r}(t) = x(t)\boldsymbol{i} + y(t)\boldsymbol{j} = \boldsymbol{i} + t\boldsymbol{j} & 0 \le t \le 2 \\ (C_2): \boldsymbol{r}(t) = x(t)\boldsymbol{i} + y(t)\boldsymbol{j} = (1-t)\boldsymbol{i} + 2\boldsymbol{j} & 0 \le t \le 1 \end{cases}$$

Comme $\mathbf{r}(t) = x(t)\mathbf{i} + y(t)\mathbf{j}$, la force de rappel est $\mathbf{F} = -k\mathbf{r} = -k[x(t)\mathbf{i} + y(t)\mathbf{j}]$. Le travail accompli par $\mathbb{F}$ est donc

$$\int_A^B \boldsymbol{F}(x,y,z).d\boldsymbol{r} = \int_A^B [F_x(t)\dot{x}(t) + F_y(t)\dot{y}(t)]dt = \int_A^B [-kx(t)\dot{x}(t) - ky(t)\dot{y}(t)]dt$$



- Sur (C_1), $x(t) = 1; \dot{x}(t) = 0; y(t) = t; \dot{y}(t) = 1$, donc

$$\int_{C_1} \boldsymbol{F}(x,y,z).d\boldsymbol{r} = \int_0^2 [-kx(t)\dot{x}(t) - ky(t)\dot{y}(t)]dt = \int_0^2 [-kt]dt = \left[-k\frac{t^2}{2}\right]_0^2 = -2k$$

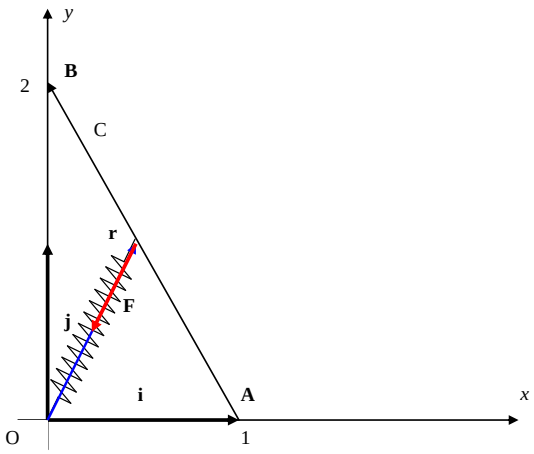
- Sur (C_2), $x(t) = 1 - t; \dot{x}(t) = -1; y(t) = 2; \dot{y}(t) = 0$, donc

$$\int_{C_2} \boldsymbol{F}(x,y,z).d\boldsymbol{r} = \int_0^1 [-kx(t)\dot{x}(t) - ky(t)\dot{y}(t)]dt = \int_0^1 [k(1-t)]dt = \left[-k\frac{(1-t)^2}{2}\right]_0^1 = \frac{k}{2}$$

En bilan,

$$\int_A^B \boldsymbol{F}(x,y,z).d\boldsymbol{r} = -2k + \frac{k}{2} = -3\frac{k}{2}$$

On remarque que le travail effectué par **F** est négatif, ce qui physiquement signifie qu'il faut globalement fournit de l'énergie externe pour amener l'extrémité du ressort de A vers B en suivant le chemin $(C_1) + (C_2)$. On peut ici se demander si le travail calculé dépend du chemin suivi entre les points A et B. Imaginons un autre chemin simple, sous la forme d'un segment de droite entre A et B (Figure ci-dessous)



On peut proposer une équation paramétrée du nouveau chemin (C) :

$$(C): \boldsymbol{r}(t) = x(t)\boldsymbol{i} + y(t)\boldsymbol{j} = (1-t)\boldsymbol{i} + 2t\boldsymbol{j} \qquad 0 \le t \le 1.$$

Remarquons au passage que l'équation paramétrée du chemin n'est pas unique; il existe une infinité d'équations acceptables. À titre d'exemple,

$$\boldsymbol{r}(t) = (1-2t)\boldsymbol{i} + 4t\boldsymbol{j} \qquad 0 \le t \le 0.5$$

est une autre équation paramétrée du même chemin.

Sur (C), $x(t) = 1-t; \dot{x}(t) = -1; y(t) = 2t; \dot{y}(t) = 2$, donc

$$\int_{C_1} \mathbf{F}(x,y,z).d\mathbf{r} \int_0^1 \left[-kx(t)\dot{x}(t) - ky(t)\dot{y}(t)\right]dt$$

$$= \int_0^1 \left[k(1-t) - 4kt\right]dt = \int_0^1 \left[k(1-5t)\right]dt = \left[-k\frac{(1-5t)^2}{10}\right]_0^1 = -16\frac{k}{10} + \frac{k}{10} = -15\frac{k}{10} = -3\frac{k}{2}$$

On note donc que sur cet exemple le travail calculé sur 2 chemins différents (avec les mêmes points de départ et d'arrivée) est identique. Cette propriété se généralise à certaines classes de champs de forces (ou de champs vectoriels) que l'on nomme conservatifs.

Chapitre 3 : Différentiation et intégration numérique

3.1 Introduction

Dans certaines situations, il n'est pas possible d'appliquer les techniques analytiques vues au chapitre 1 pour calculer une dérivée ou une intégrale. On doit alors mettre en œuvre des techniques de *différentiation* et *d'intégration* numérique. Ces techniques sont particulièrement utiles dans le cas d'applications en robotique pour lesquelles les données sont échantillonnées, c'est à dire que l'on ne dispose que de valeurs numériques à des instants donnés.

3.2 Différentiation numérique

La différentiation numérique permet de calculer une dérivée de manière approchée. Dans certaines circonstances, la fonction que l'on souhaite dériver peut avoir une expression très compliquée. Il peut même arriver que l'on ne dispose pas d'une expression analytique de cette fonction, et que l'on connaisse simplement la valeur de cette fonction en certains points seulement. Un tel cas se produit typiquement lorsque la fonction représente des points expérimentaux. Imaginons par exemple la chute libre d'une bille, dont on veut mesurer la vitesse v(t) à divers instants t, à partir de photographies de la position instantanée *x* de cette bille à ces instants.

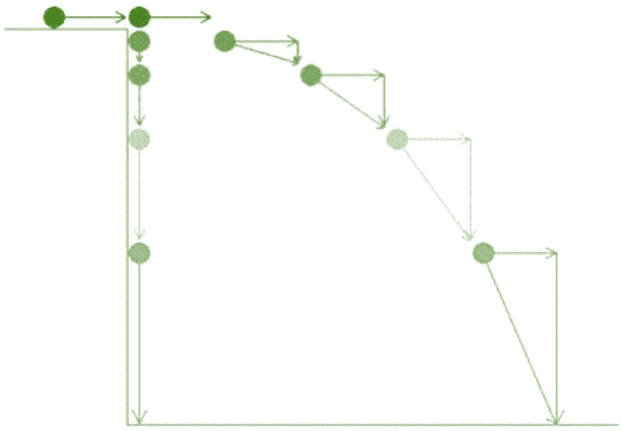


Figure 3.1 : Positions successives d'une bille en chute libre

Dans ce cas la fonction $x(t)$ n'est pas connue analytiquement; on dispose simplement de valeurs expérimentales de la position à certains instants précis du mouvement. Nous allons voir trois approches pour obtenir de façon approchée la vitesse dans ce cas (et une dérivée quelconque dans le cas général) : ce sont les formules de *dérivation-avant, dérivation-arrière, dérivation-centrée.*

3.2.1 Dérivation-avant et dérivation-arrière

Par définition (cf Chapitre 1), on sait que la dérivée d'une fonction $f(x)$ différentiable est définie par :

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Ceci suggère une approximation de$f'(x)$ dans le cas où l'on choisit le paramètre h qui n'est pas nul :

$$f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

à condition que le paramètre *h* que l'on appelle le **pas d'intégration (**ou plus simplement le **pas)** soit suffisamment petit (tout en restant positif). Cette formule s'interprète comme la pente de la droite joignant les points de coordonnées $(x, f(x))$ et $(x+h, f(x+h))$, et porte le nom de formule de **dérivation-avant**.

On peut également obtenir une approximation de $f'(x)$ en faisant la même opération de l'autre côté de la courbe, c'est-à-dire en considérant la pente de la droite joignant les points de coordonnées $(x, f(x))$ et $(x-h, f(x-h))$. c'est la formule de **dérivation-arrière** :

$$f'(x) \approx \frac{f(x) - f(x-h)}{h}$$

Dans ces formules, *h*.

En pratique, on utilisera plutôt la notation discrète correspondant à des intervalles réguliers. Pour cela, si on considère une suite de points $[x_0, \dots, x_n]$ espacés d'un pas h et si on connait la valeur de la fonction $f(x)$ en ces différents points, cad que le vecteur $[f(x_0), \dots, f(x_n)]$ est connu, alors on peut exprimer les **dérivées avant et arrière** de la manière suivante :

$$f'(x_k) \approx \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k} = \frac{f(x_{k+1}) - f(x_k)}{h} \qquad \text{(dérivation avant)}$$

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}} = \frac{f(x_k) - f(x_{k-1})}{h} \qquad \text{(dérivation arrière)}$$

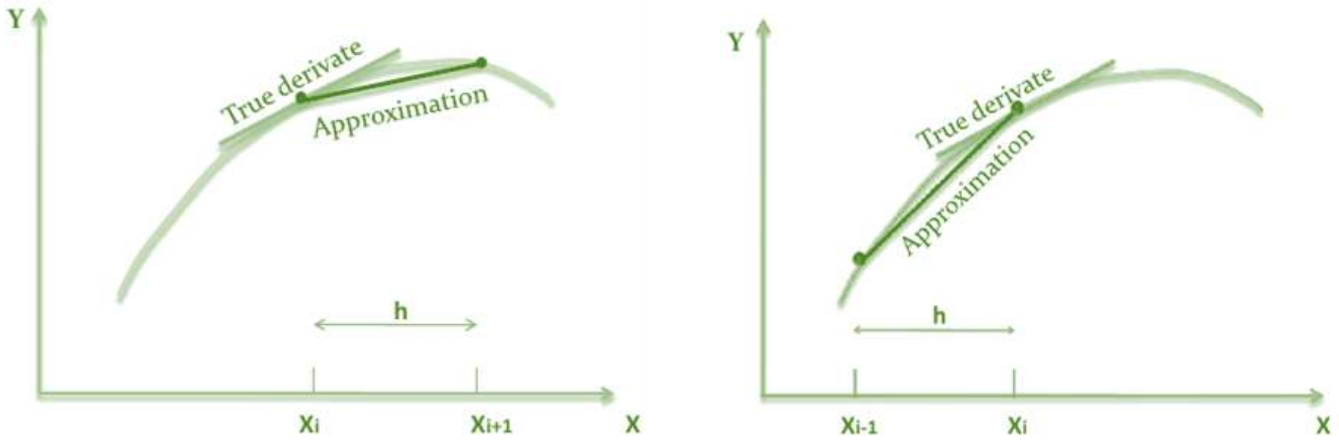


Figure 3.2 : Différentiation numérique avant (gauche) et arrière (droite)

Pour obtenir une bonne précision sur le calcul de la dérivée, il est nécessaire de choisir un pas *h* suffisamment petit. Une question très importante dans ces formules est de déterminer comment la précision de l'approximation dépend de *h*. On peut le savoir en effectuant un développement en série de Taylor de *f* au point *x*,

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \ldots + \frac{h^n}{n!} f^{(n)}(x) + \ldots,$$

d'où l'on tire

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2} f''(x) - \ldots - \frac{h^{n-1}}{n!} f^{(n)}(x) + \ldots$$

On déduit donc que l'erreur commise en approchant $f'(x)$ par la formule de dérivation-avant $\frac{f(x+h) - f(x)}{h}$ est :

$$f'(x) - \frac{f(x+h) - f(x)}{h} = -\frac{h}{2} f''(x) - \ldots - \frac{h^{n-1}}{n!} f^{(n)}(x) + \ldots$$

Dans la mesure où *h* est suffisamment petit, le premier terme est dominant dans la somme au membre de droite,

$$f'(x) - \frac{f(x+h) - f(x)}{h} \approx -\frac{h}{2} f''(x)$$

Bien entendu, il n'est pas possible en général de connaître l'erreur d'approximation $-\frac{h}{2} f''(x)$ car $f''(x)$ n'est pas connu. L'information importante cependant est de savoir que cette erreur *est proportionnelle à h.* Dans le cas de la formule de dérivation-arrière, il est facile de montrer que l'erreur est

$$f'(x) - \frac{f(x) - f(x-h)}{h} \approx \frac{h}{2} f''(x)$$

Retenons que :

L'erreur dans les méthodes de dérivation-avant et de dérivation-arrière est **proportionnelle au pas *h*.**

Ceci signifie par exemple que l'erreur est divisée par 2 lorsque le pas *h* est divisé par 2.

Exemple :

Soit la fonction $f(x) = e^x$. On sait dans ce cas calculer $f'(x) = e^x$. L'approximation de $f'(x) = e^x$ par la formule de dérivation-avant fournit :

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} = \frac{e^{x+h} - e^x}{h} = \frac{e^h - 1}{h} e^x$$

L'approximation de $f'(x) = e^x$ par la formule de dérivation-arrière fournit quant à elle :

$$f'(x) \approx \frac{f(x) - f(x-h)}{h} = \frac{e^x - e^{x-h}}{h} = \frac{1 - e^{-h}}{h} e^x$$

Les résultats des deux méthodes et leur comparaison avec la dérivée exacte sont illustrés ci-dessous pour $0 \leq x \leq 4$, et pour *h*=0.2 (21 points de calcul) ; sur ce graphique on reporte la dérivée exacte $f'(x) = e^x$, et les approximations de cette dérivée fournies par les formules de dérivation avant et arrière, pour diverses valeurs de *x*. On peut remarquer que la dérivation avant surestime la dérivée exacte dans ce cas, alors que la dérivation arrière sous-estime la dérivée exacte (justifier cette observation à partir de la formule de l'erreur de ces deux méthodes), les deux méthodes offrant des précisions comparables

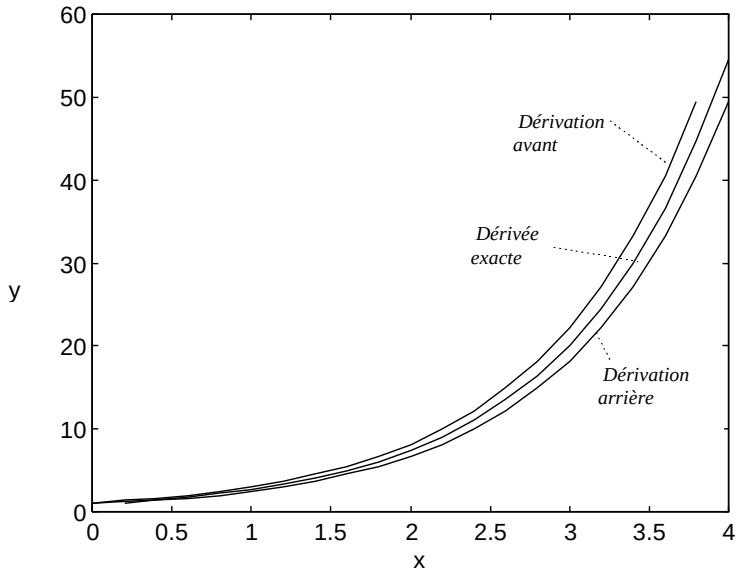


Figure 3.3 : Différentiation numérique de $f(x) = e^x$ par dérivation avant et dérivation arrière

3.2.2 Dérivation centrée

La formule de **dérivation centrée** consiste à utiliser l'approximation suivante:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

Cette formule s'interprète comme la pente de la droite joignant les points $(x-h, f(x-h))$ et $(x+h, f(x+h))$ (Figure 3.4). La notation discrète équivalente est donnée par :

$$f'(x_k) \approx \frac{f(x_{k+1}) - f(x_{k-1})}{x_{k+1} - x_{k-1}} \approx \frac{f(x_{k+1}) - f(x_{k-1})}{2h}$$

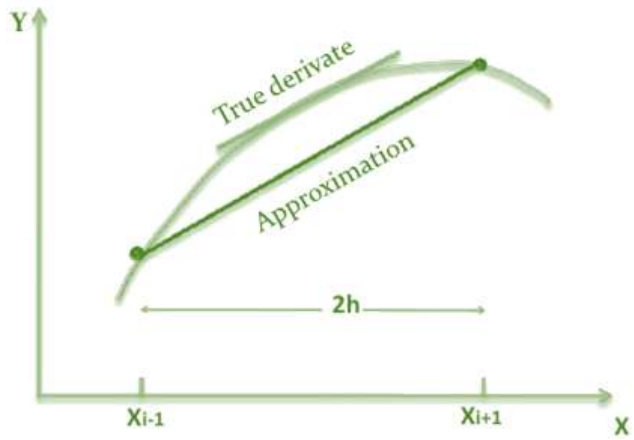


Figure 3.4 : Différentiation numérique par dérivation centrée

Cette formulation est plus utilisée en pratique que les formules de dérivation avant et arrière car elle offre une **précision supérieure**. En effet, la série de Taylor de *f* en *x* donne

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \frac{h^3}{6} f^{(3)}(x) \ldots + \frac{h^n}{n!} f^{(n)}(x) + \ldots$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2} f''(x) - \frac{h^3}{6} f^{(3)}(x) \ldots + (-1)^n \frac{h^n}{n!} f^{(n)}(x) + \ldots$$

En faisant la différence de ces deux expressions,

$$f(x+h) - f(x-h) = 2hf'(x) + \frac{h^3}{3} f^{(3)}(x) \ldots + 2\frac{h^{2n+1}}{(2n+1)!} f^{(2n+1)}(x) + \ldots$$

d'où : $\frac{f(x+h)-f(x-h)}{2h} = f'(x) + \frac{h^2}{6} f^{(3)}(x) \ldots + \frac{h^{2n}}{(2n+1)!} f^{(2n+1)}(x) + \ldots$

L'erreur d'approximation est cette fois proportionnelle à h^2, lorsque *h* est petit,

$$f'(x) - \frac{f(x+h)-f(x-h)}{2h} \approx -\frac{h^2}{6} f^{(3)}(x)$$

Retenons que :

L'erreur dans la méthode de dérivation centrée est **proportionnelle au carré du pas *h*.**

Ceci implique que la méthode de dérivation centrée est plus précise que les méthodes de dérivation avant ou arrière lorsque le pas est petit. De plus, l'erreur dans la méthode de dérivation centrée diminue rapidement avec le pas : par exemple, l'erreur est divisée par 4 lorsque le pas *h* est divisé par 2.

Exemple :

Reprenons l'exemple de la fonction $f(x) = e^x$. On sait que $f'(x) = e^x$. L'approximation de $f'(x) = e^x$ par la formule de dérivation centrée fournit :

$$f'(x) \approx \frac{f(x+h)-f(x-h)}{2h} = \frac{e^{x+h}-e^{x-h}}{2h} = \frac{e^h - e^{-h}}{2h} e^x$$

Les résultats de cette méthode et sa comparaison avec la dérivée exacte sont illustrés ci-dessous pour $0 \leq x \leq 4$, et pour *h*=0.2 (21 points de calcul) ; la formule de dérivation centrée est beaucoup plus précise que les formules de dérivation avant et arrière.

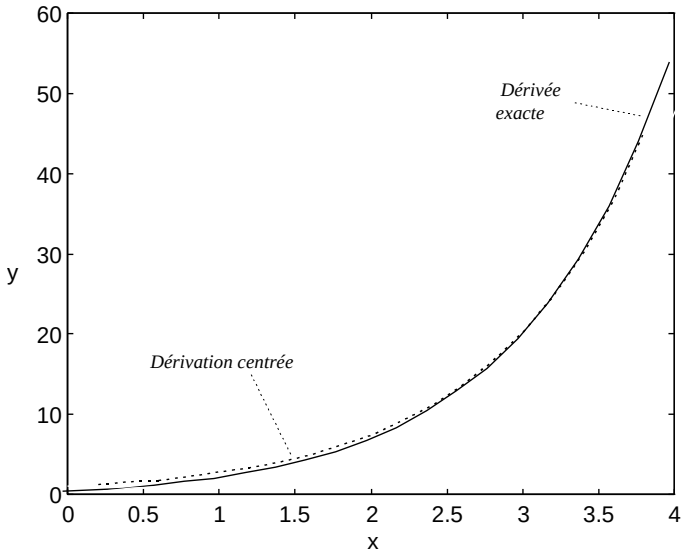


Figure 3.5 : Différentiation numérique de $f(x) = e^x$ par dérivation centrée

3.2.3 Approximation de la dérivée seconde

On peut utiliser les mêmes principes pour obtenir une approximation de la dérivée *seconde* (ex : estimation de l'accélération à partir de la position). Reprenons les séries de Taylor :

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \frac{h^3}{6} f^{(3)}(x) \ldots + \frac{h^n}{n!} f^{(n)}(x) + \ldots$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2} f''(x) - \frac{h^3}{6} f^{(3)}(x) \ldots + (-1)^n \frac{h^n}{n!} f^{(n)}(x) + \ldots$$

et effectuons cette fois-ci la somme de ces deux expressions :

$$f(x+h) + f(x-h) = 2f(x) + h^2 f''(x) + \ldots + 2\frac{h^{2n}}{(2n)!} f^{(2n)}(x) + \ldots$$

d'où l'on tire :

$$f''(x) = \frac{f(x+h) + f(x-h) - 2f(x)}{h^2} - 2\frac{h^2}{4!} f^{(4)}(x) \ldots - 2\frac{h^{2n-2}}{(2n)!} f^{(2n)}(x) + \ldots$$

Ceci suggère l'approximation suivante de la dérivée seconde $f''(x)$:

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

L'équivalent avec la notation discrète est ainsi :

$$f''(x_k) = \frac{f(x_{k+1}) - 2f(x_k) + f(x_{k-1})}{h^2}$$

On sait de plus que l'erreur de cette approximation est proportionnelle à h^2 lorsque *h* est petit, et vaut $f''(x) - \frac{f(x+h) + f(x-h) - 2f(x)}{h^2} \approx -\frac{h^2}{12} f^{(4)}(x)$.

3.2.4 Remarques sur la différentiation numérique

Les méthodes de différentiation numérique doivent être utilisées avec précaution, spécialement dans le cas de fonctions représentant des points expérimentaux. Dans ce cas en effet, la taille du pas *h* est souvent difficile à contrôler, ce qui peut conduire à de larges erreurs sur l'estimation de la dérivée. Il sera alors préférable de construire une fonction d'approximation discrète passant « au mieux » par les points expérimentaux (cf Chap. 7), puis de dériver analytiquement ou numériquement cette fonction d'interpolation.

3.3 Intégration numérique

L'intégration numérique permet de calculer de façon approchée l'intégrale définie $\int_a^b f(x)dx$. En pratique, l'intégration numérique est beaucoup plus utilisée que la différentiation numérique car dans de nombreux cas, $f(x)$ ne possède pas de primitive analytique (par exemple, $f(x) = e^{-x^2}$, $f(x) = \sqrt{1+x^4}$ ne possèdent pas de primitive analytique).

Dans d'autres cas, $f(x)$ peut représenter des points expérimentaux pour des valeurs de *x* précises, et n'a donc pas d'expression analytique. Dans ces situations, le recours à des méthodes d'intégration numérique est nécessaire. Il existe de très nombreuses méthodes d'intégration numérique plus ou moins sophistiquées et précises, qui sont mises en œuvre dans des programmes informatiques ; nous verrons ici les deux méthodes les plus simples : *la méthode des trapèzes et la méthode de Simpson*.

3.3.1 Méthode des rectangles

Une première méthode d'intégration numérique très rudimentaire est la méthode des *rectangles*, qui dérive de la définition même de la notion d'intégrale définie et des sommes de Riemann. Rappelons en effet que l'intégrale $\int_a^b f(x)dx$ est définie comme

$$\int_a^b f(x)dx = \lim_{\max(x_i - x_{i-1})\to 0} \sum_{i=1}^{n} f(c_i)(x_i - x_{i-1})$$

où $x_0 = a < x_1 < x_2 < \ldots < x_{i-1} < x_i < \ldots < x_{n-1} < x_n = b$, et c_i est un point arbitraire de l'intervalle $[x_{i-1},\ x_i]$ (voir figure 3.6).

Ceci suggère une approximation de $\int_a^b f(x)dx$ en effectuant la somme finie :

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} f(c_i)(x_i - x_{i-1})$$

à condition que *n* soit suffisamment grand. Ceci consiste strictement à approcher l'intégrale définie par sa somme de Riemann. L'approximation par la méthode dite des **rectangles** consiste habituellement à choisir tous les intervalles $[x_{i-1},\ x_i]$ de même longueur *h* et à choisir c_i comme le centre de l'intervalle$[x_{i-1},\ x_i]$.

Dans ce cas, on obtient la formule suivante pour la méthode des rectangles:

$$\int_a^b f(x)dx \approx h\,(f(a) + f(x_1) + f(x_2) + \cdots + f(x_{n-1}))$$

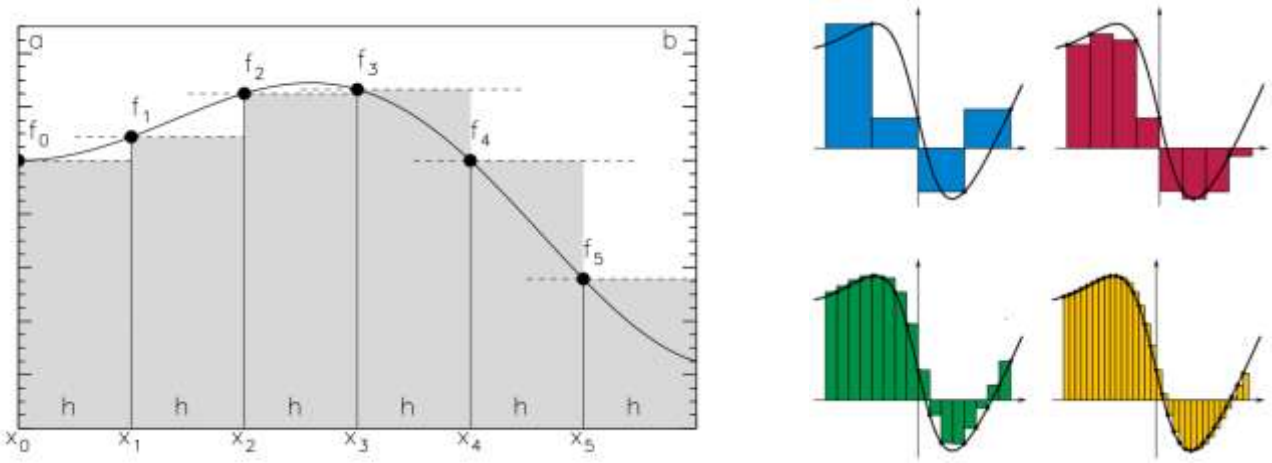


Figure 3.6 : Méthode des rectangles avec un pas constant h et illustration de la convergence pour h-> 0

3.3.2 Méthode des trapèzes

La méthode des trapèzes consiste plutôt à sommer l'aire des trapèzes élémentaires formés des points $(x_i,0)$, $(x_{i+1},0)$, $(x_i, f(x_i))$ et $(x_{i+1}, f(x_{i+1}))$ (voir figure 3.7). Si les intervalles $[x_i, x_{i+1}]$ sont de même longueur *h*, $x_{i+1} - x_i = \frac{b-a}{n} = h$, l'aire de ces trapèzes élémentaires est $\frac{f(x_i) + f(x_{i+1})}{2} h$. En faisant la somme de ces aires élémentaires,

$$\int_a^b f(x)dx \approx \frac{f(a)+f(x_1)}{2}h + \frac{f(x_1)+f(x_2)}{2}h + \ldots + \frac{f(x_i)+f(x_{i+1})}{2}h + \ldots + \frac{f(x_{n-2})+f(x_{n-1})}{2}h + \frac{f(x_{n-1})+f(b)}{2}h$$

c'est-à-dire :

$$\int_a^b f(x)dx \approx \frac{h}{2}\,(f(a) + 2f(x_1) + 2f(x_2) + \cdots + 2f(x_{n-1}) + f(b))$$

C'est la méthode des **trapèzes**. Retenons que dans cette formule, $x_{i+1} - x_i = \frac{b-a}{n} = h$ est appelé le **pas d'intégration.**

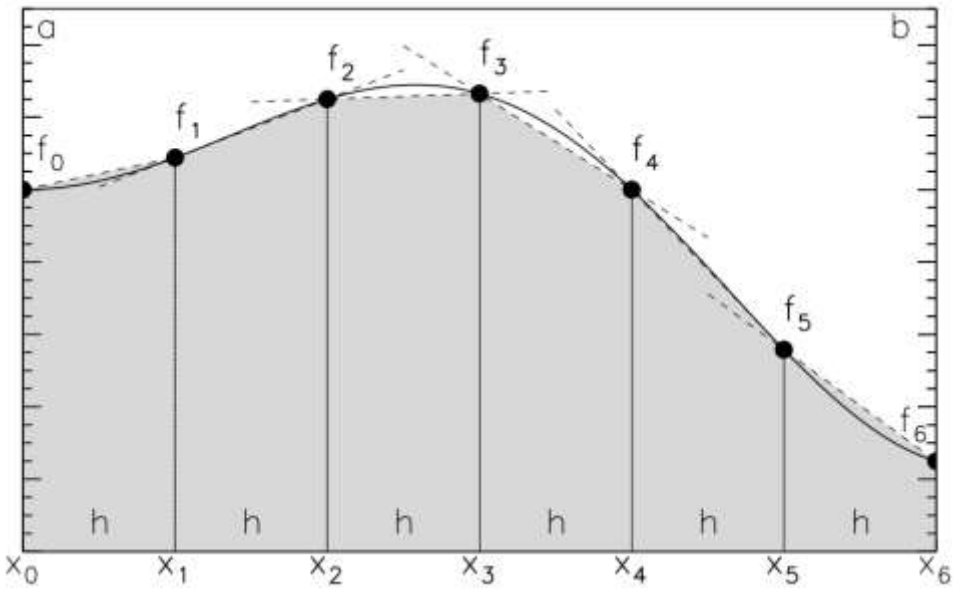


Figure 3.7 : Méthode des trapèzes (ici le pas est constant et fixé à h)

Exemple :

Soit à évaluer l'intégrale $\int_0^1 \sin(\pi x)dx$. Dans ce cas, on dispose d'une solution analytique, $\int_0^1 \sin(\pi x)dx = \left[-\frac{1}{\pi}\cos(\pi x)\right]_0^1 = \frac{2}{\pi} = 0.63662$ avec 5 chiffres significatifs. Comparons cette solution exacte avec l'approximation obtenue par la méthode des trapèzes. Le tableau suivant expose les résultats obtenus pour diverses valeurs du pas d'intégration *h*. L'erreur est définie comme la différence entre la valeur exacte et la valeur approchée.

Pas h	0.1	0.04	0.02	0.01
approximation par trapèzes	0.63138	0.63578	0.63641	0.63657
erreur	0.00524	0.00084	0.00021	0.00005

De ces résultats, nous pouvons faire une observation intéressante : l'erreur de la méthode des trapèzes semble proportionnelle au carré du pas d'intégration *h* (en effet, en passant de *h*=0.04 à *h*=0.02, on réduit l'erreur d'un facteur 4, de même qu'en passant de *h*=0.02 à *h*=0.01). Nous allons voir dans la prochaine section que cette observation peut être généralisée.

3.3.3 Contrôle de l'erreur dans la méthode des trapèzes

La méthode des trapèzes à l'avantage d'être très simple à mettre en œuvre ; elle n'implique que des additions et des multiplications, ce qui est particulièrement intéressant du point de vue de la programmation sur un ordinateur. Une question très importante dans toute méthode numérique d'approximation est l'évaluation de l'erreur commise par le schéma d'approximation. Bien sûr, cette erreur est rarement accessible (il faudrait connaître la valeur exacte de l'intégrale, ce qui est bien souvent impossible), cependant il peut être intéressant d'en trouver un majorant, et surtout de savoir comment cette erreur dépend de la taille du pas d'intégration *h*.

On définit l'erreur de la méthode des trapèzes par

$$\varepsilon_T = \int_a^b f(x)dx - \frac{h}{2}\left[f(a) + 2f(x_1) + 2f(x_2) + \ldots + 2f(x_{n-1}) + f(b)\right].$$

Théorème : Si f'' est continue sur $[a,b]$, alors il existe $c \in [a,b]$ tel que : $\varepsilon_T = -h^2 \frac{b-a}{12} f''(c)$

Ce résultat confirme l'observation faite précédemment sur l'exemple : l'erreur d'approximation de la méthode des trapèzes est proportionnelle au carré du pas d'intégration. Ceci garantira par exemple que pour diviser par 100 l'erreur commise, il suffira de choisir un pas d'intégration 10 fois plus petit (dans la limite où *h* est assez petit). De plus, il peut être intéressant de borner l'erreur ε_T. Il est naturel de s'intéresser à la valeur absolue de l'erreur,

$$|\varepsilon_T| = h^2 \frac{b-a}{12} |f''(c)|$$

Si l'on peut trouver un majorant M_T de $|f''(c)|$ sur l'intervalle $[a,b]$, alors l'erreur $|\varepsilon_T|$ est inférieure à $h^2 \frac{b-a}{12} M_T$. Le résultat est donc

Borne supérieure de l'erreur de la méthode des trapèzes :

Si l'on peut trouver M_T tel que $\forall c \in [a,b], |f''(c)| \leq M_T$ alors on peut borner l'erreur de la méthode des trapèzes,

$$|\varepsilon_T| \leq h^2 \frac{b-a}{12} M_T$$

Exemple :

Considérons à nouveau l'exemple précédent $\int_0^1 \sin(\pi x)dx$, et essayons d'estimer une borne supérieure de l'erreur de la méthode des trapèzes en fonction du pas d'intégration *h*. Ici, $f(x) = \sin(\pi x)$ donc $f''(x) = -\pi^2 \sin(\pi x)$. On peut facilement trouver un majorant de $|f''(x)|$ sur l'intervalle [0, 1],

$$|f''(x)| = \pi^2 |\sin(\pi x)| \leq \pi^2$$

Donc d'après le théorème précédent, l'erreur de la méthode des trapèzes peut être bornée,

$$|\varepsilon_T| \leq h^2 \frac{\pi^2}{12}$$

Reprenons le tableau de l'erreur réelle obtenu précédemment, et comparons l'erreur réelle $|\varepsilon_T|$ à son majorant $h^2 \frac{\pi^2}{12}$:

h	0.1	0.04	0.02	0.01
approximation par trapèzes	0.63138	0.63578	0.63641	0.63657
$\lvert\varepsilon_T\rvert$	0.00524	0.00084	0.00021	0.00005
$h^2 \frac{\pi^2}{12}$	0.00822	0.00132	0.00032	0.00008

On constate que $h^2 \frac{\pi^2}{12}$ est effectivement un majorant de $|\varepsilon_T|$ tout en donnant l'ordre de grandeur acceptable de l'erreur réelle. Ce majorant, qui peut être évalué assez simplement (à condition de pouvoir trouver un majorant de $|f''(c)|$ sur l'intervalle $[a,b]$), peut être utilisé pour calculer un pas d'intégration garantissant une erreur inférieure à une valeur prescrite.

Par exemple, pour garantir une erreur inférieure à 10^{-4}, $|\varepsilon_T| \leq 10^{-4}$, il suffira d'imposer $h^2 \frac{\pi^2}{12} \leq 10^{-4}$, c'est-à-dire $h \leq \frac{\sqrt{12 \times 10^{-4}}}{\pi} = 0.011$.

On voit donc que la méthode des trapèzes nécessite un pas d'intégration *h* assez petit (donc beaucoup de calculs) pour obtenir une approximation précise. Il existe une autre méthode d'intégration numérique dont la précision est supérieure, c'est-à-dire qui garantit une erreur moindre avec le même pas d'intégration : c'est la méthode de Simpson.

3.3.4 Méthode de Simpson

L'idée de base de la méthode de Simpson est de remplacer les fonctions d'approximation linéaires de la méthode des trapèzes par des fonctions quadratiques (polynômes de degré 2). La fonction à intégrer est donc approchée par une série d'arcs de paraboles plutôt que par une série de segments de droites. La différence, c'est qu'il faut 3 points pour définir une parabole unique, alors que 2 points suffisent pour définir droite unique. Les arcs de parabole seront donc définis pour 3 points consécutifs x_{i-1}, x_i, x_{i+1}, comme indiqué sur le schéma ci-dessous.

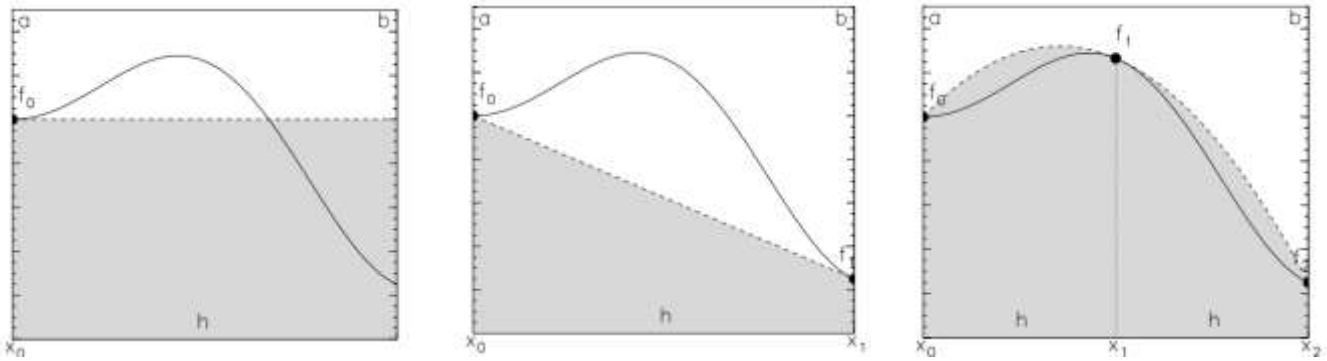


Figure 3.8 : Approximations élémentaire pour les méthodes des rectangles (gauche), des trapèze (milieu) et de Simpson (droite)

La méthode de Simpson conduit donc à approcher $\int_{x_{i-1}}^{x_{i+1}} f(x)dx$ par l'aire comprise sous l'arc de parabole passant par les 3 points $(x_{i-1}, f(x_{i-1}))$, $(x_i, f(x_i))$, $(x_{i+1}, f(x_{i+1}))$. Cet arc de parabole a pour équation :

$$y(x) = ax^2 + bx + c$$

où *a*, *b* et *c* sont trois paramètres à déterminer. À ce stade-ci, on peut effectuer un changement d'origine pour simplifier les calculs en posant

$$x_{i-1} = -h; \quad x_i = 0; \quad x_{i+1} = h$$

(Ce changement d'origine est sans conséquence sur la valeur de l'aire sous l'arc de parabole). Pour déterminer *a*, *b* et *c*, imposons à la parabole de passer par les points $(x_{i-1} = -h, f(x_{i-1}))$, $(x_i = 0, f(x_i))$, $(x_{i+1} = h, f(x_{i+1}))$:

$$\begin{cases} f(x_{i-1}) = ah^2 - bh + c \\ f(x_i) = c \\ f(x_{i+1}) = ah^2 + bh + c \end{cases} \quad \text{d'où} \quad \begin{cases} a = \dfrac{f(x_{i-1}) + f(x_{i+1}) - 2f(x_i)}{2h^2} \\ b = \dfrac{f(x_{i+1}) - f(x_{i-1})}{2h} \\ c = f(x_i) \end{cases}$$

On peut alors calculer l'aire sous l'arc de parabole,

$$\int_{-h}^{h} \left(ax^2 + bx + c\right)dx = \frac{2}{3}ah^3 + 2ch = \frac{h}{3}\left(f(x_{i-1}) + 4f(x_i) + f(x_{i+1})\right)$$

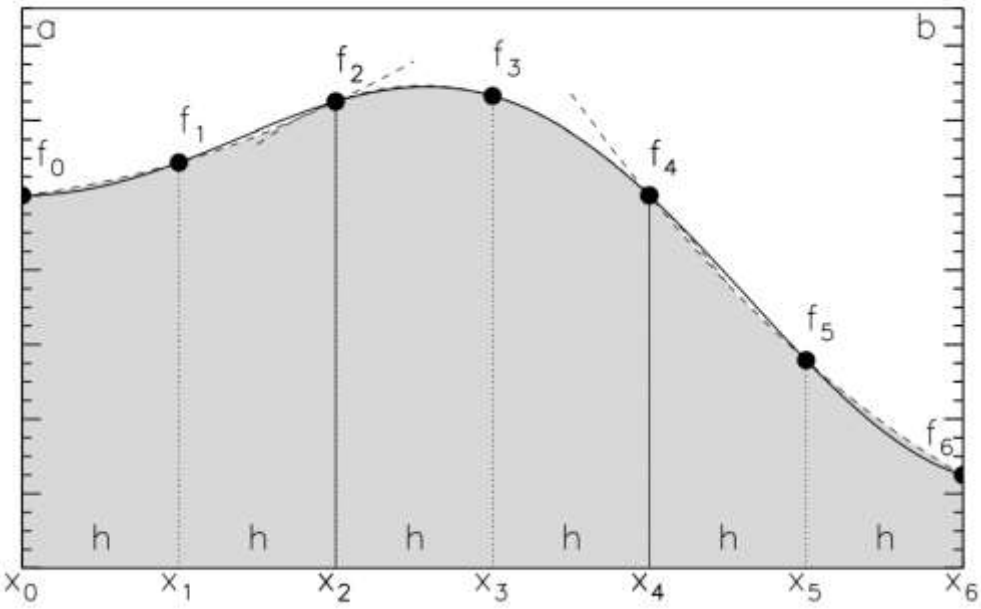


Figure 3.9 : Méthode de Simpson

En répétant ce processus pour les triplets de points $x_0 = a, x_1, x_2$, puis x_2, x_3, x_4, puis x_4, x_5, x_6 jusqu'à $x_{n-2}, x_{n-1}, x_n = b$ et en sommant les aires élémentaires des paraboles, on obtient

$$\int_a^b f(x)dx \approx \frac{h}{3}\left(f(x_0)+4f(x_1)+f(x_2)\right)+\frac{h}{3}\left(f(x_2)+4f(x_3)+f(x_4)\right)+....+\frac{h}{3}\left(f(x_{i-1})+4f(x_i)+f(x_{i+1})\right)$$

$$+...+\frac{h}{3}\left(f(x_{n-2})+4f(x_{n-1})+f(x_n)\right)$$

c'est-à-dire :

$$\int_a^b f(x)dx \approx \frac{h}{3}\left(f(a)+4f(x_1)+2f(x_2)+4f(x_3)+2f(x_4)+\cdots+2f(x_{n-2})+4f(x_{n-1})+f(b)\right)$$

C'est la formule de **Simpson**. Remarquons que dans la méthode de Simpson, le nombre d'intervalles considéré *n* doit être pair.

Exemple : Considérons à nouveau l'exemple $\int_0^1 \sin(\pi x)dx$ .Rappelons que l'on dispose dans ce cas d'une valeur exacte de l'intégrale, $\int_0^1 \sin(\pi x)dx = \left[-\frac{1}{\pi}\cos(\pi x)\right]_0^1 = \frac{2}{\pi} = 0.6366198$.

Estimons cette intégrale à l'aide de la méthode de Simpson, et comparons avec la méthode des trapèzes pour divers pas d'intégration *h*. Le tableau suivant reporte les résultats obtenus; en indiquant les erreurs de la méthode des trapèzes ($|\varepsilon_T|$)et de la méthode de Simpson ($|\varepsilon_S|$)

h	0.1	0.04	0.02	0.01
trapèzes	0.63138	0.63578	0.63641	0.63657
$\lvert\varepsilon_T\rvert$	0.00524	0.00084	0.00021	0.00005
Simpson	0.6366546		0.6366198	0.6366198
$\lvert\varepsilon_S\rvert$	0.0000349		0.0000001	0.0000000

La méthode de Simpson s'avère beaucoup plus précise que la méthode des trapèzes.

3.3.5 Contrôle de l'erreur dans la méthode de Simpson

On définit l'erreur de la méthode de Simpson,

$$\varepsilon_S = \int_a^b f(x)dx - \frac{h}{3}\left[f(a) + 4f(x_1) + 2f(x_2) + \ldots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(b)\right].$$

On dispose du résultat suivant (on ne le démontrera pas)

Borne supérieure de l'erreur de la méthode de Simpson :
Si l'on peut trouver M_S tel que $\forall c \in [a,b], \left|f^{(4)}(c)\right| \le M_S$ alors on peut borner l'erreur de la méthode de Simpson par :

$$\left|\varepsilon_S\right| \le h^4 \frac{b-a}{180} M_S$$

Exemple : Considérons à nouveau l'exemple précédent $\int_0^1 \sin(\pi x)dx$, et essayons d'estimer une borne supérieure de l'erreur de la méthode de Simpson en fonction du pas d'intégration *h*. Ici, $f(x) = \sin(\pi x)$ donc $f^{(4)}(x) = \pi^4 \sin(\pi x)$. On peut facilement trouver un majorant de $\left|f^{(4)}(x)\right|$ sur l'intervalle [0,1],

$$\left|f^{(4)}(x)\right| = \pi^4 \left|\sin(\pi x)\right| \le \pi^4$$

Donc d'après le théorème précédent, l'erreur de la méthode de Simpson pour un pas d'intégration *h* peut être bornée par

$$\left|\varepsilon_S\right| \le h^4 \frac{\pi^4}{180}$$

3.3.6 Remarques sur l'intégration numérique

Bien évidemment, on peut construire des méthodes composites d'ordres plus élevés. Les méthodes utilisant des polynômes de degré plus élevé (tout en restant < 8) sont plus précises à nombre de points égal. Plus précisément, l'erreur associée à l'utilisation d'un polynôme de degré p décroît en n (p + 1) si p est impair et n (p + 2) si p est pair. Autrement dit, l'ordre des méthodes associées est p + 1 si p est impair et p + 2 si p est pair

La précision de toutes les méthodes de Newton-Cote augmente avec le nombre de points utilisés. Tout est donc question de besoin en termes de précision et de temps de calcul. Pour des applications embarquées, on utilise généralement des méthodes rapides (rectangle ou trapèzes) avec un plus grand nombre d'intervalles pour accélérer les calculs.

Numériquement, chaque addition génère une petite erreur d'arrondi machine (l'erreur relative est de l'ordre de 10^{-17} pour des réels doubles précision). Lorsque l'on somme beaucoup de nombres, les erreurs de chaque addition s'ajoutent et l'erreur relative totale augmente. Dans le cas des méthodes d'intégration d'ordre élevé, une erreur due aux arrondis machine est donc à considérer. Et cette erreur augmente avec le nombre de points. En pratique, on ne peut donc pas augmenter infiniment le nombre d'intervalles des méthodes d'intégration numérique, et il existe un nombre de points particulier qui permet une précision optimale

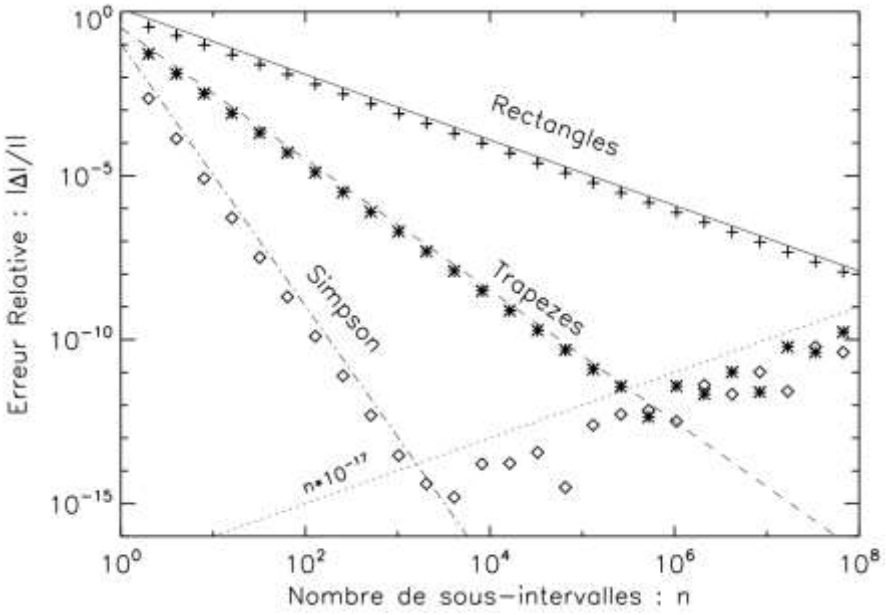


Figure 3.10 : Erreur des différentes méthodes d'intégration numériques en fonction du nombre d'intervalles utilisés pour intégrer la fonction f(x) = sinx sur l'intervalle [0, $\pi/2$]. Les points correspondent aux erreurs mesurées.

3.4 Synthèse

Différentiation numérique :

Dérivation avant : $f'(x_k) = \frac{f(x_{k+1})-f(x_k)}{h}$ (erreur proportionnelle à *h*)

Dérivation arrière : $f'(x_k) = \frac{f(x_k)-f(x_{k-1})}{h}$ (erreur proportionnelle à *h*)

Dérivation centrée : $f'(x_k) = \frac{f(x_{k+1})-f(x_{k-1})}{2h}$ (erreur proportionnelle à h^2)

Dérivée seconde : $f''(x_k) = \frac{f(x_{k+1})+f(x_{k-1})-2f(x_k)}{h^2}$ (erreur proportionnelle à h^2)

Intégration numérique :

Méthode des rectangles :

$$\int_a^b f(x)dx \approx h\,(f(a) + f(x_1) + f(x_2) + \cdots + f(x_{n-1}))$$

Erreur : $|\varepsilon_T| \le \frac{h(b-a)}{2} M_T$ avec $\forall c \in [a,b], |f'(c)| \le M_T$

Méthode des trapèzes :

$$\int_a^b f(x)dx \approx \frac{h}{2}\,(f(a) + 2f(x_1) + 2f(x_2) + \cdots + 2f(x_{n-1}) + f(b))$$

Erreur : $|\varepsilon_T| \le h^2 \frac{b-a}{12} M_T$ avec $\forall c \in [a,b], |f''(c)| \le M_T$

Méthode de Simpson :

$$\int_a^b f(x)dx \approx \frac{h}{3}\,(f(a) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(b))$$

$x_{i+1} - x_i = \frac{b-a}{n} = h$ (*n* doit être pair)

Erreur : $|\varepsilon_S| \le h^4 \frac{b-a}{180} M_S$ avec $\forall c \in [a,b], |f^{(4)}(c)| \le M_S$

Chapitre 4 : Méthodes numériques pour la résolution d'Équations Différentielles

4.1 Introduction

Les équations différentielles constituent un champ mathématique important pour l'ingénieur et le scientifique, car très souvent, les *modèles mathématiques* construits pour représenter des *phénomènes physiques* prennent la forme d'équations différentielles. Il est alors important de savoir solutionner ces équations différentielles pour obtenir la solution du problème physique modélisé, ou dans certains cas de déterminer les conditions d'existence et d'unicité des solutions.

L'étude des équations différentielles a attiré l'attention des plus grands mathématiciens des 3 siècles passés - citons Isaac Newton (1642-1727), Gottfried Wilhelm Leibniz (1646-1716), les frères Jakob (1654-1705) et Johann (1667-1748) Bernoulli, Leonhard Euler (1741-1766), Joseph-Louis Lagrange (1736-1813) et Pierre-Simon de Laplace (1749-1827). Dans le courant du 20eme siècle, les équations différentielles ont connu des développements importants au niveau des *méthodes numériques* de résolution (ces développements étant liés à l'avènement des ordinateurs) ; en parallèle, l'étude des équations différentielles *non-linéaires* a révélé des phénomènes nouveaux et fascinants qui ont permis le développement de branches importantes des mathématiques contemporaines, comme le chaos et les fractales.

Les techniques *analytiques* de résolution des équations différentielles, qui s'appliquent à divers types d'équations, offrent l'avantage de conduire à des solutions exactes (sous forme explicite ou implicite). Malheureusement, une proportion importante des équations différentielles rencontrées en pratique ne peuvent pas être résolues de façon analytique.

Pour s'en convaincre, considérons l'équation : $y'(x) = x^2 + y(x)$. Cette équation du 1er ordre n'est ni séparable, ni linéaire, ni exacte (le vérifier). En fait, malgré sa forme simple, cette équation différentielle ne possède pas de solution analytique, c'est-à-dire de solution que l'on puisse exprimer à partir de fonctions mathématiques élémentaires. Cet exemple montre que des équations différentielles de forme simple peuvent être impossibles à résoudre analytiquement. Dans ces situations, il faut recourir à des méthodes *numériques* de résolution, qui permettent d'obtenir une *approximation* de la solution. Ces méthodes numériques peuvent être très facilement programmées sur un ordinateur (ou une calculatrice), ce qui les rend très utiles de nos jours pour la résolution de problèmes complexes.

Les différentes méthodes de résolution que nous allons voir ici sont toutes basées sur le même principe, à savoir la résolution de l'équation par une récurrence partant de la condition initiale et calculant la solution de proche en proche (pas par pas) jusqu'à la valeur voulue de la variable. En général, plus ces pas sont petits (et donc plus on fait un grand nombre de pas), plus les méthodes sont précises, mais plus le temps de calcul est long.

4.2 Définitions et classification des équations différentielles

4.2.1 Définitions

À titre d'exemple simple, la vitesse *v(t)* à l'instant *t* d'un objet de masse *m* en chute libre verticale sous l'action d'une accélération gravitationnelle constante *g*, et en l'absence d'autres forces (frottements, ...) est donnée par la 2ème loi de Newton $m\frac{dv}{dt} = mg$, c'est-à-dire $\frac{dv}{dt} = g$

Dans cette équation, la fonction inconnue $v(t)$ apparaît sous la forme d'une dérivée ; il s'agit donc d'une équation différentielle. Dans ce cas, il est facile d'obtenir la solution en intégrant directement l'équation différentielle,

$dv = gdt$ donc : $\int_0^t dv = \int_0^t gdt$

Comme *g* est une constante, on obtient finalement : $v(t) - v(0) = gt$ ou encore : $v(t) = gt + v(0)$

où *v*(0) est la vitesse initiale (au temps *t*=0) de l'objet. Dans ce cas particulier simple, la résolution de l'équation différentielle se ramène à l'intégration de la fonction constante *g*. De façon générale, les techniques d'intégration auront beaucoup d'importance dans la résolution des équations différentielles. On remarque de plus sur cet exemple que la solution de l'équation différentielle implique une constante d'intégration, qui est déterminée par une condition initiale de vitesse *v*(0).

Examinons maintenant une situation moins simple : dans des cas plus complexes, la chute libre d'un objet devra faire intervenir d'autres forces (frottement de l'air, par exemple) que la seule force gravitationnelle *mg* ; si l'on note *F*(*t,v*) la force extérieure totale qui agit sur l'objet (cette force peut dépendre du temps *t* et de la vitesse *v* de l'objet), on obtient

$$m\frac{dv}{dt} = F(t,v)$$

Dans ce cas, il n'est plus possible d'intégrer simplement puisque la fonction inconnue *v*(*t*) apparaît au membre de droite de l'équation. Cet exemple d'équation différentielle est discuté plus en détail dans la section 2 de ce chapitre. Notons que l'équation différentielle précédente peut se mettre sous la forme

$G(t,v,\frac{dv}{dt}) = 0$ où $G(t,v,\frac{dv}{dt}) = F(t,v) - m\frac{dv}{dt}$.

L'écriture $G(t,v,\frac{dv}{dt}) = 0$ est la forme générale d'une équation différentielle du premier ordre. Donnons à présent les définitions générales :

Définition d'une équation différentielle

On appelle équation différentielle (ou encore équation différentielle *ordinaire*) toute expression de la forme :

$$G(x, y, y', y'', \ldots, y^{(n)}) = 0$$

où *G* est une fonction quelconque. Dans cette expression, on définit :

- *x :* variable **indépendante** (notée *t* dans les exemples précédents) ;
- *y(x) : va*riable **dépendante** ; c'est la fonction inconnue que l'on cherche à déterminer en solutionnant l'équation différentielle.

On note $y' = \frac{dy}{dx}$, $y'' = \frac{d^2y}{dx^2}$, ... $y^{(n)} = \frac{d^n y}{dx^n}$ les dérivées successives de la variable dépendante *y* par rapport à la variable indépendante *x*.

La forme $G(x, y, y', y'', \ldots, y^{(n)}) = 0$ est appelée forme implicite de l'équation différentielle.

On supposera souvent qu'il est possible d'isoler la dérivée d'ordre le plus élevé $y^{(n)}$ de façon à écrire l'équation différentielle sous la forme explicite : $y^{(n)} = F(x, y, y', \ldots y^{(n-1)})$

4.2.2 Classification des équations différentielles

Ordre d'une équation différentielle

L'**ordre** d'une équation différentielle est l'ordre de la plus haute dérivée dans l'équation différentielle. L'équation différentielle $G(x, y, y', y'', \ldots, y^{(n)}) = 0$ est d'ordre *n*, de même que l'équation différentielle $y^{(n)} = F(x, y, y', \ldots y^{(n-1)})$.

Problème aux valeurs initiales

Le **problème aux valeurs initiales** associé à l'équation différentielle d'ordre *n*

$$y^{(n)} = F(x, y, y', \ldots y^{(n-1)})$$

consiste à chercher la solution de l'équation différentielle sur un intervalle $I=[x_0, x_1]$, qui satisfait en x_0 les *n* conditions initiales :

$y(x_0)=y_0$, $y'(x_0)=y_1$, $y''(x_0)=y_2$, ... $y^{(n-1)}(x_0)=y_{n-1}$,

où $y_0, y_1, y_2, \ldots y_{n-1}$ sont *n* constantes.

Équations différentielles linéaires

Une classification très importante répartit les équations différentielles en équations **linéaires** et en équations **non-linéaires**. Une équation différentielle $G(x, y, y', y'', \ldots, y^{(n)}) = 0$ est linéaire lorsque la fonction *G* est linéaire par rapport à *y*, *y'*, ...$y^{(n)}$. La forme générale d'une équation différentielle linéaire est donc :

$$a_n(x)y^{(n)} + a_{n-1}(x)y^{(n-1)} + \ldots + a_1(x)y' + a_0(x)y = g(x)$$

Une équation différentielle linéaire se présente donc comme une combinaison linéaire des dérivées successives de la variable dépendante *y*. Les coefficients de $a_i(x)$ de cette décomposition, de même que le second membre *g(x)* ne doivent dépendre que de la variable indépendante *x*.

Équations différentielles linéaires à coefficients constants

Une équation différentielle linéaire est à **coefficients constants** lorsque les coefficients a_i sont des constantes. La forme générale d'une équation différentielle linéaire à coefficients constants est donc :

$$a_n y^{(n)} + a_{n-1} y^{(n-1)} + \ldots + a_1 y' + a_0 y = g(x)$$

Exemples : Dans les exemples suivants, indiquer s'il s'agit d'une E.D., donner l'ordre de l'équation, préciser la variable dépendante et la variable indépendante, préciser s'il l'E.D. est linéaire ou non.

$3\frac{d^2x}{dt^2} + 4\frac{dx}{dt} + 9x = 2\cos 3t$	(Vibrations mécaniques, circuits électriques)
$\frac{d^2y}{dx^2} - 2x\frac{dy}{dx} + 2y = 0$	(Équation de Hermite, mécanique quantique)
$\frac{dy}{dx} = \frac{y(2-3x)}{x(1+3y)}$	(Écologie, compétition entre 2 espèces)
$\frac{dp}{dt} = kp(P-p)$ (*k, P* constantes)	(démographie, épidémiologie, économie)
$\frac{dx}{dt} = (4-x)(1-x)$	(Vitesse d'une réaction chimique)
$8\frac{d^4y}{dx^4} = x(1-x)$	(Flexion d'une poutre)
$(y\tan\alpha)^2\frac{dy}{dt} = \sqrt{(gy)}$ (α, *g* constantes)	(Écoulement d'un liquide)

4.3 Méthode de résolution numérique d'équations différentielles

Considérons, dans toute la suite du document, le problème aux valeurs initiales suivant :

$$\frac{dy(x)}{dx} = \mathrm{F}\big(\mathrm{x}, \mathrm{y}(\mathrm{x})\big)$$

$$y(x_0) = y_0$$

Ce problème aux valeurs initiales admet une solution unique « autour » du point (x_0, y_0) Nous allons par la suite chercher à trouver une approximation de cette solution.

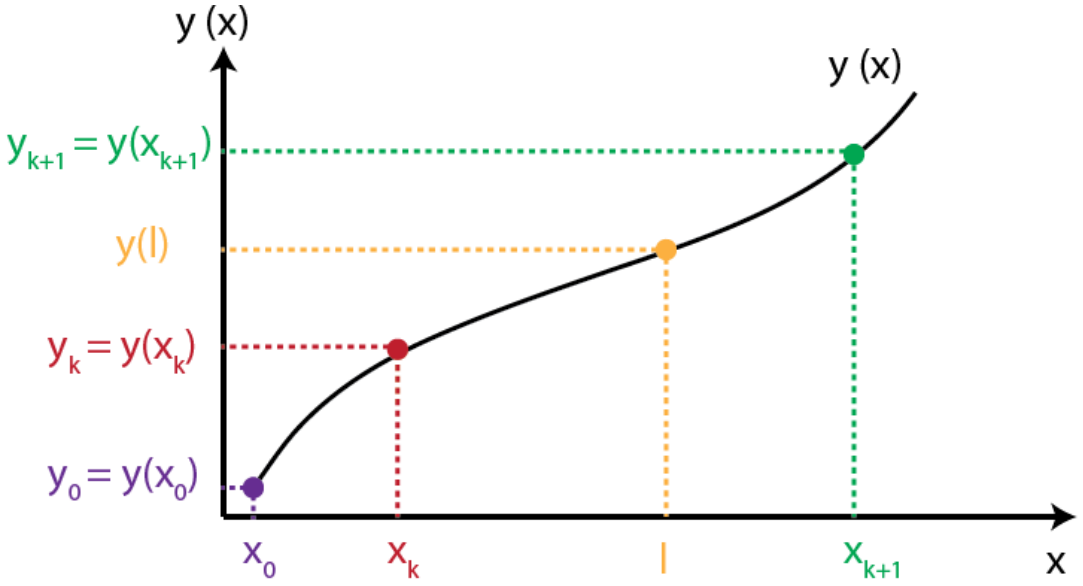


Figure 4.1 : Solution exacte $y(x)$ du problème aux valeurs initiales

Une manière de présenter les choses est de réécrire l'équation différentielle $y'(x) = F\big(x, y(x)\big)$ associée à sa condition initiale $y'(x_0) = y_0$ comme le problème intégral suivant (on utilise la définition de l'intégrale):

$$y(x) = y_0 + \int_{x_0}^{x} y'(l)dl \quad = y_0 + \int_{x_0}^{x} F\big(l, y(l)\big)\, dl$$

Le but est alors de calculer cette intégrale en appliquant des méthodes inspirées de l'intégration numériques vues précédemment. Comparé au problème d'une intégration simple $\int f(l)dl$, la difficulté essentielle provient ici du fait que l'on ne connait pas $y(l)$ dont dépend l'intégrale $\int_{x_0}^{x} F\big(l, y(l)\big)\, dl$ puisqu'il s'agit justement de la solution cherchée. On ne peut donc pas appliquer directement ces méthodes. Aussi, Pour résoudre ce problème intégral, on procède en trois étapes :

1. La discrétisation:

Comme pour les méthodes d'intégration numérique, on divise le domaine d'intégration $[x_0; x]$ en n intervalles de largeur $h = (x - x_0)/n$. h est appelé le pas d'intégration et permet ainsi de délimiter les $n + 1$ points $x_k = x_0 + k\,h$ pour $k = 0 \dots n$.

Les valeurs $y_k = y(x_k)$ de la solution en ces points sont alors exactement définies par la relation de récurrence suivante qui revient à résoudre l'équation différentielle de départ sur chacun des petits intervalles:

$$y_0 = y(x_0)$$

$$y_{k+1} = y_k + I_k \qquad \text{avec} \quad I_k = \int_{x_k}^{x_{k+1}} F(y(l), l)\,dl$$

2. La méthode d'intégration numérique :

Sur chaque intervalle $[x_k, x_{k+1}]$ on utilise une méthode d'intégration numérique, c'est à dire : on approxime la fonction $F(y(l), l)$ par un polynôme de degré p qui coïncide avec $F(y(l), l)$ en p + 1 points sur l'intervalle $[x_k, x_{k+1}]$.

Par exemple, la méthode des rectangles est une méthode d'ordre 1 utilisant un 1 point (la valeur en x_k est utilisé pour calculer l'intégrale entre $x_k \; et \; x_{k+1}$)

3. L'approximation de $y(l)$ sur chaque intervalle $[x_k, x_{k+1}]$.

Ces méthodes d'intégration reposent sur l'évaluation de $F(y(l), l)$ aux points utilisés pour l'intégration. Pour estimer ces valeurs, il faut définir des approximations pour les valeurs de $y(l)$ en ces points. En général, on construit de telles approximations à partir de la première valeur $y(x_k)$ de l'intervalle, celle-ci étant connue. Ces approximations sont d'autant plus précises que la largeur h de l'intervalle est petite.

Les erreurs numériques associées à cette résolution ont ici deux origines distinctes :

• Premièrement, la précision est limitée par le choix de la méthode d'intégration. Par exemple, nous avons vu que les méthodes des rectangles, des trapèzes et de Simpson sont des méthodes d'ordre 1, 2 et 4 respectivement. Elles participent donc ici aussi à l'erreur totale.

• D'autre part, pour $k > 0$, le calcul de I_k ne se fait pas à partir de la valeur exacte y_k, mais à partir de la valeur approchée y_k issue des calculs précédents. En comparaison d'une intégration simple, les erreurs peuvent donc s'accumuler au fil des itérations. Nous verrons que dans certains cas, la solution numérique s'éloigne progressivement de la solution exacte au fil des itérations, traduisant un problème de stabilité de la méthode.

4.4 Schémas numériques d'intégration des ODEs

4.4.1 Méthode d'Euler (Runge-Kutta d'ordre 1)

La méthode d'Euler est la plus facile à interpréter et la plus simple à mettre en œuvre. La méthode d'intégration numérique utilisée pour intégrer la fonction $F(y(l), l)$ sur chaque intervalle $[x_k, x_{k+1}]$ est celle d'ordre le plus bas, c'est à dire la méthode du **rectangle** :

$$I_k = \int_{x_k}^{x_{k+1}} F(y(l), l)dl = (x_{k+1} - x_k)\, F(y(x_k), x_k) = h\, F(y(x_k), x_k)$$

Cette méthode ne nécessite l'évaluation de $F(y(l), l)$ qu'en x_k où la fonction $y_k = y(x_k)$ est déjà connue (calculée au pas précédent). Il n'y a donc pas besoin d'approximer la fonction $y(l)$ en ce point. Au final, l'intégration par la méthode d'Euler consiste à réaliser le calcul de la récurrence suivante :

si $k = 0$ $\qquad y_0 = y(x_0)$

pour tout $k > 0$ $\qquad \begin{cases} k_1 = h\, F(y(x_k), x_k) \\ y_{k+1} = y_k + k_1 \end{cases}$

Chaque pas d'intégration nécessite une unique évaluation de la fonction $F(y(x_k), x_k)$. La méthode d'Euler est donc la méthode la plus rapide pour effectuer un nombre n de pas d'intégration donné.

La méthode d'Euler peut aussi être interprétée selon un autre point de vue. En effet, si on utilise l'approximation suivante pour l'expression de la dérivée en x_k : $y'_k = (y_{k+1} - y_k)/h$, alors l'EDO à résoudre s'écrit : $\frac{(y_{k+1} - y_k)}{h} = F(y(x_k), x_k)$, ce qui correspond exactement à la relation de récurrence obtenue précédemment. La méthode de Euler consiste donc à estimer la pente en x_k avec cette relation, et à calculer une valeur de x_{k+1} en utilisant cette pente.

Exemple :

Soit le problème aux valeurs initiales suivant : $\frac{dy(x)}{dx} = \mathrm{y(x)} \qquad y(0) = 1$

On établit facilement que la solution exacte est $y(x) = e^x$. Comparons cette solution avec la solution approchée fournie par la méthode d'Euler. Ici, $F(x, y) = y(x)$, et $y_0 = 1$ donc la méthode d'Euler génère la séquence suivante de valeurs approchées:

$y_0 = 1$
$y_1 = y_0 + hy_0 = (1+h)y_0 = 1+h$
$y_2 = y_1 + hy_1 = (1+h)y_1 = (1+h)^2$
...
$y_n = y_{n-1} + hy_{n-1} = (1+h)y_{n-1} = (1+h)^n$.

Dans cet exemple, on obtient donc une formule explicite pour l'approximation y_n en fonction du pas *h* (en général, ce n'est pas le cas et il faut plutôt calculer y_n par un calcul de récurrence). Le graphique ci-dessous représente le tracé de la solution exacte $y = e^x$ et des solutions approchées pour différentes valeurs du pas *h* entre $x = 0$ et $x = 1$.

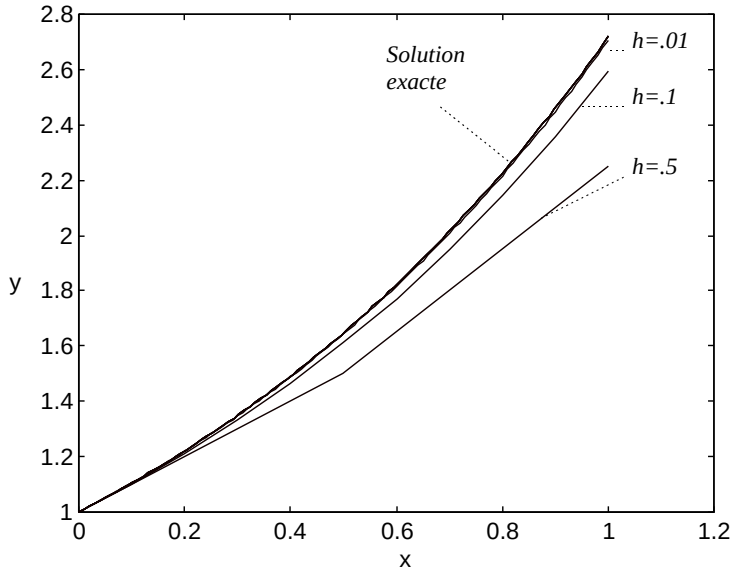


Figure 4.2 : Solution de $y'(x) = y(x)$, $y(0) = 1$ par la méthode d'Euler

Pour apprécier la précision de la méthode, analysons-en la **convergence** : examinons par exemple au point la valeur de la solution approchée pour différentes valeurs du pas *h*, et comparons avec la solution exacte $y(1) = e = 2.71828$ (à 6 chiffres significatifs).

Pas *h*	0.5	0.1	0.01	0.001
$x = 1$ correspond à	*n*=2	*n*=10	*n*=100	*n*=1000
Approximation de *y*(1)	2.250000	2.59374	2.70481	2.71692
Erreur	0.46828	0.12454	0.01347	0.00136

On peut remarquer dans cet exemple que la diminution du pas conduit à une meilleure approximation de la solution exacte. De plus, chose très importante, l'erreur est approximativement divisée par 10 lorsque le pas est divisé par 10 (comparer les erreurs lorsque $h = 0.1$, $h = 0.01$, $h = 0.001$). En fait, pour des pas très petits, l'erreur est exactement divisée par 10 lorsque le pas est divisé par 10. Ceci montre que pour garantir par exemple une erreur de l'ordre de 10^{-5}, il faut utiliser un pas de l'ordre de 10^{-5} (c'est-à-dire *n*=100 000 points entre $x = 0$ et $x = 1$).

Ce résultat important se généralise à tout problème aux valeurs initiales solutionné par la méthode d'Euler, et peut se résumer de la façon suivante (la démonstration n'est pas reproduite ici) :

L'erreur d'approximation de la méthode d'Euler est proportionnelle au pas *h*. On dit que **la méthode d'Euler est d'ordre 1**.

Notons qu'en général, l'erreur d'approximation n'est pas accessible directement car la solution exacte est inconnue. Cependant, ce résultat permet de garantir que l'erreur d'approximation « varie comme *h* ».

4.3.2 Méthode de Heun (Runge-Kutta d'ordre 2)

La convergence de la méthode d'Euler est généralement trop lente en pratique : il est souhaitable d'améliorer la méthode de façon à obtenir une faible erreur d'approximation sans nécessairement réduire excessivement le pas.

La méthode de Heun (ou méthode de Runge Kutta d'ordre 2) est un peu plus complexe à mettre en oeuvre. La méthode d'intégration numérique utilisée pour intégrer la fonction $F(y(l), l)$ sur chaque intervalle $[x_k, x_{k+1}]$ est celle d'ordre 2, c'est à dire la méthode du **trapèze** :

$$I_k = \int_{x_k}^{x_{k+1}} F(y(l), l)\, dl$$

$$I_k = (x_{k+1} - x_k)\left(\frac{F(y(x_k), x_k) + F(y(x_{k+1}), x_{k+1})}{2}\right)$$

$$I_k = h\left(\frac{F(y(x_k), x_k) + F(y(x_{k+1}), x_{k+1})}{2}\right)$$

Cette méthode nécessite l'évaluation de $F(y(l), l)$ en x_k où la fonction $y_k = y(x_k)$ est déjà connue (calculée au pas précédent), mais également au point x_{k+1} où la fonction $y_{k+1} = y(x_{k+1})$ n'est pas encore connue !

Il est donc besoin de trouver une approximation de la fonction $y(l)$ en ce point. Dans la méthode de Heun, une première estimation de $y_{k+1} = y(x_{k+1})$ est obtenue en utilisant la méthode d'Euler :

$$y_{k+1} = y_k + h\,\frac{dy(x)}{dx}\,(x_k)$$

$$y_{k+1} = y_k + h\,F(x_k, y_k)$$

Au final, l'intégration par la méthode de Heun consiste à réaliser le calcul de la récurrence suivante :

$$\text{si } k = 0 \qquad y_0 = y(x_0)$$

$$\text{pour tout } k > 0 \qquad \begin{cases} k_1 = h\, F(y(x_k), x_k) \\ k_2 = h\, F(y_k + k_1, x_{k+1}) \\ y_{k+1} = y_k + \frac{k_1 + k_2}{2} \end{cases}$$

Chaque pas d'intégration nécessite deux évaluations de la fonction $F(y(x_k), x_k)$. La méthode d'Euler est donc deux fois plus lente que la méthode de Euler pour effectuer n pas d'intégration.

L'interprétation de la méthode de Heun en termes des dérivées est plus complexe. On peut cependant le voir de la manière suivante. La valeur de y_{k+1} est obtenue en utilisant la pente moyenne entre deux pentes :

- La pente k_1 au début de l'intervalle (en x_k) qui est est connue
- La pente k_2 à la fin de l'intervalle (en x_{k+1}) que l'on approxime en utilisant une première estimation

Exemple :

Soit encore une fois le problème aux valeurs initiales

$$\frac{dy(x)}{dx} = \mathrm{y(x)} \qquad y(0) = 1$$

Ici, $F(x, y) = y$, et $y_0 = 1$ donc la méthode de Heun (ou méthode d'Euler améliorée) génère la séquence de valeurs approchées

$$y_0 = 1$$

$$y_1 = y_0 + \frac{h}{2}\left[y_0 + y_0 + hy_0\right] = (1 + h + \frac{h^2}{2})y_0 = (1 + h + \frac{h^2}{2})$$

$$y_2 = y_1 + \frac{h}{2}\left[y_1 + y_1 + hy_1\right] = (1 + h + \frac{h^2}{2})y_1 = (1 + h + \frac{h^2}{2})^2$$

...

$$y_n = y_{n-1} + \frac{h}{2}\left[y_{n-1} + y_{n-1} + hy_{n-1}\right] = (1 + h + \frac{h^2}{2})y_{n-1} = (1 + h + \frac{h^2}{2})^n .$$

Le tableau ci-dessous compare les approximations obtenues par la méthode d'Euler et par la méthode d'Euler améliorée au point *x*=1 (la solution exacte est $y(1) = e = 2.71828$).

Pas *h*	0.5	0.1	0.01	0.001
$x = 1$ correspond à	*n*=2	*n*=10	*n*=100	*n*=1000
Approximation de *y*(1) (méthode. d'Euler)	2.250000	2.59374	2.70481	2.71692
Erreur	0.46828	0.12454	0.01347	0.00136
Approximation de *y*(1) (méthode. De Heun)	2.64063	2.71408	2.71824	2.71828
Erreur	0.07766	0.00420	0.00004	0.00000

Le graphique ci-dessous compare la solution exacte $y = e^x$ avec l'approximation par la méthode d'Euler et par la méthode de Heun pour *h*=0.5. Ce graphique et le tableau précédent montrent que la méthode de Heun est beaucoup plus précise que la méthode d'Euler. Précisément, on remarque qu'entre *h*=0.1 et *h*=0.01, l'erreur de la méthode de Heun est divisée environ par 100 (par 10 pour la méthode d'Euler).

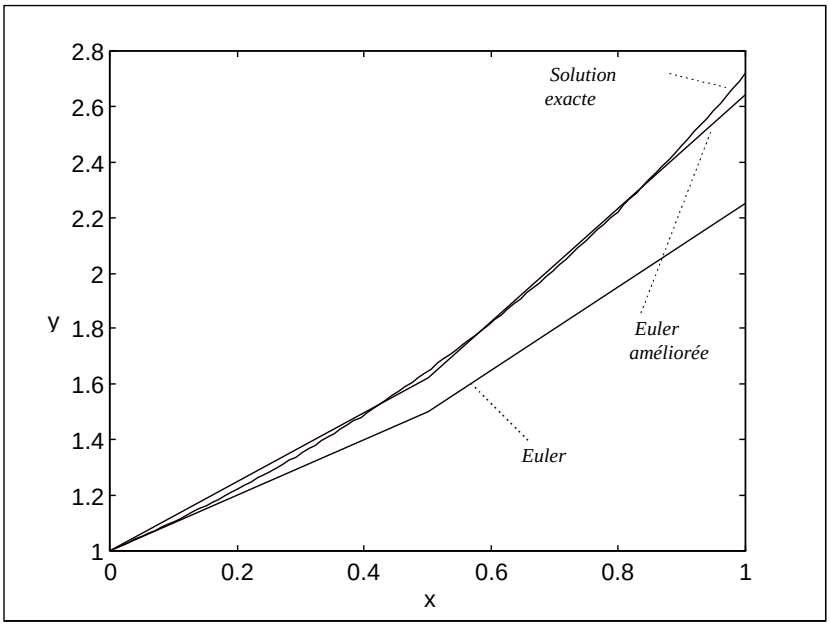


Figure 4.3 : Solution de $\frac{dy}{dx} = y$, $y(0) = 1$ par la méthode d'Euler améliorée

De façon générale, on peut montrer que :

L'erreur d'approximation de la méthode de Heun est proportionnelle à h^2. On dit que **la méthode d'Euler améliorée est d'ordre 2**.

4.3.3 Méthode de Runge-Kutta d'ordre 4

Comme nous le verrons, la méthode de Runge-Kutta est une méthode dont la précision est encore supérieure à la méthode de Heun, tout en étant relativement simple à programmer.

La méthode de Runge-Kutta 4 est encore plus complexe à mettre en œuvre, mais c'est la méthode de choix utilisée pour l'intégration numérique de problèmes courants. La méthode d'intégration numérique utilisée pour intégrer la fonction $F(y(l), l)$ sur chaque intervalle $[x_k, x_{k+1}]$ est celle d'ordre 4, c'est à dire la méthode de **Simpson** :

$$I_k = \int_{x_k}^{x_{k+1}} F(y(l), l)dl$$

$$I_k = (x_{k+1} - x_k)\left(\frac{F(y(x_k), x_k) + 4F\left(y\left(x_{k+1/2}\right), x_{k+1/2}\right) + F(y(x_{k+1}), x_{k+1})}{6}\right)$$

$$I_k = h\left(\frac{F(y(x_k), x_k) + 4F\left(y\left(x_{k+1/2}\right), x_{k+1/2}\right) + F(y(x_{k+1}), x_{k+1})}{6}\right)$$

Cette méthode nécessite l'évaluation de $F(y(l), l)$ en x_k où la fonction $y_k = y(x_k)$ est déjà connue (calculée au pas précédent), mais également au point $x_{k+1/2}$ x_{k+1} où les valeurs $y_{k+1/2} = y(x_{k+1/2})$ et $y_{k+1} = y(x_{k+1})$ ne sont pas encore connues !

Il est donc besoin de trouver une approximation de la fonction $y(l)$ en ces points. s. Dans la méthode Runge-Kutta d'ordre 4 (RK4), des premières estimations de $y_{k+1/2}$ et y_{k+1} sont obtenues en utilisant les méthodes de Euler et Heun (non détaillées ici par soucis de simplicité).

Au final, l'intégration par la méthode de Runge Kutta d'ordre 4 consiste à réaliser le calcul de la récurrence suivante :

$$\text{si } k = 0 \qquad y_0 = y(x_0)$$

$$\text{pour tout } k > 0 \quad \begin{cases} k_1 = h\, F(y(x_k), x_k) \\ k_2 = h\, F\left(y_k + \frac{k_1}{2}, x_{k+1/2}\right) \\ k_3 = h\, F\left(y_k + \frac{k_2}{2}, x_{k+1/2}\right) \\ k_4 = h\, F(y_k + k_3, x_{k+1}) \\ y_{k+1} = y_k + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6} \end{cases}$$

Chaque pas d'intégration nécessite 4 évaluations de la fonction $F(y(l), l)$. La méthode de Runge Kutta d'ordre 4 est donc 4 fois plus lente que la méthode de Euler pour effectuer n pas d'intégration.

L'interprétation de la méthode de RK4 en termes des dérivées est plus complexe mais toujours faisable ! On peut ainsi si interpréter la méthode RK4 comme l'extrapolation de y_k à y_{k+1}en utilisant une pente calculée comme la moyenne pondérée de 4 pentes différentes :

- La pente k_1 au début de l'intervalle (en x_k) qui est est connue
- La pente $k_2$au milieu de l'intervalle (en $x_{k+1/2}$) que l'on approxime en utilisant une première estimation de $y_{k+1/2}$ à l'aide de k_1
- La pente $k_3$au milieu de l'intervalle (en $x_{k+1/2}$) que l'on approxime en utilisant une seconde estimation de $y_{k+1/2}$ à l'aide de k_2
- La pente k_4 à la fin de l'intervalle (en x_{k+1}) que l'on approxime en utilisant une estimation de y_{k+1} à l'aide de k_3

Exemple :

Soit encore une fois (c'est la dernière promis !) le problème aux valeurs initiales

$$\frac{dy(x)}{dx} = \mathrm{y(x)} \qquad\qquad y(0) = 1$$

Ici, $F(x,y) = y$, et $y_0 = 1$. Nous avons, à l'itération *n*,

$k_1 = F(x_n, y_n) = y_n$,

$k_2 = F(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) = y_n + \frac{h}{2}k_1 = (1+\frac{h}{2})y_n$,

$k_3 = F(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) = y_n + \frac{h}{2}k_2 = y_n + \frac{h}{2}(1+\frac{h}{2})y_n = (1+\frac{h}{2}+\frac{h^2}{4})y_n$

$k_4 = F(x_n + h, y_n + hk_3) = y_n + hk_3 = y_n + h(1+\frac{h}{2}+\frac{h^2}{4})y_n = (1+h+\frac{h^2}{2}+\frac{h^3}{4})y_n$

On en déduit

$$y_{n+1} = y_n + \frac{h}{6}\left[k_1 + 2k_2 + 2k_3 + k_4\right] = (1+h+\frac{h^2}{2}+\frac{h^3}{6}+\frac{h^4}{24})y_n$$

La séquence des y_n est donc une suite géométrique de raison $(1+h+\frac{h^2}{2}+\frac{h^3}{6}+\frac{h^4}{24})$. Comme $y_0 = 1$, on en déduit que

$$y_n = (1+h+\frac{h^2}{2}+\frac{h^3}{6}+\frac{h^4}{24})^n$$

Le tableau ci-dessous compare les approximations obtenues par la méthode d'Euler améliorée et par la méthode de Runge-Kutta au point *x*=5 (la solution exacte est $y(5) = e^5 = 148.413$).

Pas *h*	1	0.5	0.1	0.01
$x = 5$ correspond à	*n*=5	*n*=10	*n*=50	*n*=500
Approximation de *y*(5) (méthode de Heun)	68.281	107.622	145.586	148.382
Erreur	80.132	40.791	2.827	0.031
Approximation de *y*(5) (méthode de Runge-Kutta)	145.717	148.158	148.413	148.413
Erreur	2.696	0.255	0.000	0.000

Le graphique ci-dessous compare la solution exacte $y = e^x$ avec l'approximation par la méthode de Runge-Kutta pour deux valeurs du pas *h*. Ce graphique et le tableau précédent montrent que la méthode de Runge-Kutta est très précise.

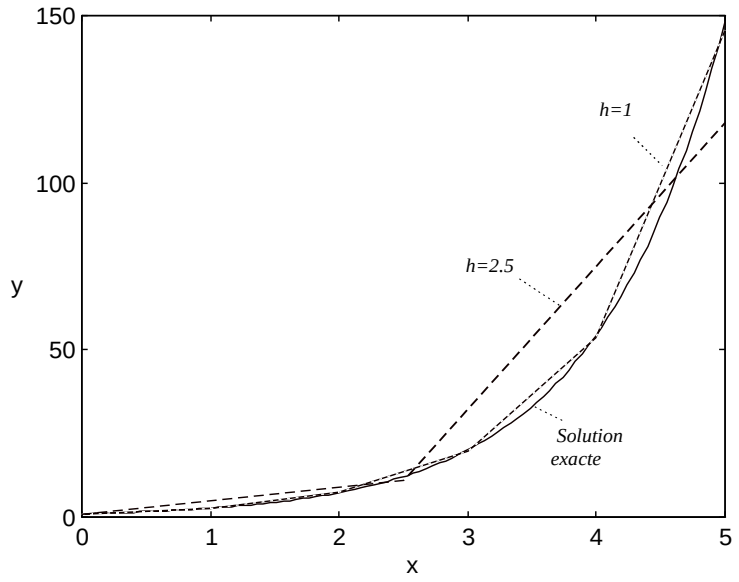


Figure 4.4 : Solution de $\frac{dy}{dx} = y$, $y(0) = 1$ par la méthode de Runge-Kutta

De façon générale, on peut montrer que :

L'erreur d'approximation de la méthode de Runge-Kutta est proportionnelle à h^4. **La méthode de Runge-Kutta est d'ordre 4**.

Remarque :

Rappelons, dans l'exemple du problème aux valeurs initiales $\frac{dy}{dx} = y$, $y(0) = 1$, les formules de récurrence obtenues par chacune des trois méthodes :

- Pour la méthode d'Euler, $y_{n+1} = (1+h)y_n$;
- Pour la méthode d'Euler améliorée, $y_{n+1} = (1+h+\frac{h^2}{2})y_n$;
- Pour la méthode de Runge-Kutta, $y_{n+1} = (1+h+\frac{h^2}{2}+\frac{h^3}{6}+\frac{h^4}{24})y_n$.

Comparons ces résultats avec la solution exacte, $y = e^x$; nous avons $y_n = e^{x_n}$ et $y_{n+1} = e^{x_{n+1}} = e^{x_n+h} = y_n e^h$. En effectuant un développement en série de Taylor de e^h,

$$e^h = 1 + h + \frac{h^2}{2} + \frac{h^3}{6} + \frac{h^4}{24} + ... + \frac{h^n}{n!} + o(h^{n+1}),$$

on a donc

$$y_{n+1} = \left[1 + h + \frac{h^2}{2} + \frac{h^3}{6} + \frac{h^4}{24} + ... + \frac{h^n}{n!} + o(h^{n+1})\right] y_n$$

On remarque que donc que chacune des 3 méthodes fournit une approximation de la relation de récurrence exacte entre y_n et y_{n+1}, où la série infinie est tronquée à un ordre plus ou moins grand suivant la méthode. L'erreur d'approximation dans l'estimation de y_{n+1} à partir de y_n est donc :

- de l'ordre de h^2 pour la méthode d'Euler,
- de l'ordre de h^3 pour la méthode d'Euler améliorée,
- de l'ordre de h^5 pour la méthode de Runge-Kutta.

Cette erreur d'approximation est appelée *erreur de troncature locale* et ne doit pas être confondue avec l'erreur d'approximation dans l'estimation absolue de y_n (qui est de l'ordre de h pour la méthode d'Euler, h^2 pour la méthode d'Euler améliorée, h^4 pour la méthode de Runge-Kutta). L'erreur d'approximation dans l'estimation absolue de y_n est aussi appelée *erreur de troncature globale*.

4.3.4 Remarques sur la stabilité des méthodes numériques

Nous avons vu que l'*ordre* des différentes méthodes numériques conditionne la précision des approximations qu'elles fournissent, et donc la valeur du pas *h* nécessaire pour obtenir un résultat avec une précision donnée (à cet égard, la méthode de Runge-Kutta est la plus précise des 3 méthodes étudiées). Il existe cependant un autre critère qui peut imposer des limitations sur la valeur du pas *h* pour certains types de problèmes ; ce second critère est la *stabilité* du schéma numérique, qui est examinée ici par le biais d'un exemple simple.

Soit le problème aux valeurs initiales $\frac{dy}{dx} = -y$, $y(0) = 1$. On établit facilement que la solution de ce problème est $y(x) = e^{-x}$. Cette solution tend asymptotiquement vers 0 lorsque *x* devient très grand,

$$\lim_{x \to +\infty} y = 0$$

Il est alors souhaitable que tout schéma numérique qui génère des approximations y_n de $y(x)$ ait la propriété suivante,

$$\lim_{n \to +\infty} y_n = 0$$

Si c'est le cas, on dit que le schéma numérique possède la propriété de **stabilité absolue**. Dans le cas contraire, le schéma numérique est dit instable. L'instabilité numérique résulte de l'erreur d'approximation à chaque pas x_n ; cette erreur peut décroître d'un pas x_n au suivant x_{n+1}, créant ainsi un schéma stable, ou au contraire croître d'un pas x_n au suivant x_{n+1}, créant ainsi un schéma instable.

Examinons la stabilité de la méthode d'Euler et de la méthode de Runge-Kutta sur cet exemple :

Ici, $F(x, y) = -y$ donc la méthode d'Euler conduit à l'approximation

$$y_{n+1} = y_n - h y_n = (1 - h) y_n$$

avec $y_0 = 1$. On obtient donc

$$y_n = (1 - h)^n$$

Pour garantir la stabilité de la méthode d'Euler, il faut que $\lim_{n \to +\infty} (1 - h)^n = 0$. On sait que $\lim_{n \to +\infty} a^n = 0$ lorsque $|a| < 1$. La méthode d'Euler est donc stable lorsque $|1 - h| < 1$, c'est-à-dire

$$h < 2$$

Le graphique ci-dessous montre la solution exacte $y(x) = e^{-x}$ et les approximations par la méthode d'Euler pour des pas $h = 0.5$ (schéma stable), $h = 2$ (limite de stabilité) et $h = 2.5$ (schéma instable).

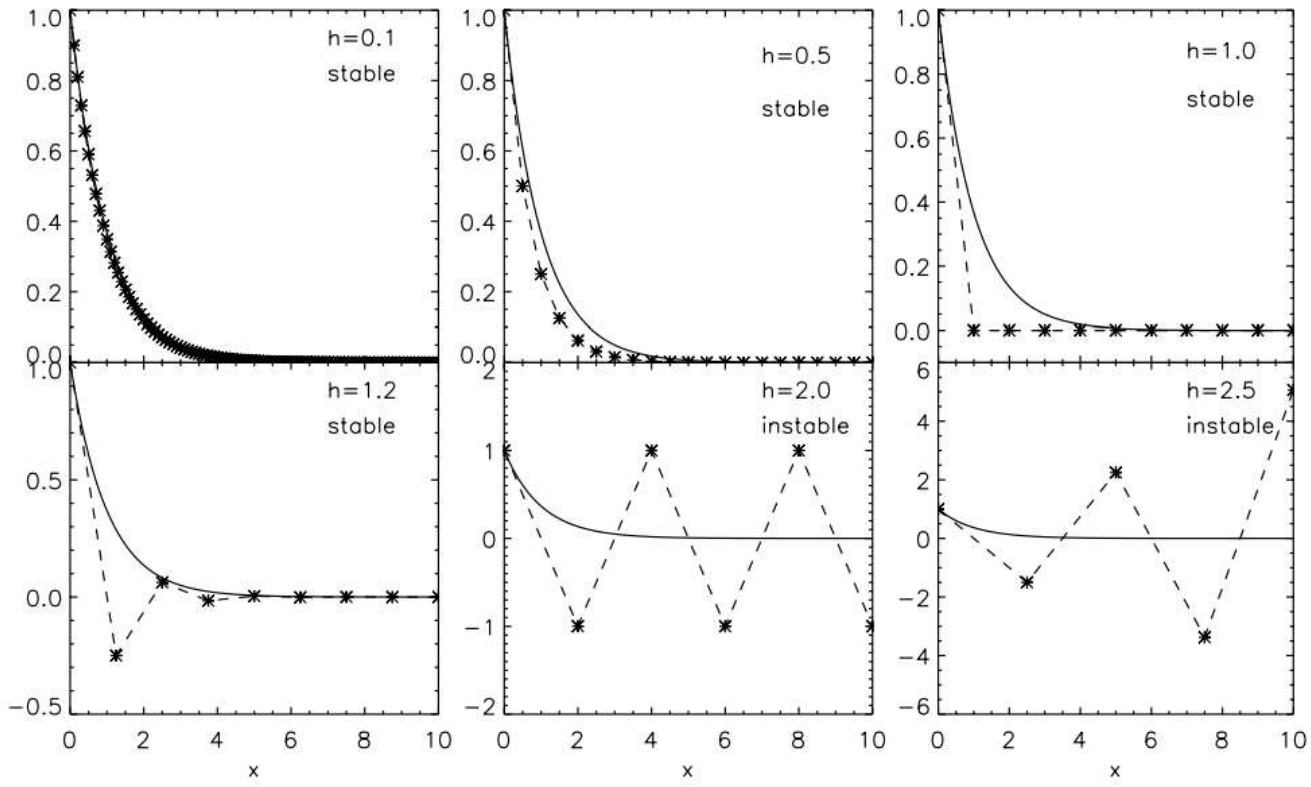


Figure 4.5 : Stabilité de la méthode d'Euler dans le cas de l'équation $\frac{dy}{dx} = -y,\ y(0) = 1.$

Dans le cas de la méthode de Runge-Kutta, le schéma d'approximation est

$$y_{n+1} = y_n + \frac{h}{6}\left[k_1 + 2k_2 + 2k_3 + k_4\right]$$

avec :

$$k_1 = F(x_n, y_n) = -y_n,$$

$$k_2 = F(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) = -(y_n + \frac{h}{2}k_1) = (-1 + \frac{h}{2})y_n,$$

$$k_3 = F(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) = -(y_n + \frac{h}{2}k_2) = (-1 + \frac{h}{2} - \frac{h^2}{4})y_n$$

$$k_4 = F(x_n + h, y_n + hk_3) = -(y_n + hk_3) = (-1 + h - \frac{h^2}{2} + \frac{h^3}{4})y_n$$

On en déduit

$$y_{n+1} = y_n(1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{24})$$

Comme $y_0 = 1$, on obtient

$$y_n = (1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{24})^n$$

Pour garantir la stabilité de la méthode de Runge-Kutta, il faut que $\lim_{n\to+\infty}(1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{24})^n = 0$. La méthode de Runge-Kutta est donc stable lorsque $\left|1 - h + \frac{h^2}{2} - \frac{h^3}{6} + \frac{h^4}{24}\right| < 1$, c'est-à-dire (on peut le montrer numériquement)

$$h < 2.78$$

Le graphique ci-dessous montre la solution exacte $y(x) = e^{-x}$ et les approximations par la méthode de Runge-Kutta pour des pas $h = 2$ (schéma stable), et $h = 3.33$ (schéma instable). Il faut bien garder à l'esprit que les limites de stabilité trouvées ici sont relatives à cet exemple seulement. Dans certains problèmes, la contrainte de stabilité du schéma numérique peut être très importante.

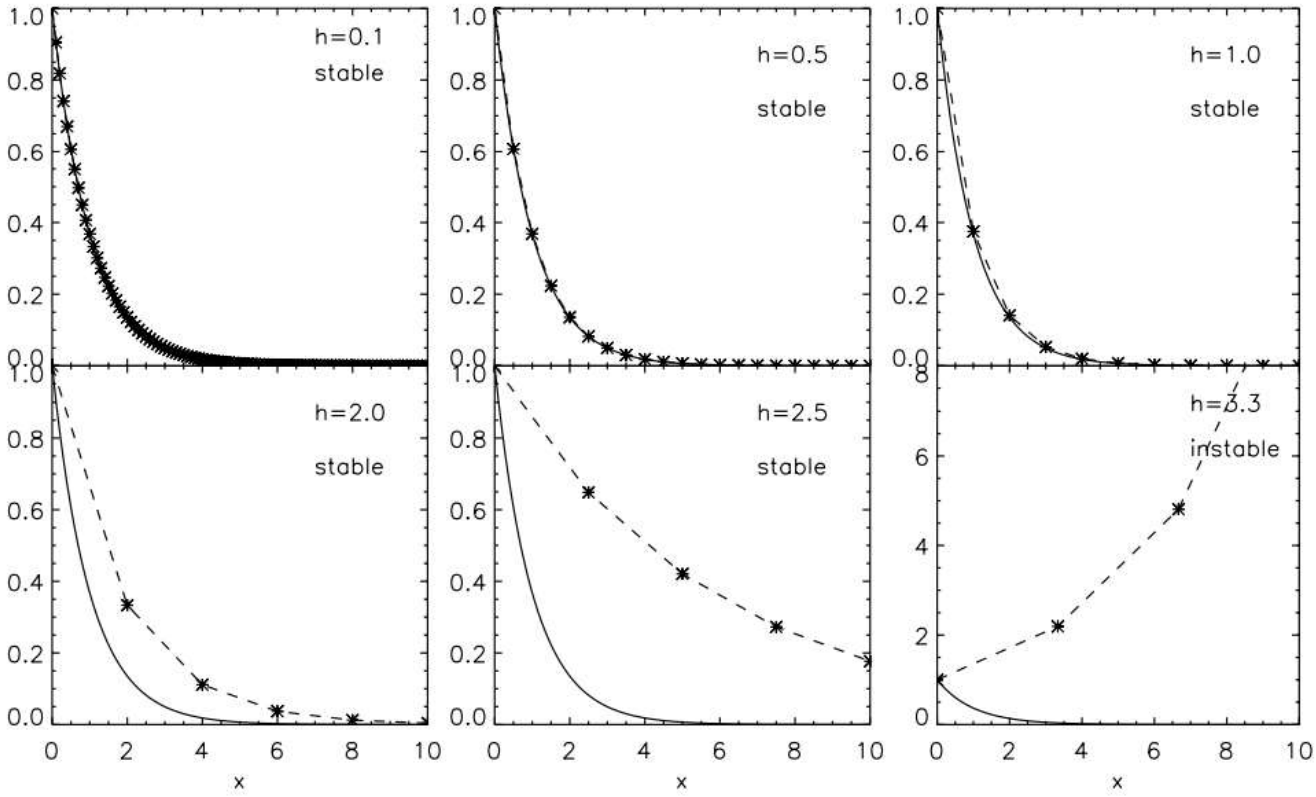


Figure 4.6 : Stabilité de la méthode de Runge-Kutta dans le cas de l'équation $\frac{dy}{dx} = -y,\ y(0) = 1.$

4.4 Application aux systèmes d'équations différentielles d'ordre supérieur

4.4.1 Système de *n* équations différentielles d'ordre 1

Soit le problème aux valeurs initiales

$$\frac{dy_1}{dx} = F_1(x, y_1, y_2, \ldots y_n) \quad y_1(x_0) = y_0^{(1)}$$
$$\frac{dy_2}{dx} = F_2(x, y_1, y_2, \ldots y_n) \quad y_2(x_0) = y_0^{(2)}$$
$$\ldots$$
$$\frac{dy_n}{dx} = F_n(x, y_1, y_2, \ldots y_n) \quad y_n(x_0) = y_0^{(n)}$$

où les fonctions $F_1, F_2, \ldots F_n$ sont continués dans une région de l'espace $(x, y_1, y_2, \ldots y_n)$ contenant le point $(x_0, y_0^{(1)}, y_0^{(2)}, \ldots y_0^{(n)})$, de même que les dérivées $\frac{\partial F_1}{\partial y_1}, \ldots \frac{\partial F_1}{\partial y_n}, \ldots \frac{\partial F_n}{\partial y_1}, \ldots \frac{\partial F_n}{\partial y_n}$. On peut montrer qu'alors, le problème aux valeurs initiales ci-dessus admet une solution unique « autour » du point $(x_0, y_0^{(1)}, y_0^{(2)}, \ldots y_0^{(n)})$, c'est-à-dire qu'il existe un *n*-uplet unique de fonctions $y_1, y_2, \ldots y_n$ qui satisfont le problème aux valeurs initiales.

On pose $x_{m+1} = x_m + h, m = 0,1,2,\ldots$ et l'on calcule des approximations $y_{1,m}, y_{2,m}, \ldots y_{n,m}$ de $y_1, y_2, \ldots y_n$ aux points x_m. Il est commode ici d'utiliser une notation vectorielle. Posons

$$\mathbf{y}_m = \begin{Bmatrix} y_{1,m} \\ y_{2,m} \\ \ldots \\ y_{n,m} \end{Bmatrix}, \mathbf{y}_0 = \begin{Bmatrix} y_0^{(1)} \\ y_0^{(2)} \\ \ldots \\ y_0^{(n)} \end{Bmatrix}$$

$$\mathbf{F}(x_m, \mathbf{y}_m) = \begin{Bmatrix} F_1(x_m, y_{1,m}, y_{2,m}, \ldots y_{n,m}) \\ F_2(x_m, y_{1,m}, y_{2,m}, \ldots y_{n,m}) \\ \ldots \\ F_n(x_m, y_{1,m}, y_{2,m}, \ldots y_{n,m}) \end{Bmatrix}$$

Les schémas d'intégration (Euler / Heune/ Runge-Kutta) sont alors formellement identique au cas d'une équation différentielle du premier ordre, sauf que dans ce cas, toutes les variables sont **vectorielles.**

4.4.2 Équations différentielles d'ordre *n*

Exemple : équation d'ordre 2

Comme nous l'avons déjà vu, les équations différentielles d'ordre 2 sont très importantes dans le cadre de la dynamique des particules ou des corps rigides, par exemple. Nous allons donc prendre un exemple de problème aux valeurs initiales sous la forme générale :

$$\frac{d^2y}{dx^2} = F(x, y, \frac{dy}{dx})$$
$$y(x_0) = y_0$$
$$\frac{dy}{dx}(x_0) = y_0'$$

Pour appliquer ces méthodes à la résolution d'un problème aux valeurs initiales d'ordre 2, nous devons tout d'abord **transformer ce problème en un système de 2 problèmes aux valeurs initiales d'ordre *1*.** Ceci est facilement possible en posant :

$$y_1(x) = y(x)$$
$$y_2(x) = \frac{dy}{dx}(x)$$

Le problème aux valeurs initiales d'ordre 2 peut donc se mettre sous la forme

$$\begin{cases} \frac{dy_1}{dx} = y_2 \\ \frac{dy_2}{dx} = F(x, y_1, y_2) \end{cases} \text{ avec } \begin{matrix} y_1(x_0) = y_0 \\ y_2(x_0) = y_0' \end{matrix},$$

ce qui constitue bien un système de 2 problèmes aux valeurs initiales d'ordre 1 simultanés qui peut être simplement résolu comme décrit à la section précédente.

Cas général :

La résolution numérique d'un problème aux valeurs initiales d'ordre *n* se ramène à la résolution d'un système de *n* équations différentielles d'ordre 1. Considérons le problème aux valeurs initiales d'ordre *n*,

$$\frac{d^n y}{dx^n} = F(x, y, \frac{dy}{dx}, \ldots \frac{d^{n-1} y}{dx^{n-1}})$$

$$y(x_0) = y_0^{(1)}$$
$$\frac{dy}{dx}(x_0) = y_0^{(2)}$$
$$\ldots$$
$$\frac{d^{n-1} y}{dx^{n-1}}(x_0) = y_0^{(n)}$$

Posons alors n variables distincites:

$$y_1(x) = y(x)$$
$$y_2(x) = \frac{dy}{dx}(x)$$
$$\ldots$$
$$y_n(x) = \frac{d^{n-1} y}{dx^{n-1}}(x)$$

Le problème aux valeurs initiales d'ordre *n* peut alors s'écrire sous la forme du système de *n* équations différentielles d'ordre 1 :

$$\frac{dy_1}{dx} = y_2(x) y_1(x_0) = y_0^{(1)}$$
$$\frac{dy_2}{dx} = y_3(x) y_2(x_0) = y_0^{(2)}$$
$$\ldots$$
$$\frac{dy_{n-1}}{dx} = y_n(x) y_{n-1}(x_0) = y_0^{(n-1)}$$

$$\frac{dy_n}{dx} = F(x, y_1, \ldots y_{n-1}) y_n(x_0) = y_0^{(n)}$$

On peut alors directement appliquer le schéma de Runge-Kutta pour un système. On pose $x_{m+1} = x_m + h, m = 0,1,2,\ldots$ et l'on calcule des approximations $y_{1,m}$, $y_{2,m}, \ldots y_{n,m}$ de y_1, $y_2, \ldots y_n$ aux points x_m. Définissons les vecteurs

$$\mathbf{y}_m = \begin{Bmatrix} y_{1,m} \\ y_{2,m} \\ \cdots \\ y_{n,m} \end{Bmatrix} \quad ,\mathbf{y}_0 = \begin{Bmatrix} y_0^{(1)} \\ y_0^{(2)} \\ \cdots \\ y_0^{(n)} \end{Bmatrix} \quad \mathbf{F}(x_m, \mathbf{y}_m) = \begin{Bmatrix} y_{2,m} \\ y_{3,m} \\ \cdots \\ y_{n,m} \\ F\ (x_m, y_{1,m}, y_{2,m}, \ldots y_{n,m}) \end{Bmatrix}$$

Le schéma de Runge-Kutta est alors formellement identique au cas d'un système de *n* équations différentielles d'ordre 1,

4.5 Implantation de la méthode de Runge-Kutta sous MATLAB

4.5.1 Mise en équation vectorielle

Afin de se familiariser avec l'implantation des équations différentielles sous Matlab, considérons l'équation différentielle ordinaire du second ordre :

$$\frac{\mathrm{d}^2 \mathrm{y}}{\mathrm{dx}^2} + a\frac{dy}{dx} + b\,y = \mathrm{e}^{-\mathrm{x}}$$

Avec pour conditions initiales : $y(0) = 0$ et $\frac{dy}{dx}(0) = 1$.

MATLAB ne sait résoudre que des systèmes différentiels du premier ordre. Une étape préliminaire à la résolution numérique est donc de mettre celle-ci sous la forme d'un système différentiel du 1er ordre. Pour ce faire, désignons par $y_1(x)$ et $y_2(x)$ les fonctions suivantes :

$$\begin{cases} y_1(x) = y(x) \\ y_2(x) = \dfrac{dy}{dx} = \dfrac{dy_1}{dx} \end{cases}$$

L'équation différentielle peut ainsi se réduire à :

$$\frac{dy_2}{dx} = \frac{\mathrm{d}^2 \mathrm{y}}{\mathrm{dx}^2} = \mathrm{e}^{-\mathrm{x}} - a\frac{dy}{dx} - b\,y$$

On peut ainsi introduire le vecteur de variables :

$$Y = \begin{pmatrix} y_1(x) \\ y_2(x) \end{pmatrix}$$

Et on peut ainsi écrire que l'équation de départ peut s'exprimer ainsi :

$$\frac{dY}{dx} = \begin{pmatrix} \frac{dy_1}{dx} \\ \frac{dy_2}{dx} \end{pmatrix} = \begin{pmatrix} \frac{dy_1}{dx} \\ \frac{dy_2}{dx} \end{pmatrix} = \begin{pmatrix} \frac{dy_1}{dx} \\ \mathrm{e}^{-\mathrm{x}} - a\frac{dy}{dx} - b\,y \end{pmatrix}$$

L'idée est ensuite d'exprimer $\frac{dY}{dx}$ à l'aide des variables $y_1(x) = y(x)$ et $y_2(x) = \frac{dy}{dx}$ qui composent le vecteur Y :

$$\frac{dY}{dx} = \begin{pmatrix} y_2(x) \\ \mathrm{e}^{-\mathrm{x}} - a y_2(x) - b\,y_1(x) \end{pmatrix} = \begin{pmatrix} F_1(x,Y) \\ F_2(x,Y) \end{pmatrix} = F(x,Y)$$

4.5.2 Écriture de l'équation sous Matlab

La seconde étape consiste à écrire une fonction MATLAB dans un fichier séparé décrivant le système différentiel. La fonction doit être de la forme :

```
function dY=edofct(x,Y,P)
```

où **`edofct`** est le nom (choisi arbitrairement) de la fonction MATLAB codant la fonction mathématique $F(x,Y)$. Le résultat **`dY`** est un vecteur colonne contenant les composantes $F_1(x,Y)$ et $F_2(x,Y)$ de la fonction $F(x,Y)$.

Même si $F(x,Y)$ ne dépend pas explicitement de x, les deux premiers paramètres d'entrée de la fonction sont obligatoires. Le troisième paramètre **`P`** est optionnel et permet de passer des paramètres de l'utilisateur.

Dans le cas de notre exemple avec a=1 et b=2, le code associé est de la forme :

```
function dY=ode1(x,Y)

% Y' = F(x,Y(x))
A = 1; b=2;
dY(1,:) = Y(2);
dY(2,:) = exp(-x) - a*Y(2) - b*Y(1);

end
```

4.5.3 Résolution numérique sous Matlab

La troisième étape consiste à choisir le solveur MATLAB devant résoudre le problème. Il existe pour cela deux types de solveurs : les solveurs pour les problèmes classiques (**ode23**, **ode45)** et ceux pour les problèmes dits « raides » (**ode23s**, **ode15s)** que nous aborderons en exercices dirigés.

Tous ces solveurs sont basés sur la méthode de Runge-Kutta d'ordre 2 (**ode23**) ou 4 (**ode45)** sont dits multi-pas, c'est à dire que le pas d'intégration est choisi automatiquement par l'algorithme afin de garantir la stabilité de la solution. La syntaxe générique d'appel du solveur est :

```
[t,Y] = odeXX( edofct, TSPAN,Y0)
```

où **odeXX** désigne le solveur choisi, **edofct** désigne le nom de la fonction MATLAB (.m) associée au système différentiel, **TSPAN** désigne les instants initiaux et finaux pour l'intégration et **Y0** désigne le vecteur de conditions initiales.

Les paramètres de sortie sont le vecteur **t** désignant les nœuds de l'intervalle **TSPAN** où a été calculée la solution et la matrice **Y** contenant les valeurs de la solution (toutes les composantes) à chaque instant **t.**

Plusieurs méthodes pour le vecteur **TSPAN** sont permises pour la résolution :

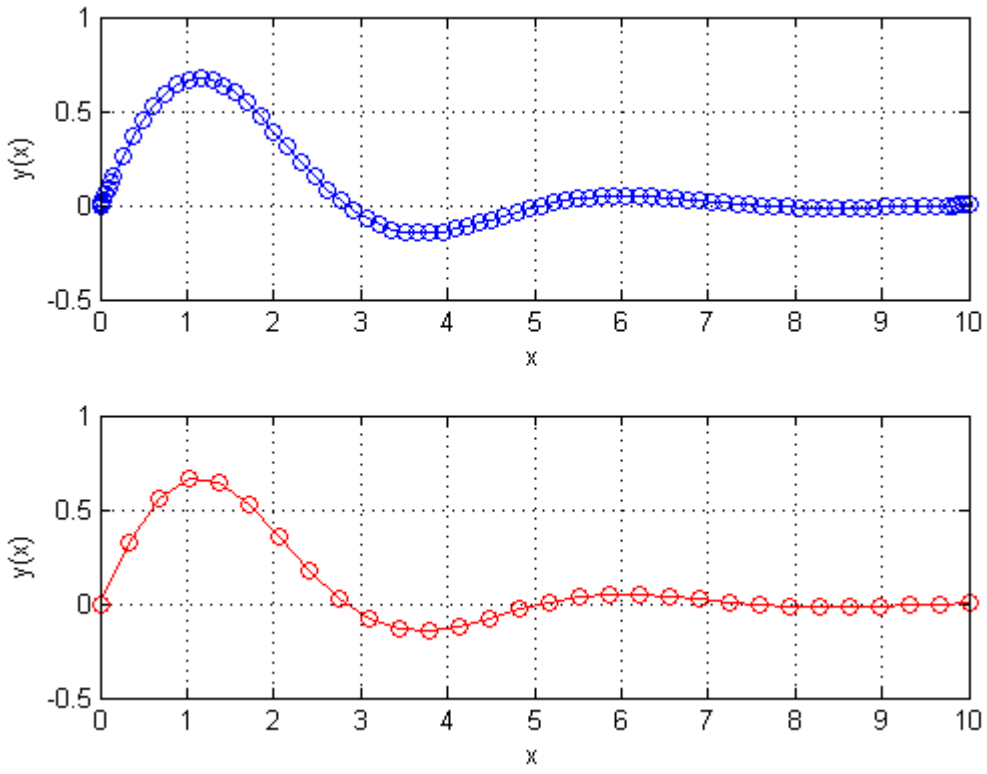
- **TSPAN = [t0,t1]** où **t0** et **t1** sont deux constantes. Dans ce cas, les pas de temps sont choisis par le solveur et stockées dans la variable t.
- **TSPAN = linspace(t0,t1,N).** Dans ce cas, les pas de temps pour la résolution sont imposés mais le solveur peut au besoin les ajuster linternement. Dans ce cas, la sortie du solveur est données aux instants **t** correspondants au vecteur **TSPAN**

Dans notre cas, le solveur peut s'écrire :

```
% Conditions initiales Y0 = [y1(x) ; y2(x)] = [y(x) ; y'(x)]
Y0 = [0 ; 1];
TSPAN = [0 10];
[t,Y] = ode45( ode1, TSPAN,Y0)
subplot(211)
plot(t,Y(:,1),'o-')

TSPAN = linspace(0,10,30);
[t,Y] = ode45( ode1, TSPAN,Y0)
subplot(212)
plot(t,Y(:,1),'ro-')
```

Et la solution est donnée ci-dessous. Notez que dans le premier cas, le pas de temps n'est pas contant et est déterminé automatiquement par le solveur. C'est également le cas dans le deuxième graphique, mais seules les solutions au pas de temps de **TSPAN** sont sauvegardées. Il est également important de noter que les solutions sont identiques et ne dépendent pas du pas d'intégration et du vecteur **TSPAN** choisi.



4.5.4 Paramètres avancés

Un argument supplémentaire peut être inséré dans la ligne du solveur. Il s'agit d'un paramètre **`options`** qui peut être facilement modifié à l'aide de la fonction **`optimset`**:

```
options = optimset('RelTol',1e-6,'AbsTol',1e-4)
[t, Y] = ode45( edofct, TSPAN,Y0,options)
```

Par exemple, la ligne de code ci-dessus permet de garantir une tolérance relative de 10^{-6} et une tolérance 10^{-4} absolue de entre la solution numérique et la solution exacte. Pour plus d'informations sur les options possibles, vous pouvez consulter l'aide de la fonction **`optimset.`**

Pour certains problèmes, un ou des arguments doivent être envoyés à la fonction calculant la dérivée du vecteur Y (fonction **`ode1`** dans l'exemple précédent). Dans ce cas, il est nécessaire de créer une fonction alias déclarée à l'aide du symbole **`@.`** Voici l'exemple avec l'équation précédente mais dans le cas où les deux variables (a et b) sont données dans le programme contenant le solveur :

solveur.m	ode1.m

`a = 1;`
`b = 2;`
`% Fonction alias`
`ode2 = @(x,Y) ode1(x,Y,a,b)`
 	
`% Consitions initiales Y0`
`Y0 = [0 ; 1];`
`TSPAN = [0 10];`
 	
`% Solveur ODE`
`[t,Y] = ode45( ode2, TSPAN,Y0)`
	`function dY=ode1(x,Y,a,b)`
	
	`% Y' = F(x,Y(x))`
	
	`dY(1,:) = Y(2);`
	
	`dY(2,:) = exp(-x) - a*Y(2) - b*Y(1);`
	
	`end`

Une autre méthode consisterait à utiliser des variables globales qui sont partagées par les différents programmes. Cette méthode doit être utilisée avec attention car peut parfois mener à des problèmes d'interprétation et de compréhension dans le cas où les variables (a et b dans notre cas) sont remplacées à de multiples reprises dans différents sous-programmes.

solveur.m	ode1.m

`global a b`
 	
`a = 1; b = 2;`
 	
`% Consitions initiales Y0`
`Y0 = [0 ; 1];`
`TSPAN = [0 10];`
 	
`% Solveur ODE`
`[t,Y] = ode45( ode1, TSPAN,Y0)`
	`function dY=ode1(x,Y)`
	
	`% Y' = F(x,Y(x))`
	
	`global a b`
	
	`dY(1,:) = Y(2);`
	`dY(2,:) = exp(-x) - a*Y(2) - b*Y(1);`
	
	`end`

Tous ces problèmes peuvent également être contournés en utilisant une fonction alias qui permet de s'affranchir d'une fonction externe (ode1.m) en intégrant l'équation différentielle dans le code du solveur. Cette méthode, bien que très efficace pour un utilisateur confirmé, vous est déconseillée en premier temps.

```
a = 1; b = 2;
Y0 = [0 ; 1];
TSPAN = [0 10];

% Fonction alias
ode2 = @(x,Y) [Y(2); exp(-x) - a*Y(2) - b*Y(1)];

% Solveur ODE
[t,Y] = ode45( ode1, TSPAN,Y0)
```

4.6 Synthèse

Problème aux valeurs initiales d'ordre 1 :

$$\frac{dy}{dx} = F(x, y)$$
$$y(x_0) = y_0$$

méthode d'Euler (méthode d'ordre 1)**:**

$$x_n = x_0 + nh\,,\ n = 0,1,....$$

$$y_{n+1} = y_n + F(x_n, y_n)h$$

méthode d'Euler améliorée (méthode d'ordre 2)**:**

$$x_n = x_0 + nh\,,\ n = 0,1,....$$

$$y_{n+1} = y_n + \frac{h}{2}\left[F(x_n, y_n) + F(x_{n+1}, y_n + hF(x_n, y_n))\right]$$

méthode de Runge-Kutta (méthode d'ordre 4)**:**

$$x_n = x_0 + nh\,,\ n = 0,1,....$$

$$y_{n+1} = y_n + \frac{h}{6}\left[k_1 + 2k_2 + 2k_3 + k_4\right]$$

avec $k_1 = F(x_n, y_n)$,

$$k_2 = F(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1)\,,$$

$$k_3 = F(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2)\,,$$

$$k_4 = F(x_n + h, y_n + hk_3)\,.$$

Système de *n* problèmes aux valeurs initiales d'ordre 1 :

$$\frac{dy_1}{dx} = F_1(x, y_1, y_2, \ldots y_n) \quad y_1(x_0) = y_0^{(1)}$$
$$\frac{dy_2}{dx} = F_2(x, y_1, y_2, \ldots y_n) \quad y_2(x_0) = y_0^{(2)}$$
$$\ldots$$
$$\frac{dy_n}{dx} = F_n(x, y_1, y_2, \ldots y_n) \quad y_n(x_0) = y_0^{(n)}$$

méthode de Runge-Kutta: sous forme vectorielle,

$$x_n = x_0 + nh\,,\ n = 0,1,\ldots.$$

$$\mathbf{y}_{m+1} = \mathbf{y}_m + \frac{h}{6}\left[\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right]$$

avec

$$\mathbf{y}_m = \begin{Bmatrix} y_{1,m} \\ y_{2,m} \\ \ldots \\ y_{n,m} \end{Bmatrix}, \mathbf{y}_0 = \begin{Bmatrix} y_0^{(1)} \\ y_0^{(2)} \\ \ldots \\ y_0^{(n)} \end{Bmatrix}$$

$$\mathbf{F}(x_m, \mathbf{y}_m) = \begin{Bmatrix} F_1(x_m, y_{1,m}, y_{2,m}, \ldots y_{n,m}) \\ F_2(x_m, y_{1,m}, y_{2,m}, \ldots y_{n,m}) \\ \ldots \\ F_n(x_m, y_{1,m}, y_{2,m}, \ldots y_{n,m}) \end{Bmatrix}$$

$$\mathbf{k}_1 = \mathbf{F}(x_m, \mathbf{y}_m)$$

$$\mathbf{k}_2 = \mathbf{F}(x_m + \frac{h}{2}, \mathbf{y}_m + \frac{h}{2}\mathbf{k}_1)$$

$$\mathbf{k}_3 = \mathbf{F}(x_m + \frac{h}{2}, \mathbf{y}_m + \frac{h}{2}\mathbf{k}_2)$$

$$\mathbf{k}_4 = \mathbf{F}(x_m + h, \mathbf{y}_m + h\mathbf{k}_3)$$

Chapitre 5 : Fonctions multi-variables

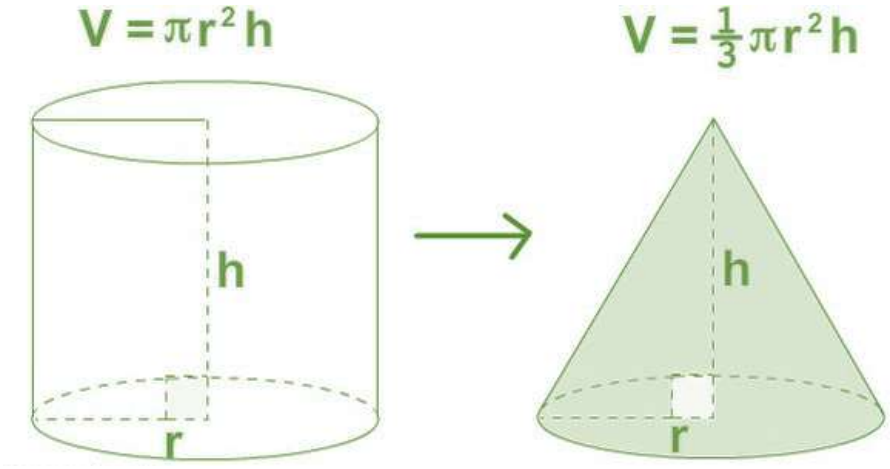
5.1. Introduction et mise en contexte

Jusqu'à présent, nous n'avons considéré que des fonctions réelles $y = f(x)$ dépendant d'*une seule* variable indépendante *x*. En pratique, une grandeur *y* peut dépendre de *plus d'une* variable indépendante.

Les fonctions réelles de plusieurs variables réelles ont une importance cruciale dans de multiples branches du génie. Les notions relatives aux fonctions multi-variables peuvent être appliquées à la dynamique pour les notions de travail d'une force, d'énergie potentielle, de fonction potentielle, de champ conservatif, mais également à bien d'autres problèmes, comme l'optimisation, la mécanique des milieux continus, etc.

Les deux exemples suivants sont des illustrations simples de fonctions à plusieurs variables :

- Le volume *V* d'un réservoir de forme cylindrique ou conique, de rayon *r* et de hauteur *h* est donné par :



Le volume *V* est donc une fonction des 2 variables indépendantes *r* et *h* que l'on peut les faire varier de façon indépendante), on écrit donc :

$$V = V(r, h)$$

- En thermodynamique, il existe une relation entre la pression *p* (Pa) d'un gaz, sa température *T* (°K) et le volume V (m3) qu'il occupe. Pour 1 mole d'un gaz dit parfait,

$$pV = RT$$

où *R* est la constante des gaz parfaits (*R*=8.32J/mole.°K). De cette relation, on peut isoler la pression,

$$p = \frac{RT}{V} = p(V,T)$$

La pression d'un gaz parfait est donc une fonction de 2 variables indépendantes - sa température et son volume. Le graphique ci-dessous montre la *surface représentative* de la pression *p* (Pa) en fonction du volume *V* (m^3) entre 0.1 m^3 et 0.2 m^3 et de la température *T* (°K) entre 250 °K et 400 °K, (Quelques modes de représentation graphique de fonctions à plusieurs variables seront discutés plus loin). On voit sur ce graphique que la pression est une fonction croissante de la température, alors qu'elle est une fonction décroissante du volume.

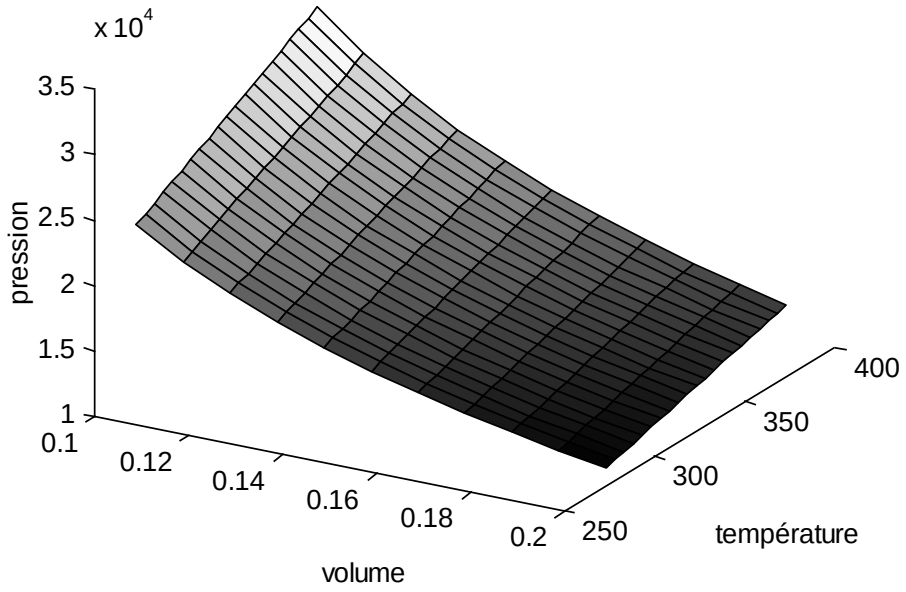


Figure 5.1 : Surface représentative de la pression en fonction du volume et de la température pour un gaz parfait

5.2. Définitions

Soit *D* un ensemble de *n*-uplets réels (*D* est un sous-ensemble de l'ensemble des *n*-uplets réels, R^n).
On définit la fonction réelle *f* de *n* variables réelles,

$$\begin{aligned} f \quad &: D \subset R^n \to R \\ &(x_1, x_2, \ldots x_n) \mapsto w = f(x_1, x_2, \ldots x_n) \end{aligned}$$

On appelle $x_1, x_2, \ldots x_n$ les *n* variables réelle **indépendantes**, *w* est la variable réelle **dépendante.**

Le domaine de définition de la fonction *f* est l'ensemble des valeurs de $(x_1, x_2, \ldots x_n)$ pour lesquelles la valeur de *w* est définie. On appellera donc domaine de définition D_f de *f* le sous-ensemble de *D* tel que

$$D_f = \left\{(x_1, x_2, \ldots x_n) \in R^n \,/\, \exists\, z \in R \text{ tel que } z = f(x_1, x_2, \ldots x_n)\right\}$$

Exemples :

- $w = \sqrt{x_1 - 2{x_2}^2}$; $D_f = \left\{(x_1, x_2) \,/\, x_1 \geq 2{x_2}^2\right\}$
- $w = \dfrac{\ln x_3}{\sqrt{{x_1}^2 + {x_2}^2}}$; $D_f = \left\{(x_1, x_2, x_3) \,/\, (x_1, x_2) \neq (0,0) \text{ et } x_3 > 0\right\}$
- $w = \cos(x_1 x_2 x_3)$; $D_f = R^3$

Remarque : usuellement, on utilisera les conventions suivantes pour nommer les variables dans le cas de fonctions de 2 ou 3 variables réelles :

- Fonctions de 2 variables :

$$\begin{aligned} f \quad &: D \subset R^2 \to R \\ &(x, y) \mapsto z = f(x, y) \end{aligned}$$

- Fonctions de 3 variables :

$$\begin{aligned} f \quad &: D \subset R^3 \to R \\ &(x, y, z) \mapsto w = f(x, y, z) \end{aligned}$$

5.2 Calcul et représentation graphique des fonctions multi-variables

5.2.1 Méthodes de calcul des fonctions multi-variables

Considérons tout d'abord le cas d'une fonction de 2 variables $f:(x,y)\mapsto z=f(x,y)$ et regardons plus particulièrement le cas la fonction suivante sur le domaine $x\in[-2;2]$, $y\in[-2;2]$

$$f:D\subset R^2\to R$$
$$(x,y)\mapsto z=xe^{-x^2-y^2}$$

Afin de calculer la valeur $z=f(x)$ pour chaque valeur de x et y , il est courant d'employer deux méthodes, à savoir une méthode de boucle ou une méthode matricielle. Ces deux techniques sont équivalentes, bien que l'approche matricielle soit plus rapide sous MATLAB, alors que l'approche par boucle sera plus efficace en C++. Les deux implantations sous MATLAB sont données plus bas :

Méthode par boucle

```
% Creation des vecteurs lignes X et Y
X = linspace(-2,2,20);
Y = linspace(-2,2,30);


% Calcul par boucle
for ix=1:length(X)
  for iy=1:length(Y)
    Z(iy,ix) = X(ix)*exp(-X(ix)^2-Y(iy)^2)
  end
end
```

Méthode matricielle

```
% Creation des vecteurs lignes X et Y
X = linspace(-2,2,20);
Y = linspace(-2,2,30);

% Creation de matrices pour X et Y
[Xmat,Ymat] = meshgrid(X,Y);

% Calcul matriciel
Z = Xmat .* exp(-Xmat.^2-Ymat.^2)
```

Dans notre exemple, la matrice **Z** a donc une taille de 30 x 20 car l'axe Y correspond bien aux colonnes de la matrice (attention à ne pas inverser lignes et colonnes !).

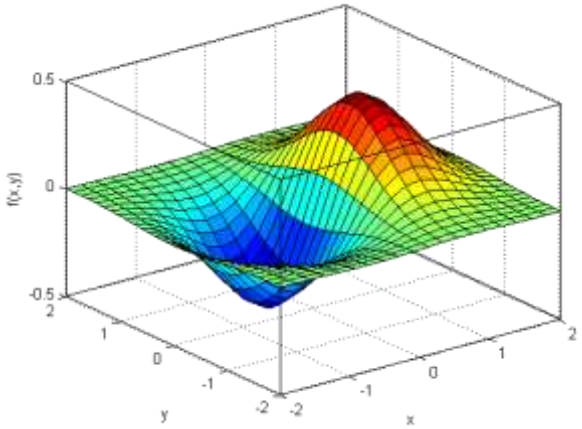
Il est important de noter que la méthode matricielle, bien que très économe en termes de lignes de calcul est plus compliquée à mettre en œuvre. Il est important de noter que dans le cas matriciel, les opérations sont effectuées sur les matrices `Xmat` et `Ymat` (qui sont toutes les deux de taille 30 x 20) et que les opérations sont effectuées point par point à l'aide des opérateurs (`.*`) et (`.^`) au lieu de (`*`) et (`^`), ce qui est couramment appelé une opération point par point ou élément par élément (l'aide de **.*** sous MATLAB est bien détaillée).

5.2.1 Représentation des fonctions multi-variables

Habituellement, on peut représenter graphiquement une fonction de 2 variables de quatre manières :

- par une **surface** d'équation $z = f(x,y)$ dans le repère cartésien (*x,y,z*). Cette surface est donc l'ensemble des points $(x, y, f(x,y))$ dans le repère cartésien (*x,y,z*). La fonction MATLAB ainsi que la représentation graphique permettant de réaliser cette opération est :

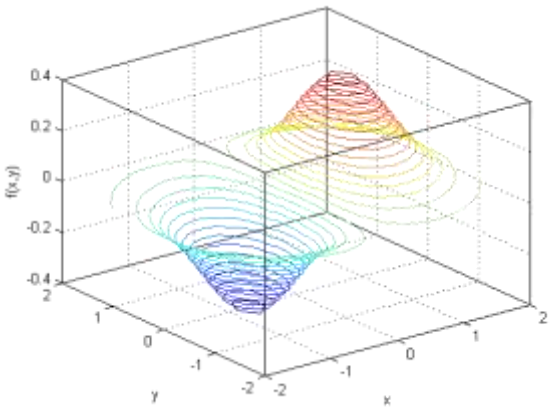
```
surf(X,Y,Z)
xlabel('x')
ylabel('y')
zlabel('f(x,y)')
box on;
grid on;
```



Cette représentation très visuelle et facile d'interprétation peut cependant devenir très couteuse en temps de calcul lorsque l'on possède un nombre élevé de points. Dans ce cas, on préférera des techniques 2D (image et contours)

- par un ensemble de **lignes de contour** d'équation $f(x,y) = c$ et $Z = C$, (où *c* est un réel) dans l'espace ; ces lignes représentent dans l'espace l'ensemble des points (*x,y,z*) tels que $f(x,y) = c$. La fonction MATLAB ainsi que la représentation graphique permettant de réaliser cette opération est :

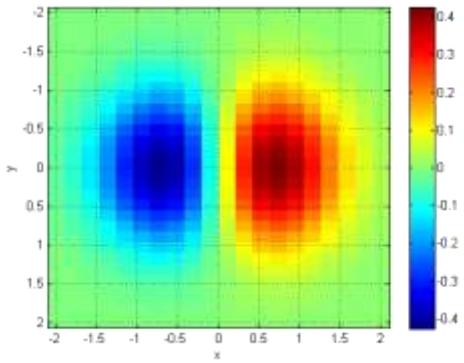
```
contour3(X,Y,Z,30)
xlabel('x')
ylabel('y')
zlabel('f(x,y)')
box on;
grid on;
```



Ici on demande 30 lignes de contour pour la représentation. On pourrait aussi définir un vecteur `C = linspace(Cmin,Cmax,N)` afin d'obtenir N contours distribués entre les valeurs `Cmin` et `Cmax`.

- par une **image** projetant la valeur de $f(x,y)$ pour chaque couple de valeurs (x,y) . L'image peut être vue comme une projection dans le plan (*x,y*) de la surface $z = f(x,y)$. La fonction MATLAB ainsi que la représentation graphique permettant de réaliser cette opération est :

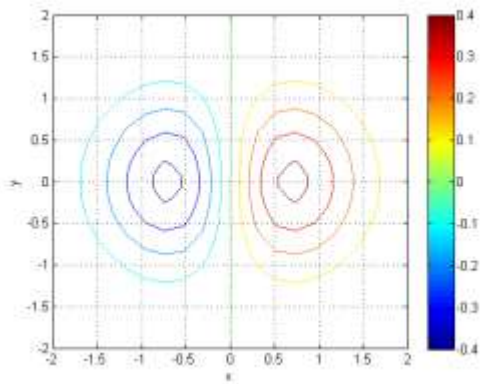
```
imagesc(X,Y,Z)
xlabel('x')
ylabel('y')
box on;
grid on;
colorbar('vert')
```



Cette méthode très simple permet de représenter une fonction à deux variables dans le cas d'un grand nombre de points. On peut aussi effectuer de l'interpolation afin de lisser les courbes au besoin.

- par un ensemble de **lignes de niveau** d'équation $f(x,y) = c$, (où *c* est un réel) dans le plan (*x,y*) ; ces lignes représentent dans le plan l'ensemble des points (*x,y*) tels que $f(x,y) = c$. La fonction MATLAB ainsi que la représentation graphique permettant de réaliser cette opération est :

```
contour(X,Y,Z)
xlabel('x')
ylabel('y')
box on;
grid on;
colorbar('vert')
```



Pour les deux dernières représentations, il est usuel de rajouter une barre de couleur (`colorbar`) qui permet de connaître les valeurs associées aux couleurs de pixels.

Dans le cas de fonction à 3 variables (aussi nommées champs vectoriels), c'est à dire des fonctions dont la valeur dépend à la fois de x, y et z, les outils de MATLAB s'avèrent plus compliqués et on a souvent recours à des représentations en coupe afin de bien visualiser le volume représenté. C'est le cas dans le domaine de l'imagerie médicale (IRM par exemple) pour lequel on associe, pour chaque pixel de l'espace (on parle de voxel en 3D), une valeur numérique. La représentation usuelle est ainsi de précéder par coupes et de représenter une image pour différentes valeurs de z.

5.3 Dérivées partielles

5.3.1 Définition

Comme dans le cas des fonctions d'une seule variable, il est important de pouvoir introduire la notion de « dérivée » d'une fonction de plusieurs variables. Dans ce dernier cas, il existe maintenant plusieurs variables indépendantes par rapport auxquelles on peut dériver la fonction.

En gardant constantes toutes les variables indépendantes sauf une, la fonction de plusieurs variables devient une fonction d'une seule variable, et il est alors possible de dériver par rapport à cette variable : c'est la notion de dérivée *partielle* que l'on définit de la sorte

Soit la fonction à n variables :

$$\begin{aligned} f \quad &: D \subset R^n \to R \\ &(x_1, x_2, \ldots x_n) \mapsto w = f(x_1, x_2, \ldots x_n) \end{aligned}$$

On définit la dérivée partielle de *f* par rapport à la variable indépendante $x_i \ (i = 1,2,\ldots n)$, au point $\left(x_1^0, x_2^0, \ldots x_n^0\right)$ comme la limite (si elle existe) :

$$\lim_{h \to 0} \frac{f(x_1{}^0, x_2{}^0, \ldots x_i{}^0 + h, \ldots x_n{}^0) - f(x_1{}^0, x_2{}^0, \ldots x_i{}^0, \ldots x_n{}^0)}{h} = \frac{\partial f}{\partial x_i}(x_1^0, x_2^0, \ldots x_n^0)$$

Remarquons l'utilisation du symbole ∂ (« d rond ») dans le cas des dérivées partielles. On peut interpréter la dérivée partielle $\frac{\partial f}{\partial x_i}(x_1^0, x_2^0, \ldots x_n^0)$ comme le taux d'accroissement de la fonction *f* par rapport à la variable indépendante x_i au point $\left(x_1^0, x_2^0, \ldots x_n^0\right)$. En pratique, pour calculer la dérivée partielle $\frac{\partial f}{\partial x_i}(x_1^0, x_2^0, \ldots x_n^0)$, on traitera toutes les variables autres que x_i comme des constantes et l'on utilisera les règles de dérivation des fonctions d'une seule variable pour dériver par rapport à x_i.

5.3.2 Exemples

- $f(x,y) = 3x^2 + 4\sin y - xe^y + 1$. Calculons $\frac{\partial f}{\partial x}$ et $\frac{\partial f}{\partial y}$ au point (1,0).

Pour calculer la dérivée partielle $\frac{\partial f}{\partial x}$ il suffit de considérer y comme une constante et d'utiliser les règles connues de la dérivation des fonctions d'une seule variable.

$\frac{\partial f}{\partial x} = 6x - e^y$ donc $\frac{\partial f}{\partial x}(1,0) = 6 - 1 = 5$

De la même façon, $\frac{\partial f}{\partial y} = 4\cos y - xe^y$ donc $\frac{\partial f}{\partial y}(1,0) = 4 - 1 = 3$

- $f(x,y,z) = x^2 + 3y^2 + z^2 - xy + 4xz + yz - 2$. Calculons $\frac{\partial f}{\partial x}$, $\frac{\partial f}{\partial y}$ et $\frac{\partial f}{\partial z}$.

$\frac{\partial f}{\partial x} = 2x - y + 4z$; $\frac{\partial f}{\partial y} = 6y - x + z$; $\frac{\partial f}{\partial z} = 2z + 4x + y$.

- Considérons à nouveau la relation thermodynamique des gaz parfaits $p = \frac{RT}{V} = p(V,T)$, où *p* est la pression du gaz, *V* est son volume et *T* sa température (*R*=8.32J/mole.°K). Supposons des conditions $V = 0.15m^3$, $T = 300^0 K$. Cherchons alors à calculer le taux de variation de la pression avec le volume du gaz (à température constante), ainsi que le taux de variation de la pression avec la température du gaz (à volume constant), au point $V = 0.15m^3$, $T = 300^0 K$. Ces taux de variation sont donnés par les dérivées partielles $\frac{\partial p}{\partial V}(0.15,300)$ et $\frac{\partial p}{\partial T}(0.15,300)$.

$$\frac{\partial p}{\partial V}(V,T) = -\frac{RT}{V^2} \qquad \text{donc} \qquad \frac{\partial p}{\partial V}(0.15,300) = -\frac{8.32 \times 300}{0.15^2} = -110933 Pa/m^3.$$

(Le signe négatif montre que la pression décroît avec le volume). De même,

$$\frac{\partial p}{\partial T}(V,T) = \frac{R}{V} \qquad \text{donc} \qquad \frac{\partial p}{\partial T}(0.15,300) = \frac{8.32}{0.15} = 55.47 Pa/^0 K$$

(La pression croît avec la température).

La dérivée partielle $\frac{\partial p}{\partial V}(0.15,300)$ représente la pente de la droite qui est tangente à la surface $p = \frac{RT}{V} = p(V,T)$ au point $V = 0.15m^3$, $T = 300^0 K$ et qui est contenue dans le plan $T = cte$. Cette droite a donc pour équation dans l'espace (*V,T,p*)

$$D_v : \begin{cases} p = 16640 - 110933(V - 0.15) \\ T = 300 \end{cases}$$

De même, la dérivée partielle $\frac{\partial p}{\partial V}(0.15,300)$ représente la pente de la droite qui est tangente à la surface $p = \frac{RT}{V} = p(V,T)$ au point $V = 0.15m^3$, $T = 300^0 K$ et qui est contenue dans le plan $V = cte$. Cette droite a donc pour équation dans l'espace (*V,T,p*)

$$D_T : \begin{cases} p = 16640 + 55.47(T - 300) \\ V = 0.15 \end{cases}$$

Ces droites sont représentées graphiquement ci-dessous avec la surface $p = \frac{RT}{V} = p(V,T)$.

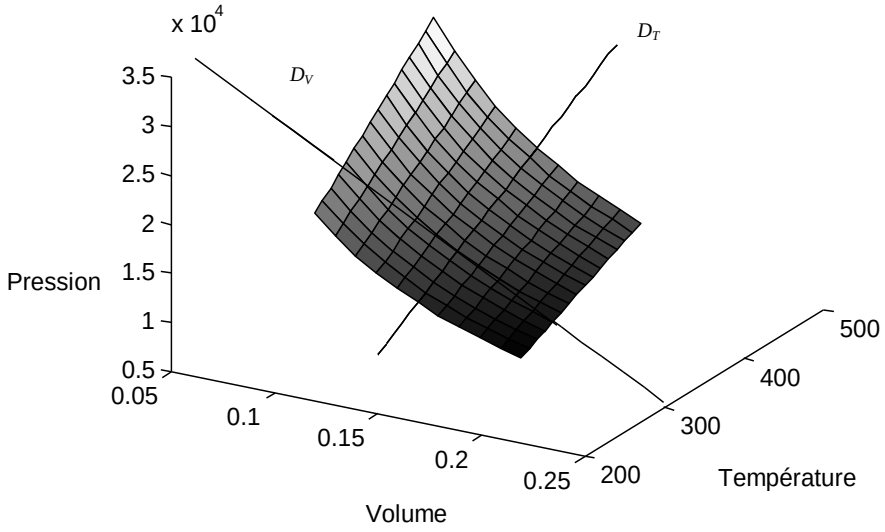
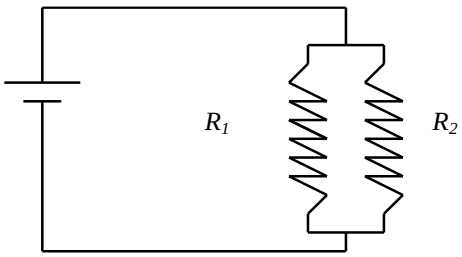


Figure.5.2 : Surface et droites tangentes en un point

Retenons de manière générale que pour une fonction de deux variables $f(x,y)$, la dérivée partielle $\frac{\partial f}{\partial x}(x_0,y_0)$ représente la pente de la droite tangente à la surface $z=f(x,y)$ au point (x_0,y_0) et contenue dans le plan $y=y_0$. De même, la dérivée partielle $\frac{\partial f}{\partial y}(x_0,y_0)$ représente la pente de la droite tangente à la surface $z=f(x,y)$ au point (x_0,y_0) et contenue dans le plan $x=x_0$.

- Une application importante des dérivées partielles de fonctions multivariables est l'étude de la sensibilité d'une fonction multivariables par rapport à ses différents variables indépendantes. Considérons l'exemple simple suivant :<

Soit un circuit électrique comportant deux résistances $R_1=30\Omega$ et $R_2=90\Omega$ en parallèle.



On désire savoir si la résistance équivalente formée par R_1 et R_2 est plus sensible aux variations de R_1 ou aux variations de R_2. Pour cela, on calcule la résistance équivalente *R*,

$$\frac{1}{R}=\frac{1}{R_1}+\frac{1}{R_2}, \qquad \text{d'où} \qquad R=\frac{R_1R_2}{R_1+R_2}=R(R_1,R_2)$$

Le taux de variation de *R* par rapport à R_1 est $\frac{\partial R}{\partial R_1}=\frac{{R_2}^2}{(R_1+R_2)^2}$; le taux de variation de *R* par rapport à R_2 est $\frac{\partial R}{\partial R_2}=\frac{{R_1}^2}{(R_1+R_2)^2}$. Comme $R_2>R_1$, on en déduit que $\frac{\partial R}{\partial R_1}>\frac{\partial R}{\partial R_2}$; la résistance équivalente est donc plus sensible aux variations de la plus faible des deux résistances, R_1.

5.3.3 Recherche d'extremums d'une fonction multi-variables

Dans le cas des fonctions d'une variable réelle, la dérivée est utilisée pour la recherche d'extremums. De façon similaire, les dérivées partielles seront utiles pour la recherche d'extremums de fonctions multi-variables. Nous devons tout d'abord définir la notion de point critique d'une fonction multi-variables $f(x_1, x_2, \ldots x_n)$. $(x_1, x_2, \ldots x_n)$ est appelé un **point critique** de *f* lorsque toutes les dérivées partielles de *f* s'annulent en $(x_1, x_2, \ldots x_n)$:

$$\frac{\partial f}{\partial x_i}(x_1, x_2, \cdots x_n) = 0 \quad \forall i = 1, \cdots n$$

Exemples :

Cherchons les points critiques de $f(x, y) = x^3 + y^3 - 12x - 12y + 6$.

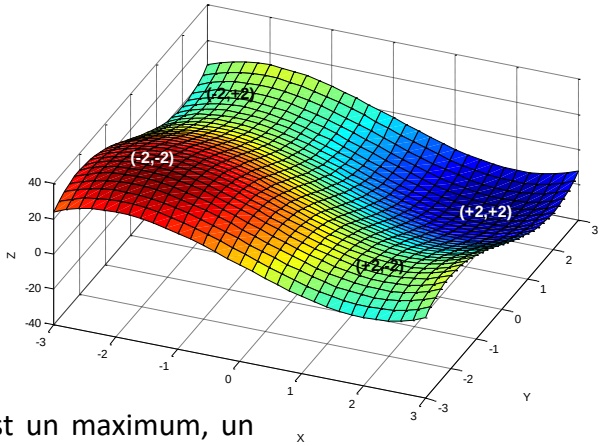
$\frac{\partial f}{\partial x} = 3x^2 - 12$; $\quad \frac{\partial f}{\partial x} = 0 \Leftrightarrow 3x^2 - 12 = 0 \Leftrightarrow x^2 = 4 \Leftrightarrow x = \pm 2$.

$\frac{\partial f}{\partial y} = 3y^2 - 12$. $\quad \frac{\partial f}{\partial y} = 0 \Leftrightarrow 3y^2 - 12 = 0 \Leftrightarrow y^2 = 4 \Leftrightarrow y = \pm 2$.

Il existe donc 4 points critiques : (-2,-2); (-2,+2); (+2,-2); (+2,+2).

La fonction $f(x, y) = x^3 + y^3 - 12x - 12y + 6$ est représentée graphiquement sur la figure ci-dessous. On peut observer que ces 4 points critiques s'interprètent de diverses façons :

- **(-2,-2) est un maximum;**
- **(+2,+2) est un minimum;**
- **(-2,+2) et (+2,-2) sont des points de selle**.



La suite de cette section expliquera en détail comment déterminer la nature des points critiques.

Examinons à présent dans quels cas un point critique est un maximum, un minimum ou encore un point de selle. Nous nous intéresserons à cette question **uniquement dans le cas des fonctions de 2 variables** $f(x, y)$. On dispose pour ceci du théorème suivant :

si $\frac{\partial f}{\partial x}(a,b)=0$ et $\frac{\partial f}{\partial y}(a,b)=0$, alors :

- (a,b) est un maximum local si $\frac{\partial^2 f}{\partial x^2}(a,b)<0$ et $\frac{\partial^2 f}{\partial x^2}(a,b)\frac{\partial^2 f}{\partial y^2}(a,b)-\left[\frac{\partial^2 f}{\partial x\partial y}(a,b)\right]^2>0$;
- (a,b) est un minimum local si $\frac{\partial^2 f}{\partial x^2}(a,b)>0$ et $\frac{\partial^2 f}{\partial x^2}(a,b)\frac{\partial^2 f}{\partial y^2}(a,b)-\left[\frac{\partial^2 f}{\partial x\partial y}(a,b)\right]^2>0$;
- (a,b) est un point de selle si $\frac{\partial^2 f}{\partial x^2}(a,b)\frac{\partial^2 f}{\partial y^2}(a,b)-\left[\frac{\partial^2 f}{\partial x\partial y}(a,b)\right]^2<0$;
- On ne peut pas conclure si $\frac{\partial^2 f}{\partial x^2}(a,b)\frac{\partial^2 f}{\partial y^2}(a,b)-\left[\frac{\partial^2 f}{\partial x\partial y}(a,b)\right]^2=0$.

Reprenons l'exemple de la fonction $f(x,y)=x^3+y^3-12x-12y+6$, dont les points critiques sont $(\pm 2,\pm 2)$. On calcule

$$\frac{\partial^2 f}{\partial x^2}=6x\,; \qquad \frac{\partial^2 f}{\partial y^2}=6y\,; \qquad \frac{\partial^2 f}{\partial x\partial y}=0\,. \qquad \frac{\partial^2 f}{\partial x^2}\frac{\partial^2 f}{\partial y^2}-\left[\frac{\partial^2 f}{\partial x\partial y}\right]^2=36xy\,.$$

- Au point $(-2,-2)$, $\frac{\partial^2 f}{\partial x^2}=6x=-12<0$ et $\frac{\partial^2 f}{\partial x^2}\frac{\partial^2 f}{\partial y^2}-\left[\frac{\partial^2 f}{\partial x\partial y}\right]^2=36xy=144>0$ -> maximum local;
- Au point $(-2,+2)$, $\frac{\partial^2 f}{\partial x^2}=6x=-12<0$ et $\frac{\partial^2 f}{\partial x^2}\frac{\partial^2 f}{\partial y^2}-\left[\frac{\partial^2 f}{\partial x\partial y}\right]^2=36xy=-144<0$ -> point de selle;
- Au point $(+2,-2)$, $\frac{\partial^2 f}{\partial x^2}=6x=12>0$ et $\frac{\partial^2 f}{\partial x^2}\frac{\partial^2 f}{\partial y^2}-\left[\frac{\partial^2 f}{\partial x\partial y}\right]^2=36xy=-144<0$ -> point de selle;
- Au point $(+2,+2)$, $\frac{\partial^2 f}{\partial x^2}=6x=12>0$ et $\frac{\partial^2 f}{\partial x^2}\frac{\partial^2 f}{\partial y^2}-\left[\frac{\partial^2 f}{\partial x\partial y}\right]^2=36xy=144>0$ -> minimum local.

Le tracé de la fonction $f(x,y)=x^3+y^3-12x-12y+6$ plus haut confirme ces résultats.

5.3.4 Dérivées partielles d'ordre supérieur

Soit la fonction :

$$\begin{aligned} f \quad &: D \subset R^n \to R \\ &(x_1, x_2, \ldots x_n) \mapsto w = f(x_1, x_2, \ldots x_n) \end{aligned}$$

On a défini la dérivée partielle de f par rapport à la variable indépendante $x_i \ (i = 1,2,\ldots n)$, $\frac{\partial f}{\partial x_i}(x_1, x_2, \ldots x_n)$. Cette dérivée partielle est à son tour une fonction de *n* variables indépendantes, qui peut être dérivée par rapport à chacune de ses variables indépendantes, $x_j \ (j = 1,2,\ldots n)$.

On définit ainsi (lorsqu'elle existe), la dérivée partielle d'ordre 2,

$$\frac{\partial}{\partial x_j}\left(\frac{\partial f}{\partial x_i}\right) = \frac{\partial^2 f}{\partial x_j \partial x_i}(x_1, x_2, \ldots x_n).$$

Ainsi de suite, on peut construire (lorsqu'elles existent) les dérivées partielles d'ordre quelconque d'une fonction de plusieurs variables.

Dans le cas d'une fonction de deux variables,

$$\begin{aligned} f \quad &: D \subset R^2 \to R \\ &(x, y) \mapsto z = f(x, y) \end{aligned}$$

on définit ainsi quatre dérivées d'ordre 2:

$$\frac{\partial}{\partial x}\left(\frac{\partial f}{\partial x}\right) = \frac{\partial^2 f}{\partial x^2}(x,y) \ ; \qquad \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial y}\right) = \frac{\partial^2 f}{\partial x \partial y}(x,y) \ ; \qquad \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial x}\right) = \frac{\partial^2 f}{\partial y \partial x}(x,y) \ ;$$

$$\frac{\partial}{\partial y}\left(\frac{\partial f}{\partial y}\right) = \frac{\partial^2 f}{\partial y^2}(x,y)$$

.Les dérivées secondes $\frac{\partial^2 f}{\partial x^2}(x,y)$ et $\frac{\partial^2 f}{\partial y^2}(x,y)$ représentent les courbures de la surface $z = f(x,y)$ dans les plans $y = cte$ et $x = cte$, respectivement. Les autres dérivées secondes $\frac{\partial^2 f}{\partial x \partial y}(x,y)$ et $\frac{\partial^2 f}{\partial y \partial x}(x,y)$ sont appelées dérivées *mixtes* et sont égales, cad $\frac{\partial^2 f}{\partial x \partial y}(x_0, y_0) = \frac{\partial^2 f}{\partial y \partial x}(x_0, y_0)$

En d'autres mots, l'ordre de dérivation est sans conséquence sur le résultat .

5.4 Différentielle totale

Alors que les dérivées partielles d'une fonction multi variables indiquent le taux de variation de la fonction lorsqu'*une* des variables change (toutes les autres variables demeurant constantes), la *différentielle* d'une fonction multi variables indique les variations de la fonction lorsque *toutes* les variables changent simultanément. La notion de différentielle est reliée à la notion de *linéarisation* d'une fonction multi variables et a beaucoup d'importance pratique, dans le calcul d'erreurs par exemple.

5.4.1 Linéarisation d'une fonction multi variables

Supposons que l'on désire trouver une approximation d'une fonction multi variables $f(x_1, x_2, \ldots x_n)$ autour d'un point $(x_1^0, \ldots x_n^0)$. Si le point $(x_1, x_2, \ldots x_n)$ est « suffisamment » proche du point $(x_1^0, \ldots x_n^0)$, alors on obtient :

$$f(x_1, \ldots x_n) \approx f(x_1^0, \ldots x_n^0) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(x_1^0, \ldots x_n^0)(x_i - x_i^0)$$

C'est la linéarisation de $f(x_1, x_2, \ldots x_n)$ autour du point $(x_1^0, \ldots x_n^0)$.

Retenons que :

- La linéarisation de $f(x_1, x_2, \ldots x_n)$ autour du point $(x_1^0, \ldots x_n^0)$ est la fonction

$$L(x_1, \ldots x_n) = f(x_1^0, \ldots x_n^0) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(x_1^0, \ldots x_n^0)(x_i - x_i^0)$$

- L'approximation linéaire standard de $f(x_1, x_2, \ldots x_n)$ autour du point $(x_1^0, \ldots x_n^0)$ consiste à écrire

$$f(x_1, \ldots x_n) \approx f(x_1^0, \ldots x_n^0) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(x_1^0, \ldots x_n^0)(x_i - x_i^0)$$

Exemple :

- Considérons à nouveau la relation thermodynamique des gaz parfaits $p = \frac{RT}{V} = p(V,T)$, où *p* est la pression du gaz, *V* est son volume et *T* sa température (*R*=8.32J/mole.°K). Supposons des conditions $V = 0.15m^3$, $T = 300^0 K$. Cherchons alors à calculer l'approximation linéaire standard de la surface $p = \frac{RT}{V} = p(V,T)$ au point $V = 0.15m^3$, $T = 300^0 K$. Nous avons déjà calculé (section 7.4.2) $\frac{\partial p}{\partial V}(0.15,300) = -110933 Pa/m^3$ et $\frac{\partial p}{\partial T}(0.15,300) = 55.47 Pa/^0 K$. On obtient donc l'approximation linéaire

$$L(V,T) = 16640 - 110933(V - 0.15) + 55.47(T - 300)$$

La surface $L(V,T)$ est représentée sur le graphique ci-dessous, avec la surface $p = \frac{RT}{V} = p(V,T)$. L'approximation linéaire standard $L(V,T)$ est l'équation d'un *plan tangent* à la surface $p = \frac{RT}{V} = p(V,T)$ au point $V = 0.15m^3$, $T = 300^0 K$. Ce plan fournit la meilleure approximation linéaire de la surface $p = \frac{RT}{V} = p(V,T)$ autour du point considéré.

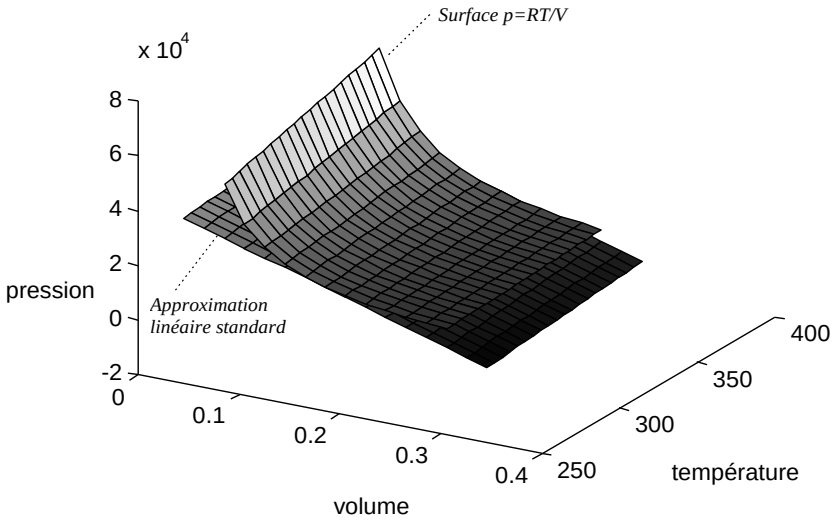


Figure 5.3 : Linéarisation d'une fonction de 2 variables et plan tangent en un point

Retenons que pour une fonction de deux variables $f(x,y)$, l'approximation linéaire standard au point (x_0,y_0), $L(x,y) = f(x_0,y_0) + \frac{\partial f}{\partial x}(x_0,y_0)(x - x_0) + \frac{\partial f}{\partial y}(x_0,y_0)(y - y_0)$ représente l'équation d'un plan tangent à la surface $f(x,y)$ au point (x_0,y_0).

5.4.2 Différentielle totale

La notion de différentielle totale d'une fonction multi-variables découle naturellement de celle de linéarisation, vue précédemment. Soit une fonction de *n* variables $w = f(x_1, x_2, \ldots x_n)$ pour laquelle on désire trouver la variation Δf lorsque les variables indépendantes subissent des accroissements $\Delta x_1, \Delta x_2, \ldots \Delta x_n$ autour du point $(x_1^0, \ldots x_n^0)$. On définit la *différentielle totale* (ou plus simplement la *différentielle*) *df*

$$df = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(x_1^0, \ldots x_n^0) dx_i$$

La différentielle totale permet d'approcher les variations de *f* pour de petites variations dx_i des variables x_i autour du point $(x_1^0, \ldots x_n^0)$.

Dans le cas d'une fonction de 2 variables $f(x, y)$, la différentielle totale est :

$$df = \frac{\partial f}{\partial x}(x_0, y_0) dx + \frac{\partial f}{\partial y}(x_0, y_0) dy$$

Exemples :

- Calculons la différentielle totale de la fonction $f(x, y, z) = x \cos y - e^z$ au point (1,0,0).

On a $\frac{\partial f}{\partial x} = \cos y$, $\frac{\partial f}{\partial y} = -x \sin y$, $\frac{\partial f}{\partial z} = -e^z$, donc $df = \cos y dx - x \sin y dy - e^z dz$. Au point (1,0,0), $df = dx - dz$.

- Soit un réservoir de forme cylindrique, de rayon *r* et de hauteur *h*. Les valeurs nominales de *r* et *h* sont $r_0 = 0.4m$ et $h_0 = 1m$ avec des incertitudes absolues sur le rayon de 0.003*m*, et sur la hauteur de 0.006*m,* c'est-à-dire $r = 0.4m \pm 0.003m$ $h = 1m \pm 0.006m$. On veut connaître *l'incertitude absolue* sur le volume *V* du réservoir.

Les calculs de petites variations, d'erreurs et d'incertitudes sont une application privilégiée de la notion de différentielle totale. Le volume du réservoir est

$$V = \pi r^2 h = V(r, h)$$

Les incertitudes sur *r* et *h* peuvent être considérées comme de petites variations *dr* et *dh* autour des valeurs nominales $r_0 = 0.4m$ et $h_0 = 1m$. Pour déterminer l'incertitude sur *V*, formons la différentielle totale

$$dV = \frac{\partial V}{\partial r}(r_0, h_0)dr + \frac{\partial V}{\partial h}(r_0, h_0)dh = 2\pi r_0 h_0 dr + \pi {r_0}^2 dh$$

L'incertitude absolue sur *V* est la valeur maximale prise par $|dV|$ lorsque $dr = \pm 0.003m$ et $dh = \pm 0.006m$.

$$|dV| = \left|2\pi r_0 h_0 dr + \pi {r_0}^2 dh\right| \le 2\pi r_0 h_0 |dr| + \pi {r_0}^2 |dh|$$

d'où

$$|dV| \le 2.513 \times 0.003 + 0.503 \times 0.006 = 0.011 m^3$$

L'incertitude absolue sur le volume est donc $0.0011m^3$ (pour un volume nominal de $\pi r {r_0}^2 h_0 = 0.503m^3$). Remarquons qu'un calcul direct à partir de l'expression du volume permet de déterminer l'intervalle de variation du volume en fonction des incertitudes sur *r* et *h* :

$$V(r_0, h_0) = \pi {r_0}^2 h_0 = 0.503m^3$$
$$V(r_0 + dr, h_0 + dh) = \pi (r_0 + 0.003)^2 (h_0 + 0.006) = 0.513m^3$$
$$V(r_0 - dr, h_0 - dh) = \pi (r_0 - 0.003)^2 (h_0 - 0.006) = 0.492m^3$$

On obtient donc une incertitude absolue de $0.010m^3$, proche de celle calculée avec la différentielle.

Il est usuel de raisonner également en termes d'incertitudes *relatives*, plutôt qu'*absolues*. Considérons de nouveau la différentielle totale

$$dV = 2\pi r_0 h_0 dr + \pi {r_0}^2 dh$$

On en déduit

$$\frac{dV}{V} = \frac{2\pi r_0 h_0}{\pi {r_0}^2 h_0} dr + \frac{\pi {r_0}^2}{\pi {r_0}^2 h_0} dh = \frac{2dr}{r_0} + \frac{dh}{h_0}$$

L'incertitude relative sur le volume est la valeur maximale prise par $\left|\frac{dV}{V}\right|$. On obtient donc une incertitude relative

$$\left|\frac{dV}{V}\right| \le 2\left|\frac{dr}{r_0}\right| + \left|\frac{dh}{h_0}\right|,$$

où $\left|\frac{dr}{r_0}\right|, \left|\frac{dh}{h_0}\right|$ sont les incertitudes relatives sur le rayon et sur la hauteur respectivement. En valeur numérique, l'incertitude relative sur le volume est donc $2\frac{0.003}{0.4} + \frac{0.006}{1} = 0.021 = 2.1\%$.

- Considérons à nouveau la relation thermodynamique des gaz parfaits $p = \frac{RT}{V} = p(V,T)$, où *p* est la pression du gaz, *V* est son volume et *T* sa température (*R*=8.32J/mole.ºK). Supposons des conditions $V_0 = 0.15m^3$, $T_0 = 300^0 K$. Cherchons alors à calculer la variation de pression engendrée par des variations de température $dT = 1^0 K$ (accroissement) et de volume $dV = -0.001m^3$ (diminution).

Formons la différentielle : $dp = \frac{\partial p}{\partial V}(V_0, T_0)dV + \frac{\partial p}{\partial T}(V_0, T_0)dT$

Nous avons déjà calculé (section 7.4.2) $\frac{\partial p}{\partial V}(0.15,300) = -110933 Pa/m^3$ et $\frac{\partial p}{\partial T}(0.15,300) = 55.47 Pa/^0 K$. On obtient donc

$$dp = -110933 \times (-0.001) + 55.47 \times 1 = 110.9 + 55.47 = 166.4 Pa$$

Notons que l'accroissement de température et la diminution de volume concourent tous deux à un accroissement de la pression. Par ailleurs, remarquons encore une fois qu'un calcul direct à partir de l'expression initiale $p = \frac{RT}{V} = p(V,T)$ fournirait la réponse exacte,

$$p(V_0, T_0) = 16640 Pa$$
$$p(V_0 + dV, T_0 + dT) = 16808 Pa$$

d'où un accroissement de pression de 167.5Pa, ce qui est très proche de la valeur trouvée à l'aide de la différentielle.

5.5 Synthèse

Définition d'une fonction réelle de plusieurs variables réelles :

$$f \quad : D \subset R^n \to R$$
$$(x_1, x_2, \ldots x_n) \mapsto w = f(x_1, x_2, \ldots x_n)$$

On appelle $x_1, x_2, \ldots x_n$ les *n* variables réelle **indépendantes**, *w* est la variable réelle **dépendante.**

Dérivées partielles :

On définit la dérivée partielle de *f* par rapport à la variable indépendante $x_i \ (i = 1,2,\ldots n)$, au point $\left(x_1^0, x_2^0, \ldots x_n^0\right)$ comme :

$\lim_{(x_1, x_2, \ldots x_n) \to (x_1^0, x_2^0, \ldots x_n^0)} \frac{f(x_1, x_2, \ldots x_i + h, \ldots x_n) - f(x_1, x_2, \ldots x_i, \ldots x_n)}{h} = \frac{\partial f}{\partial x_i}(x_1^0, x_2^0, \ldots x_n^0)$ lorsque cette limite existe.

Théorème sur l'égalité des dérivées mixtes :

Si *f*, de même que ses dérivées partielles $\frac{\partial f}{\partial x}$, $\frac{\partial f}{\partial y}$, $\frac{\partial^2 f}{\partial x \partial y}$ et $\frac{\partial^2 f}{\partial x \partial y}$ sont toutes définies dans un voisinage du point (x_0, y_0) et sont continues au point (x_0, y_0), alors $\frac{\partial^2 f}{\partial x \partial y}(x_0, y_0) = \frac{\partial^2 f}{\partial y \partial x}(x_0, y_0)$

(se généralise à des fonctions de *n* variables)

Différentielle :

$$df = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(x_1^0, \ldots x_n^0) dx_i$$

Linéarisation d'une fonction multi-variable :

L'approximation linéaire standard de $f(x_1, x_2, \ldots x_n)$ autour du point $(x_1^0, \ldots x_n^0)$ consiste à écrire

$$f(x_1, \ldots x_n) \approx f(x_1^0, \ldots x_n^0) + \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(x_1^0, \ldots x_n^0)(x_i - x_i^0)$$

Chapitre 6 : Résolution numérique d'équations non-linéaires

6.1 Introduction et formulation du problème

Le but de ce chapitre est d'aborder la résolution numérique d'équations ou de systèmes d'équations du genre :

$$f(x) = 0$$

où la fonction $f(x)$ peut être une fonction à une variable réelle, mais également une fonction multi-variable ou encore une fonction à plusieurs dimensions (système d'équations).

Ce problème est très fortement lié à des problèmes connexes très communs en physique et en dynamique des structures tel que :

- Trouver la valeur particulière d'une fonction $g(x) = c$ est équivalent à trouver les zéros de :
$$f(x) = g(x) - c$$

- Minimiser une fonction $F(x)$ est équivalent à trouver les zéros de sa dérivée :
$$f(x) = F'(x)$$

- Trouver un point fixe x tel que $g(x) = x$ est équivalent à trouver les zéros de
$$f(x) = g(x) - x$$

On sait résoudre ce type de problèmes analytiquement (à la main) lorsque la fonction f est un polynôme d'ordre inférieur ou égal à 4. Cependant, dès que la situation devient plus compliquée, la résolution numérique devient indispensable. En particulier quand :

- la fonction $f(x)$ est fortement non-linéaire
- il n'existe pas de solution analytique explicite (par exemple $x = e^x$ ou $x = \cos x$)
- le problème implique un système linéaire d'ordre élevé : $M\vec{X} = \vec{Y}$ avec M une matrice
- le problème implique un système non-linéaire : $F(X) = 0$

Ce chapitre est principalement dédié à la résolution d'équations de faible dimension.

Les méthodes numériques de résolution d'équations sont des méthodes itératives. Elles reposent sur le calcul d'une récurrence qui part d'un point x_0 (a priori différent de la racine que l'on cherche), et qui, de proche en proche, tend vers la racine x_r. Elles permettent donc d'obtenir une valeur approchée de la racine.

Elles reposent donc sur 3 choix importants :

- **Le point de départ x_0.** En général, plus on part proche de la racine cherchée, plus les méthodes sont efficaces. Dans certains cas, une méthode peut ne pas converger du tout si le point de départ est inadapté.

- **La méthode de récurrence $x_{n+1} = f(x_n)$**. Nous verrons plusieurs méthodes différentes.

- **Le critère d'arrêt.** La récurrence tend en général asymptotiquement (au bout d'une infinité d'itérations) vers la racine. Il faut donc se définir un critère qui arrête la récurrence lorsque l'on est suffisamment proche de la solution. On utilise souvent l'un ou plusieurs des 4 critères suivants :
    - $n > n_{max}$ **:** un nombre maximal d'itérations à ne pas dépasser quoi qu'il arrive
    - $|x_{n+1} - x_n| < \epsilon_1$ **:** lorsque deux itérations successives donnent des résultats très proches, on peut supposer que l'on a convergé vers une solution. Cependant, dans le cas d'une fonction complexe, on peut parfois, par malchance, avoir deux itérations proches tout en étant loin de la racine. De même un tel critère appliqué à une fonction très raide ne garantira pas que f = 0 à un grand niveau de précision.
    - $|f(x_n)| < \epsilon_2$: lorsque x est proche de la racine, $f(x)$ doit être proche de 0. Cependant, un tel critère appliqué à une fonction très plate (dont les dérivées sont aussi nulles) ne fournira pas forcément une approximation précise de la racine.

Selon ces choix, les méthodes peuvent converger ou non, et lorsqu'elles convergent, elles peuvent le faire plus ou moins vite. On estime la performance d'une méthode en comparant le nombre d'itérations nécessaires pour arriver suffisamment proche de la solution et le temps nécessaire pour exécuter une unique itération. Pour caractériser les différentes méthodes, on définit souvent l'erreur ϵ_n à l'itération n comme l'écart entre la valeur x_n de l'algorithme et à la racine théorique x_r :

$$\epsilon_n = x_n - x_r$$

Quelle que soit la méthode choisie, de nombreux peuvent problèmes gêner la convergence, comme :

- la présence de plusieurs solutions (on parle de racines multiples)
- la présence de pôles (pour lesquels la fonction $f(x)$ diverge)
- la présence de racines de multiplicité élevée (cad que les dérivées sont nulles)
- des asymptotes finies en $\pm\infty$

6.2 Résolution numérique d'équations non-linéaires

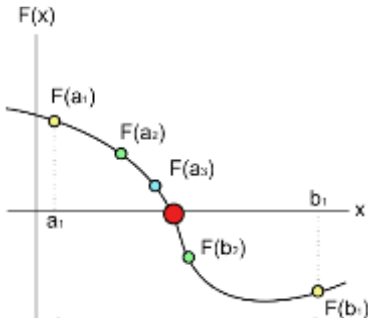
Il s'agit dans cette section de résoudre une équation du type $f(x) = 0$ où $f(x)$ et x sont des scalaires.

6.2.1 Méthode de dichotomie (ou bissection)

La méthode de dichotomie requiert deux valeurs de départ a_1 et b_1 qui encadrent la racine x_r à trouver. Cela signifie donc que $f(a_1)$ et $f(b_1)$ sont de signes opposés. On fait ensuite une double récurrence sur a_n et b_n en réduisant à chaque itération la taille de l'intervalle par 2.

Pour cela, on regarde le signe de la fonction au point milieu $c = \frac{a_n + b_n}{2}$ et on détermine un nouvel encadrement de la solution en utilisant ce point :

- Si $f(a_n)\, f(c) > 0$ alors $\begin{cases} a_{n+1} = c \\ b_{n+1} = b_n \end{cases}$
- Sinon $\begin{cases} a_{n+1} = a_n \\ b_{n+1} = c \end{cases}$



Cette méthode est très simple à mettre en œuvre. De plus elle est extrêmement robuste une fois que l'on a trouvé a_1 et b_1. Cependant, elle nécessite une première recherche de a_1 et b_1 sans lesquelles elle ne peut être appliquée. De plus, sa convergence est très lente. Enfin, cette méthode est inadaptée pour les zéros de multiplicité élevée (si les dérivées sont aussi nulles).

6.2.1 Méthode de Newton Raphson

La méthode de Newton-Raphson (dénotée NR et parfois appelée simplement méthode de Newton) utilise une approximation de Taylor pour trouver la racine de l'équation.

Selon la série de Taylor, évaluée autour d'un point initial x_0, la fonction $f(x)$ évaluée au point x est :

$$f(x) = f(x_0) + \frac{f'(x_0)}{1!}(x - x_0) + + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots$$

Autour de la racine (du zéro) de la fonction, c'est à dire pour $x = x_r$ cette expression devient alors :

$$f(x_r) = f(x_0) + \frac{f'(x_0)}{1!}(x_r - x_0) + + \frac{f''(x_a)}{2!}(x_r - x_0)^2 + \cdots = 0$$

Le polynôme étant une série infinie sans solution analytique, la méthode de Newton-Raphson utilise les deux premiers termes de la série de Taylor:

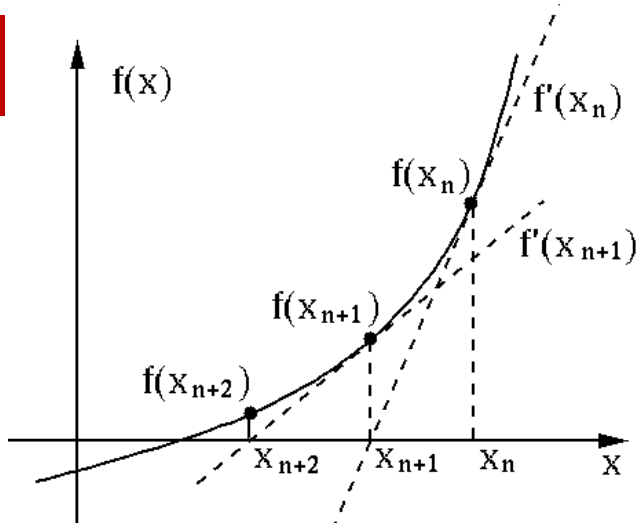
$$f(x_a) + f'(x_0)(x_r - x_0) = 0 \qquad \text{cad} \qquad x_r = x_0 - \frac{f(x_0)}{f'(x_0)}$$

À part le cas où la fonction originale $f(x)$ était une droite (auquel cas, la méthode de Newton-Raphson aurait été efficace mais inutile pour en trouver la racine), la solution obtenue avec cette équation ne sera pas exactement la racine, à cause des termes d'ordre supérieur à 2 qui ont été négligés dans la série de Taylor. Il faut donc itérer pour converger vers la solution :

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Comme pour la méthode de la bissection, les itérations sont arrêtées quand la précision requise est suffisante ou selon le critère de convergence désiré.

En général, la méthode de Newton-Raphson prend moins d'itérations que la méthode de bissection pour obtenir la même précision dans le calcul de la racine. Cependant, elle n'est pas sans faiblesse pour certains types de fonctions.



Par exemple, dans le cas d'une fonction avec plusieurs racines et donc avec plusieurs solutions possibles, la solution obtenue dépend du point de départ x_0 des itérations et son choix près de la racine ne signifie pas nécessairement que les itérations vont se terminer à cette racine !

6.2.1 Méthode de la sécante

Dans certains problèmes physiques ou numériques on peut ne pas connaître l'expression de la dérivée. Dans ce cas, on peut estimer numériquement le terme de dérivée dans l'équation de Newton-Raphson par une méthode de dérivation arrière :

$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

Dans ce cas, la relation de récurrence de Newton-Raphson devient :

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n)$$

Vu que la dérivée doit être évaluée avec un schéma de dérivation arrière, il faut initialiser les itérations avec deux valeurs de la variable indépendante x, dénotées par x_0 et x_1 . En pratique, on ne peut choisir que x_0 et fixer x_1 en fonction de x_0 (par exemple, $x_1 = x_0 + 10^{-4}$).

6.3 Résolution de systèmes d'équations non-linéaires

Il s'agit dans cette section de résoudre un système d'équations du type :

$$\begin{gathered} f_1(x_1, x_2, \dots, x_m) = 0 \\ f_2(x_1, x_2, \dots, x_m) = 0 \\ \dots \\ f_m(x_1, x_2, \dots, x_m) = 0 \end{gathered}$$

Où les fonctions $f_1(x) \dots f_m(x)$ sont m fonctions multi-variables à m inconnues notées $x_1, x_2, \dots, x_m$. Ce problème est très souvent rencontré en statique ou en dynamique des structures et doit dans la plupart des cas être résolu en temps réel, d'où la nécessité de déployer des solutions numériques rapides et efficaces qui pourront éventuellement être embarquées sur des plateformes mobiles.

Dans ce cas, la technique de la bissection n'est plus applicable et nous devons donc étendre la technique de Newton-Raphson au cas des systèmes d'équations non-linéaires. La formulation est identique au cas d'une équation à une variable (cf section précédente) sauf que nous introduisons le problème sous forme vectorielle :

$$X = [x_1, x_2, \dots, x_m]$$

$$F(X) = [f_1(X), f_2(X), \dots, f_m(X)]$$

Où les vecteurs X et $F(X)$ sont deux vecteurs colonnes de longueur m. L'idée est de développer un algorithme itératif qui à chaque itération n, permet d'obtenir une solution pour le vecteur X que l'on nomme X_n. La formule de Newton-Raphson dans le cas d'un problème à m dimensions devient alors pour l'itération m:

$$X_{n+1} = X_n - J_F^{-1}(X_n)\, F(X_n)$$

Où $J_F^{-1}(X)$ désigne la matrice inverse de la matrice $J_F(X)$ qui se nomme le **Jacobien** du système $F(X)$ que l'on exprime par :

$$J_F(X) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(X) & \cdots & \frac{\partial f_1}{\partial x_m}(X) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1}(X) & \cdots & \frac{\partial f_m}{\partial x_m}(X) \end{bmatrix}$$

Cette formule est une généralisation du cas à 1 dimension et sa justification (non présentée par soucis de simplification) fait appel à la notion de différentielle totale (cf Chapitre 5).

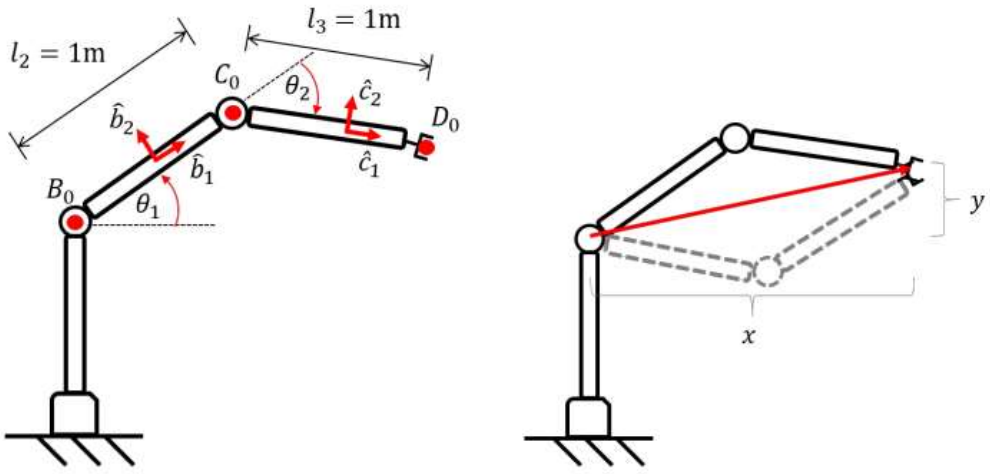
En pratique, il est souvent impossible ou difficile d'estimer de manière analytique les dérivées partielles de F. Dans ce cas, une méthode similaire à la méthode de la sécante peut être dérivée en remplaçant les expressions des dérivées partielles par :

$$\frac{\partial f_i}{\partial x_j}(X_n) \approx \frac{f_i(X_n) - f_i(X_{n-1})}{x_j^n - x_j^{n-1}}$$

Où x_j^n représente la composante j du vecteur X_nà l'itération n. Cette approche nécessite donc deux points initiaux qui peuvent être proches et choisis ide manière dépendante comme pour la méthode de la sécante.

Application : Bras robotisé et cinématique inverse

Considérons l'exemple ci-dessous d'un bras manipulateur à 2 degrés de liberté (θ_1, θ_2) que l'on aiemrait piloter afin d'arriver au point (x, y) de coordonnées connues.



La modélisation directe de la cinématique nous donne le système d'équations suivant (cf GRO 203) :

$$x = \cos\theta_1 + \cos(\theta_1 + \theta_2)$$

$$y = \sin\theta_1 + \sin(\theta_1 + \theta_2)$$

On transforme donc ce problème en un système à deux équations multivariables $f_1(\theta_1, \theta_2)$ et $f_2(\theta_1, \theta_2)$ et à deux inconnues (θ_1, θ_2).

$$f_1(\theta_1, \theta_2) = x - \cos\theta_1 - \cos(\theta_1 + \theta_2) = 0$$

$$f_2(\theta_1, \theta_2) = y - \sin\theta_1 - \sin(\theta_1 + \theta_2) = 0$$

Le Jacobien de ce système s'écrit ainsi :

$$J_F(\theta_1,\theta_2) = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \frac{\partial f_1}{\partial \theta_2} \\ \frac{\partial f_2}{\partial \theta_1} & \frac{\partial f_2}{\partial \theta_2} \end{bmatrix} = \begin{bmatrix} \sin\theta_1 + \sin(\theta_1+\theta_2) & \sin(\theta_1+\theta_2) \\ -\cos\theta_1 - \cos(\theta_1+\theta_2) & -\cos(\theta_1+\theta_2) \end{bmatrix}$$

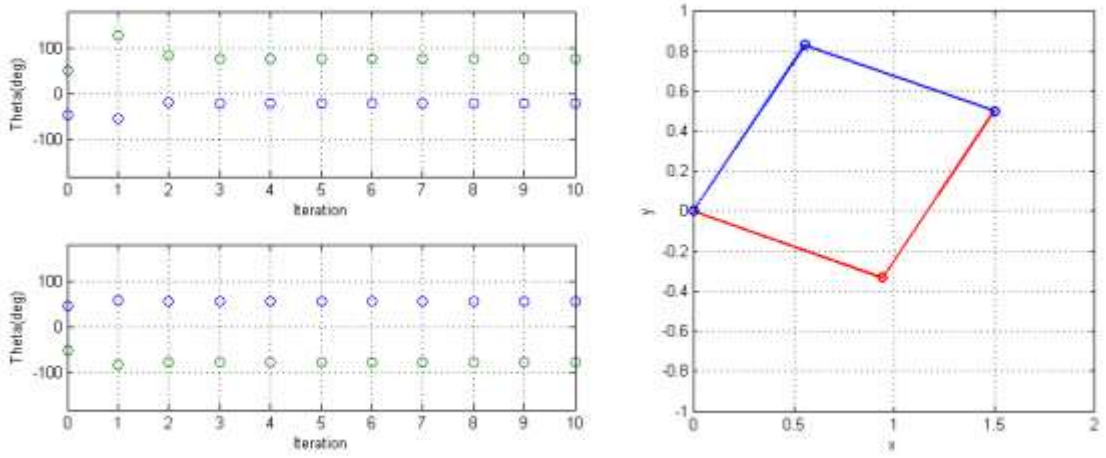
$$J_F(\theta_1,\theta_2)^{-1} = \begin{bmatrix} -\frac{\cos(\theta_1+\theta_2)}{\sin\theta_2} & -\frac{\sin(\theta_1+\theta_2)}{\sin\theta_2} \\ \frac{\cos\theta_1+\cos(\theta_1+\theta_2)}{\sin\theta_2} & \frac{\sin\theta_1+\sin(\theta_1+\theta_2)}{\sin\theta_2} \end{bmatrix}$$

Ce qui permet d'obtenir la formule de récurrence pour déterminer la suite d'angles (θ_1^n, θ_2^n) à l'itération n qui permettent d'atteindre la position (x, y) après convergence.

$$\begin{bmatrix} \theta_1^{n+1} \\ \theta_2^{n+1} \end{bmatrix} = \begin{bmatrix} \theta_1^n \\ \theta_2^n \end{bmatrix} - J_F^{-1}(\theta_1^n, \theta_2^n) \begin{bmatrix} x - \cos\theta_1^n - \cos(\theta_1^n+\theta_2^n) \\ y - \sin\theta_1^n - \sin(\theta_1^n+\theta_2^n) \end{bmatrix}$$

En pratique, cette résolution est effectuée en temps réel (plusieurs dizaines de fois par seconde) à l'aide d'un processeur intégré afin d'obtenir la convergence vers la consigne optimale $(\theta_1^\infty, \theta_2^\infty)$.

Par exemple, pour $y = 0.5$ et $x = 1.5$, deux solutions existent au problème de cinématique inverse (cf. Figure ci-dessus) . Dans ce cas, le choix de la valeur initiale pour les variables (θ_1^0, θ_2^0) va déterminer vers quelle solution l'équation de récurrence va converger. Par exemple, si on choisit : $(\theta_1^0, \theta_2^0) = (45°, 50°)$, on converge vers la solution $(\theta_1^\infty, \theta_2^\infty) = (-19°, 75°)$, alors que si on choisit : $(\theta_1^0, \theta_2^0) = (-45°, 50°)$, on converge vers la solution $(\theta_1^\infty, \theta_2^\infty) = (56°, -75°)$.



Il est important de noter que dans ce cas, l'algorithme est capable de converger après seulement 4 itérations, démontrant la rapidité de cette méthode dans des cas pratiques. L'implantation sous Matlab peut ainsi se faire de deux manières distinctes :

Cas #1 :

Dans le cas où les fonctions $f_1(\theta_1, \theta_2)$ et $f_2(\theta_1, \theta_2)$ sont analytiques, on peut alors les dériver analytiquement afin de calculer le Jacobien de la fonction $F(\theta_1, \theta_2) = [f_1(\theta_1, \theta_2)\ ;\ f_2(\theta_1, \theta_2)]$. Dans ce cas l'inversion peut également s'effectuer analytiquement et dans ce cas, le code Matlab est :

```
% INITIALISATION
x = 1.5;
y = 0.5;
theta_1_init = -45;
theta_2_init = 50;
THETA=[theta_1_init;theta_2_init]*pi/180;

for ii=1:10

%     CALCUL DE LA FONCTION F(THETA_1,THETA_2)
    F = [x-cos(THETA(1))-cos(THETA(1)+THETA(2));
         y-sin(THETA(1))-sin(THETA(1)+THETA(2))];

%     CALCUL DE L'INVERSE DU JACOBIEN
    invJF = -1./sin(THETA(2))*[cos(THETA(1)+THETA(2)),sin(THETA(1)+THETA(2));
    -cos(THETA(1))-cos(THETA(1)+THETA(2)) , -sin(THETA(1))-sin(THETA(1)+THETA(2))];

    THETA = THETA - invJF*F;

end
```

Cas #2 :

Dans le cas où les fonctions $f_1(\theta_1, \theta_2)$ et $f_2(\theta_1, \theta_2)$ ne sont pas analytiques, on ne peut pas les dériver analytiquement et le Jacobien de la fonction $F(\theta_1, \theta_2) = [f_1(\theta_1, \theta_2)\ ;\ f_2(\theta_1, \theta_2)]$ doit être calculé de manière numérique avant d'être inversé (la phase d'initialisation est omise par soucis de clarté). On doit ainsi introduire un nouveau paramètre `D_THETA` de pas de différentiation. Notez également l'utilisation d'un fonction alias (`help @` pour plus de renseignements) permettant de définir la fonction $F(\theta_1, \theta_2)$ comme une fonction multi-variables à deux variables `THETA(1)` et `THETA(2)` et deux dimensions.

```
F = @(THETA) [x-cos(THETA(1))-cos(THETA(1)+THETA(2)) ;
              y-sin(THETA(1))-sin(THETA(1)+THETA(2)) ];

D_THETA = 1e-8;

for ii=1:10

     % Calcul du Jacobien de manière numérique (centrée)
     DF_THETA1 = ( F(THETA+[DT;0]) - F(THETA-[DT;0]) ) / (2* D_THETA);
     DF_THETA2 = ( F(THETA+[0;DT]) - F(THETA-[0;DT]) ) / (2* D_THETA);
     JF = [DF_THETA1 , DF_THETA2];

     THETA = THETA - inv(JF)*F(THETA);

end
```

6.4 Synthèse

Méthode de dichotomie (ou bissection)

On regarde le signe de la fonction au point milieu $c = \frac{a_n + b_n}{2}$

Si $f(a_n)\, f(c) > 0$ alors $\begin{cases} a_{n+1} = c \\ b_{n+1} = b_n \end{cases}$ Sinon $\begin{cases} a_{n+1} = a_n \\ b_{n+1} = c \end{cases}$

Méthode de la sécante

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n)$$

Méthode de Newton-Raphson

Résolution d'un équation à 1 inconnue :

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Résolution d'un système de m quations à m inconnues :

$$X_{n+1} = X_n - J_F^{-1}(X_n)\, F(X_n)$$

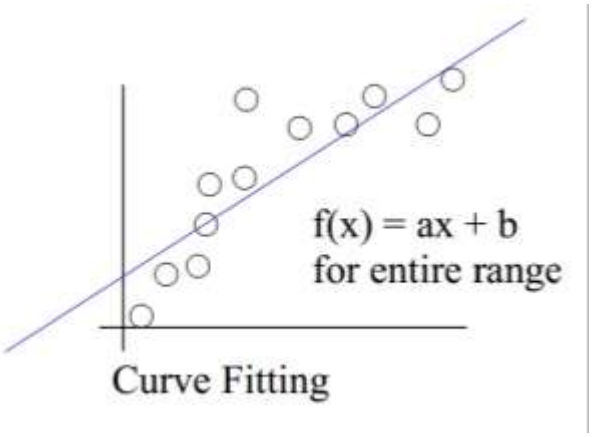
$$J_F(X) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(X) & \cdots & \frac{\partial f_1}{\partial x_m}(X) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1}(X) & \cdots & \frac{\partial f_m}{\partial x_m}(X) \end{bmatrix}$$

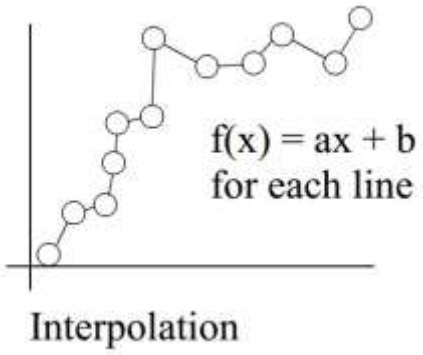
Chapitre 7 : Approximation discrète de données

7.1 Introduction

Dans ce chapitre, nous nous intéressons au problème d'approximation de données (*curve fitting* en anglais) qui consiste à développer et résoudre les équations algébriques permettant de représenter des données expérimentales ou une relation fonctionnelle complexe par une fonction continue simple et facile à évaluer.

Contrairement à l'interpolation de données (non traitée dans ce cours), on ne tente pas ici de représenter exactement les données puisque, typiquement, elles contiennent du bruit et un lissage des données est requis.





L'approximation de données permet de trouver la tendance de données expérimentales tout en excluant le bruit de mesure par effet de lissage de points de mesures. C'est donc un outil très utilisé pour l'analyse de données expérimentales.

Elle permet également une représentation simplifiée d'une fonction difficile à évaluer (par exemple une simulation numérique ou une expérience réalisée en un nombre faible de points) , ce qui permet un calcul plus rapide de la fonction ou d'en déduire plus facilement certaines propriétés (e.g. points minimal/maximal).

Dans le cadre du cours GRO305, nous nous intéressons à l'approximation de données discrètes, cad que nous considérons que la fonction que l'on cherche à « fitter » aux données mesurées peut être paramétrisée par un nombre fini de paramètres (par exemple, pour une approximation linéaire, 2 paramètres sont requis – la pente et l'ordonnée à l'origine)

7.2 Concepts de base

On considère un ensemble de n paires de données (x_1, y_1) … , (x_n, y_n) que l'on aimerait approximer par une fonction $g(x)$. Le concept de base derrière l'approximation de données est de s'arranger pour que le tracé de la fonction $g(x)$ se rapproche de ces points (sans nécessairement passer par ces points) tout en minimisant un critère d'erreur. Plusieurs choix existent pour la fonction $g(x)$; il faut choisir le meilleur candidat !

Il faut donc transformer ce problème géométrique (mesure de distance entre la courbe et les données) en un problème algébrique et le solutionner. Nous introduisons donc un critère d'erreur que l'on nomme le critère du moindre carré, qui est défini de la sorte :

- Pour chaque point p associé à une coordonnée (x_p, y_p), on définit **l'erreur d'approximation** δ_p entre la fonction $g(x)$ et le point p par :

$$\delta_p = g(x_p) - y_p$$

- La somme sur tous les points des erreurs d'approximation au carré est appelée **l'erreur quadratique** E et est définie par:

$$E = \sum_{p=1}^{n} \delta_p^2 = \sum_{p=1}^{n} \left(g(x_p) - y_p\right)^2$$

L'erreur quadratique est un des meilleurs indicateurs de la qualité d'une approximation. Aussi, une petite valeur de E implique que la fonction $g(x)$ représente "bien" les données (x_1, y_1) … , (x_n, y_n).

L'approche pour la sélection de la fonction d'approximation $g(x)$ comporte 5 étapes distinctes :

- Identifier une fonction candidate $g(x)$, avec M coefficients libres $A = [a_1 \dots a_M]$ à déterminer. Par exemple dans le cas d'une régression linéaire, on a : $g(x) = a_1 x + a_2$ donc $M = 2$ coefficients libres.

- Écrire l'équation algébrique de l'erreur quadratique (somme des erreurs au carré) qui dépend du choix des M coefficients $A = [a_1 \dots a_M]$:

$$E(A) = \sum_{p=1}^{n} \left(g(x_p) - y_p\right)^2$$

- Calculer les conditions pour avoir un minimum de E . Sachant que la fonction $E(A)$ est une fonction multi-variables à M variables indépendants, il est nécessaire d'en rechercher les points critiques, c'est-à-dire de déterminer les points pour lesquels toutes les dérivées partielles sont nulles :

$$\frac{\partial E}{\partial a_1} = 0 \qquad \dots \qquad \frac{\partial E}{\partial a_M} = 0$$

Ces équations sont appelées les **équations normales** et seront utilisées afin de définir les coefficients optimaux pour l'approximation de données.

- Résoudre ce équations normales pour les M coefficients libres $A = [a_1 \dots a_M]$ de la fonction $g(x)$. Cette phase requiert la plupart du temps (sauf dans les cas simples d'approximation polynomiale) une résolution numérique d'un système d'équations non-linéaires (cf chapitre 6)

- Reconstruire l'approximation et vérifier la qualité de l'approximation. Au besoin, si le résultat n'est pas satisfaisant, on devra recommencer avec une autre fonction candidate

7.3 Approximation polynomiale de données

7.3.1 Approximation linéaire de données (1 coefficient)

Considérons dans un premier temps l'approximation de données à l'aide d'une droite passant par l'origine. Cette droite est paramétrée par $M = 1$ coefficient que l'on note a_1 et on définit la courbe par:

$$g(x) = a_1 x$$

En suivant la démarche décrite plus haut, on écrit dans le cas où l'on a N points (x_k, y_k) à approximer, l'erreur quadratique $E(a_1)$:

$$E(a_1) = \sum_{k=1}^{N} [g(x_k) - y_k]^2 = \sum_{k=1}^{N} [a_1 x_k - y_k]^2$$

On détermine ensuite l'équation normale pour le problème en recherchant le minimum de $E(a_1)$ par rapport au coefficient a_1 :

$$\frac{dE}{da_1} = 0 \qquad \Rightarrow \qquad \sum_{k=1}^{N} 2[a_1 x_k - y_k]\, x_k = 0$$

On veut trouver la solution pour a_1. Dans ce cas, on isole ce paramètre en le sortant de la sommation :

$$2a_1 \sum_{k=1}^{N} x_k^2 - 2 \sum_{k=1}^{N} y_k x_k = 0$$

Afin d'obtenir la solution finale :

$$a_1 = \frac{\sum_{k=1}^{N} y_k x_k}{\sum_{k=1}^{N} x_k^2}.$$

Ensuite, en remplaçant cette valeur de a_1 dans la fonction $g(x) = a_1 x$, on obtient la meilleure ligne droite qui passe par les points (x_k, y_k) $k = 1 \dots N$.

7.3.1 Approximation linéaire de données (2 coefficients)

Répétons maintenant la même procédure dans le cas où l'on recherche la meilleure approximation linéaire ne passant pas par l'origine. Cette droite est paramétrée par $M = 2$ coefficients que l'on note $A = [a_0, a_1]$ et on définit la courbe par:

$$g(x) = a_1 x + a_0$$

En suivant la démarche décrite plus haut, on écrit dans le cas où l'on a N points (x_k, y_k) à approximer, l'erreur quadratique $E(a_0, a_1)$:

$$E(a_{0.}, a_1) = \sum_{k=1}^{N} [g(x_k) - y_k]^2 = \sum_{k=1}^{N} [a_1 x_k + a_0 - y_k]^2$$

On détermine ensuite l'équation normale pour le problème en recherchant le minimum de $E(a_0, a_2)$ par rapport au coefficients a_1 et a_0:

$$\frac{dE}{da_1} = 0 \qquad \Rightarrow \qquad \sum_{k=1}^{N} 2[a_1 x_k + a_0 - y_k]\, x_k = 0$$

$$\frac{dE}{da_0} = 0 \qquad \Rightarrow \qquad \sum_{k=1}^{N} 2[a_1 x_k + a_0 - y_k] = 0$$

On veut trouver la solution pour $a_0\ et\ a_1$. Dans ce cas, on isole ces 2 paramètres en les sortant de la sommation :

$$a_1 \sum_{k=1}^{N} x_k^2 + a_0 \sum_{k=1}^{N} x_k = \sum_{k=1}^{N} y_k x_k \qquad\qquad a_1 \sum_{k=1}^{N} x_k + a_0 N = \sum_{k=1}^{N} y_k$$

Afin d'obtenir un problème linéaire que l'on peut écrire sous une forme d'équation matricielle :

$$\begin{bmatrix} \sum_{k=1}^{N} x_k^2 & \sum_{k=1}^{N} x_k \\ \sum_{k=1}^{N} x_k & N \end{bmatrix} \begin{bmatrix} a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{N} y_k x_k \\ \sum_{k=1}^{N} y_k \end{bmatrix}$$

La solution est donc obtenue par inversion du système matriciel:

$$\begin{bmatrix} a_1 \\ a_0 \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{N} x_k^2 & \sum_{k=1}^{N} x_k \\ \sum_{k=1}^{N} x_k & N \end{bmatrix}^{-1} \begin{bmatrix} \sum_{k=1}^{N} y_k x_k \\ \sum_{k=1}^{N} y_k \end{bmatrix}$$

Enfin, en utilisant ces valeurs de a_1 et a_2 dans l'équation d'approximation $g(x) = a_1 x_1 + a_2$, on obtient la meilleure approximation au sens du moindre carré des points $(x_k, y_k)\ k = 1 \dots N$.

7.3.1 Approximation polynomiale de données

Dans le cas où l'on approxime la série de points $(x_k, y_k)\ k = 1 \dots N$ par un polynôme d'ordre M, la fonction d'approximation $g(x)$ peut s'écrire :

$$g(x) = a_M x^M + a_{M-1} x^{M-1} \dots + a_1 x_1 + a_0$$

On peut alors montrer (pas si simple dans le cas général !) que les coefficients $A = [a_0, a_1, \dots, a_M]$ sont solution du problème matriciel :

$$\begin{bmatrix} a_M \\ a_0 \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{N} x_k^{2M} & \cdots & \sum_{k=1}^{N} x_k^M \\ \vdots & \ddots & \vdots \\ \sum_{k=1}^{N} x_k^M & \cdots & N \end{bmatrix}^{-1} \begin{bmatrix} \sum_{k=1}^{N} y_k x_k^M \\ \\ \sum_{k=1}^{N} y_k \end{bmatrix}$$

7.4 Approximation de données avec fonction à 2 paramètres

L'approximation linéaire générale $g(x) = a_1 x_1 + a_0$ est facile et rapide d'utilisation. C'est pourquoi elle est souvent généralisée à d'autres fonctions candidates non-linéaires ayant deux paramètres. Cette généralisation consiste à transformer une fonction non linéaire à deux paramètres en une fonction d'approximation linéaire générale de la forme $g(x) = a_1 x_1 + a_0$. Une fois l'approximation linéaire obtenue, on applique la transformation inverse pour régénérer la fonction non linéaire originale.

Dans une approximation de type « boîte noire », on recherche une fonction qui semble bien représenter la tendance des données à approximer et on ajuste ses paramètres pour minimiser l'erreur entre les données et la fonction. Pour ce faire, les fonctions non linéaires *g(x)* à deux paramètres les plus utilisées sont :

- exponentielle : $g(x) = \alpha\, e^{\beta x}$
- puissance : $g(x) = \alpha\, x^{\beta}$
- logarithmique : $g(x) = \alpha + \beta \ln x$
- réciproque : $g(x) = \alpha + \dfrac{\beta}{x}$
- hyperbolique : $g(x) = \dfrac{\alpha}{\beta + x}$
- hyperbolique-linéaire : $g(x) = \dfrac{\alpha x}{\beta + x}$

Le choix de la fonction candidate non linéaire dépend de la tendance des données à approximer. On choisit la fonction qui semble mieux représenter la tendance des mesures obtenues. Dans une approche « boîte grise », on peut se fier aussi au principe physique à la base des mesures, si ce principe est connu. Par exemple, il peut être démontré par l'équilibre hydrostatique que la densité de l'atmosphère a une variation de forme généralement exponentielle en fonction de l'altitude. Ainsi, l'utilisation de la forme $g(x) = \alpha\, e^{\beta x}$ représenterait bien des mesures de densité en fonction de l'altitude.

Le problème d'approximation par moindre carré d'une fonction candidate non-linéaire *g(x)* avec deux paramètres α et β est :

Minimiser $E = \sum_{n=1}^{N} [g(x_n) - y_n]^2$ par la solution des *équations normales* : $\dfrac{\partial E}{\partial \alpha} = 0,\ \dfrac{\partial E}{\partial \beta} = 0$.

Si on appliquait directement les équations normales $\frac{\partial E}{\partial \alpha} = 0,\ \frac{\partial E}{\partial \beta} = 0$ aux fonctions candidates non linéaires ci-dessus, les équations qui en résulteraient seraient non linéaires et souvent impossibles à résoudre analytiquement. On peut éviter ce problème par une transformation.

En effet, il est possible de transformer ces équations non linéaires $g(x)$ sous une forme linéaire générale $G(X) = a_1 X + a_0$ et utiliser les solutions aux équations normales du cas linéaire pour calculer la solution pour α et β.

Ici, les lettres majuscules X, G pour les variables et les lettres romaines a_1, a_0 pour les paramètres sont utilisés dans le domaine des variables transformées dans le domaine linéaire alors que les lettres minuscules x, g pour les variables et les lettres grecques α, β pour les paramètres sont utilisés dans le domaine des variables originales du domaine non linéaire.

Pour faire la transformation, on utilise par exemple la fonction logarithmique qui permet de transformer les multiplications en additions, les divisions en soustractions et les puissances en multiplications et divisions.

Exemple :
Supposons que des données de mesures bruitées ont une tendance exponentielle. On veut donc une approximation de la forme : $g(x) = \alpha e^{\beta x}$. En prenant le logarithme naturel de chaque côté, on obtient : $\ln g = \ln \alpha + \beta x$ qui est exactement de forme linéaire générale $G(X) = a_1 X + a_0$. En faisant la comparaison, on voit que la correspondance est :

$$G(X) = \ln g(x), \quad a_1 = \beta, \quad X = x, \quad a_0 = \ln \alpha.$$

On applique donc la méthode analytique décrite plus haut avec les données transformées $G_k = \ln g_k$, $X_k = x_k$ pour trouver a_1 et a_0 et on applique les transformations inverses pour trouver α, β :

$$\beta = a_1, \quad \alpha = e^{a_0}.$$

7.5 Qualité de l'approximation

L'erreur quadratique *E*, est l'indice de performance qui est utilisé pour qualifier la précision de l'approximation. Sa forme mathématique permet de rapidement développer les équations normales. Cependant, cette erreur est une somme d'erreurs individuelles mises au carré et elle grandit avec le nombre de données. Pour qualifier la qualité de l'approximation, il serait préférable de prendre la moyenne de ces erreurs au carré. De plus, vu que cette moyenne représente des erreurs au carré, il est aussi préférable de prendre la racine carrée pour obtenir une erreur moyenne dans les unités de mesure originales.

Donc, bien que l'erreur quadratique *E* soit utile dans la formulation mathématique du problème, la racine carrée de la moyenne de ces erreurs, dénotée ***erreur RMS*** **(pour '*root mean square error*'),** donne une meilleure mesure physique de l'erreur d'approximation :

$$E_{RMS} = \sqrt{\frac{E}{N}} = \sqrt{\frac{\sum_{n=1}^{N}[g(x_n) - y_n]^2}{N}} \qquad \textbf{(1)}$$

Une autre façon de déterminer la qualité de l'approximation est l'utilisation d'un indice qui normalise les erreurs autour de valeurs moyennes. C'est le ***coefficient de détermination*** $\boldsymbol{R^2}$**.** C'est un rapport de variances :

$$R^2 = \frac{\sum_{n=1}^{N}\left[g(x_n) - \bar{y}\right]^2}{\sum_{n=1}^{N}\left[y_n - \bar{y}\right]^2} \quad \text{avec} \quad \bar{y} = \frac{1}{N}\sum_{n=1}^{N} y_n \qquad \textbf{(2)}$$

Une bonne approximation correspond à $R^2 \approx 1$, une mauvaise à $R^2 \approx 0$, mais il faut être très prudent dans l'interprétation de ce coefficient pour les raisons suivantes :

- $R^2 \approx 0$ indique toujours une mauvaise approximation.
- $R^2 \approx 1$ indique une bonne approximation seulement si le nombre de points *N* est suffisamment plus grand que le nombre de paramètres *M* dans l'équation candidate *g(x)*.
- La différence $N - M$ est le nombre de degré de liberté de l'approximation. Plus cette différence est grande, plus les données seront lissées. Un minimum de 3 est recommandé. Plus *M* se rapproche de *N*, plus la courbe d'approximation tend à interpoler les *N* données.
- Quand *M* = *N*, il y a interpolation des données par *g(x)* : l'erreur quadratique *E* est nulle et le coefficient de détermination R^2 est égal à 1. Cependant, si les données originales sont des mesures bruitées, l'interpolation n'est pas désirée. De plus, avec l'interpolation, le comportement de *g(x)* entre les points initiaux peut être complètement inacceptable. ¸
- Le coefficient de détermination est développé pour les prédictions linéaires et polynomiales. Il faut être très prudent lorsque l'on l'utilise dans d'autres situations.

En résumé, une bonne qualité de l'approximation de données bruitées vise $\boldsymbol{R^2 \approx 1}$ et $\boldsymbol{N - M \geq 3}$.

7.6 Approximation discrète sous MATLAB

- La fonction MATLAB **`sum`** est utilisée pour calculer les sommes. On peut aussi utiliser le produit matriciel correspondant.

- Par exemple, pour calculer l'erreur quadratique $E = \sum_{n=1}^{N}[g_n - y_n]^2$, on peut utiliser deux formulations équivalentes :

  ```
  sum((g-y).*(g-y))
  ```

  ou `(g-y)'*(g-y)` si g et y sont des colonnes.

  Cette seconde option s'avère plus rapide pour des grands vecteurs.

- L'erreur RMS peut être calculée avec la fonction `mean` :

  ```
  err_rms = sqrt(mean((g-y).*(g-y)))
  ```

  ou avec la fonction `rms` de MATLAB :

  ```
  err_rms = rms(g-y).
  ```

- La fonction `polyfit` permet de rechercher les coefficients du polynôme d'ordre m approximant au mieux une série de points de coordonnées (x, y) :

  ```
  COEFFS = polyfit(x,y,m)
  ```

- La fonction `polyval` permet ensuite de calculer les valeurs de l'approximation en d'autres points de coordonnées (x_{extrap}, y_{extrap}) :

  ```
  y_ extrap = polyval(COEFFS,x_extrap)
  ```

- Il faut réaliser qu'il y a deux séries de points qui sont utilisés sur MATLAB : (1) les données originales discrètes (x, y), et (2) les données x_{extrap} utilisées pour générer le graphique de l'approximation y_{extrap}.