# Méthodes Numériques (G3SEMN)*

Sup'Galilée, Ingénieurs Energétique, 1ère année

François Cuvelier

Laboratoire d'Analyse Géométrie et Applications
Institut Galilée
Université Paris XIII.

2026/02/10

*Compilé le 2026/02/10 à 06:22:04



## Plan du cours





## Plan





**Définition**

Soit $g$ une fonction et $q \in \mathbb{N}$. On dit que $g$ **se comporte comme un grand O de** $h^q$ quand h tend vers 0 si et seulement si il existe $H > 0$ et $C > 0$ tel que

$$\forall h \in ]-H, H[, \quad |g(h)| \leqslant C|h|^q.$$

On note alors $g(h) = \mathcal{O}(h^q)$.

On peut noter entre autres

- $\forall \alpha \in \mathbb{R}^*$,
$$g(h) = \mathcal{O}(h^q) \quad \Rightarrow \quad \alpha g(h) = \mathcal{O}(h^q),$$
- $\forall p \in \mathbb{Z},\ p + q \geqslant 0$,
$$g(h) = \mathcal{O}(h^q) \quad \Rightarrow \quad h^p g(h) = \mathcal{O}(h^{p+q}).$$
- $\forall (p, q) \in \mathbb{N}^2$, $\forall (\alpha_1, \alpha_2) \in \mathbb{R}^* \times \mathbb{R}^*$,
$$g_1(h) = \mathcal{O}(h^p),\ g_2(h) = \mathcal{O}(h^q) \quad \Rightarrow \quad \alpha_1 g_1(h) + \alpha_2 g_2(h) = \mathcal{O}(h^{\min(p,q)}).$$

**Proposition 1.1 : *Développement de Taylor***

Soit $f$ une application $f : [a,b] \longrightarrow \mathbb{R}$. Si $f \in \mathcal{C}^{r+1}([a,b])$ alors le **développement de Taylor à l'ordre** $r$ s'écrit

- $\forall (x,y) \in [a,b]^2$ il existe un $\xi \in ]\min(x,y), \max(x,y)[$ tel que

$$f(x) = f(y) + \sum_{k=1}^{r} \frac{f^{(k)}(y)}{k!}(x-y)^k + \frac{f^{(r+1)}(\xi)}{(r+1)!}(x-y)^{r+1} \quad (1)$$

- $\forall t \in [a,b]$, $\forall h \in \mathbb{R}^*$ vérifiant $(t+h) \in [a,b]$, il existe $\xi \in ]\min(t,t+h), \max(t,t+h)[$ tel quel

$$f(t+h) = f(t) + \sum_{k=1}^{r} \frac{h^k}{k!} f^{(k)}(t) + \frac{h^{r+1}}{(r+1)!} f^{(r+1)}(\xi) \quad (2)$$

- $\forall t \in [a,b]$, $\forall h \in \mathbb{R}^*$ vérifiant $(t+h) \in [a,b]$, alors

$$f(t+h) = f(t) + \sum_{k=1}^{r} \frac{f^{(k)}(t)}{k!} h^k + \mathcal{O}(h^{r+1}). \quad (3)$$



**Exercice 1**

Soit $\varphi : [a,b] \longrightarrow \mathbb{R}$ une fonction.

**Q. 1** *Montrer que si $\varphi \in \mathcal{C}^2([a,b];\mathbb{R})$ alors $\forall x \in [a,b[$, $\forall h > 0$ tel que $(x+h) \in [a,b]$, on a*

$$\frac{d\varphi}{dx}(x) = \frac{\varphi(x+h) - \varphi(x)}{h} + \mathcal{O}(h) \quad (1)$$

**Q. 2** *Montrer que si $\varphi \in \mathcal{C}^2([a,b];\mathbb{R})$ alors $\forall x \in ]a,b]$, $\forall h > 0$ tel que $(x-h) \in [a,b]$, on a*

$$\frac{d\varphi}{dx}(x) = \frac{\varphi(x) - \varphi(x-h)}{h} + \mathcal{O}(h) \quad (2)$$

**Q. 3** *Montrer que si $\varphi \in \mathcal{C}^3([a,b];\mathbb{R})$ alors $\forall x \in ]a,b[$, $\forall h > 0$ tel que $(x+h) \in [a,b]$ et $(x-h) \in [a,b]$, on a*

$$\frac{d\varphi}{dx}(x) = \frac{\varphi(x+h) - \varphi(x-h)}{2h} + \mathcal{O}(h^2) \quad (3)$$

**Q. 4** *Montrer que si $\varphi \in \mathcal{C}^4([a,b];\mathbb{R})$ alors $\forall x \in ]a,b[$, $\forall h > 0$ tel que $(x+h) \in [a,b]$ et $(x-h) \in [a,b]$, on a*

$$\frac{d^2\varphi}{dx^2}(x) = \frac{\varphi(x+h) - 2\varphi(x) + \varphi(x-h)}{h^2} + \mathcal{O}(h^2) \quad (4)$$



**Définition 1.1**

Soit $\varphi : \mathbb{R} \longrightarrow \mathbb{R}$ une fonction suffisament régulière. Soient $x \in \mathbb{R}$, $h \in \mathbb{R}^{+*}$, $n \in \mathbb{N}^*$ et $p \in \mathbb{N}^*$. On dit que $\square_h(x)$ est une **approximation d'ordre** $p$ **de** $\frac{d^n\varphi}{dx^n}(x)$ si

$$\frac{d^n\varphi}{dx^n}(x) - \square_h(x) = \mathcal{O}(h^p)$$

On a donc établi:
Soient $\varphi : \mathbb{R} \longrightarrow \mathbb{R}$ une fonction suffisament régulière, $x \in \mathbb{R}$ et $h \in \mathbb{R}^{+*}$.

- $\dfrac{\varphi(x+h) - \varphi(x)}{h}$ et $\dfrac{\varphi(x) - \varphi(x-h)}{h}$ sont des approximations d'ordre 1 de $\frac{d\varphi}{dx}(x)$,
- $\dfrac{\varphi(x+h) - \varphi(x-h)}{2h}$ est une approximation d'ordre 2 de $\frac{d\varphi}{dx}(x)$,
- $\dfrac{\varphi(x+h) - 2\varphi(x) + \varphi(x-h)}{h^2}$, est une approximation d'ordre 2 de $\frac{d^2\varphi}{dx^2}(x)$.

Existe-t'il d'autres approximations d'ordre 2 de $\frac{d\varphi}{dx}(x)$?



## TD

**Exercice 2**

Soit $\varphi : \mathbb{R} \longrightarrow \mathbb{R}$ une fonction suffisament régulière, $x \in \mathbb{R}$ et $h \in \mathbb{R}^{+*}$.

**Q. 1** *Montrer que*

$$\frac{d\varphi}{dx}(x) = \frac{-3\varphi(x) + 4\varphi(x+h) - \varphi(x+2h)}{2h} + \mathcal{O}(h^2) \quad (1)$$

**Q. 2** *Montrer que*

$$\frac{d\varphi}{dx}(x) = \frac{3\varphi(x) - 4\varphi(x-h) + \varphi(x-2h)}{2h} + \mathcal{O}(h^2) \quad (2)$$

Ces résultats impliquent:

$$\frac{-3\varphi(x) + 4\varphi(x+h) - \varphi(x+2h)}{2h} \quad \text{et} \quad \frac{3\varphi(x) - 4\varphi(x-h) + \varphi(x-2h)}{2h}$$

sont des approximations d'ordre 2 de $\frac{d\varphi}{dx}(x)$.

## Plan





**Définition 2.1**

Soit $N \in \mathbb{N}^*$. On appelle **discrétisation régulière de** $[a, b]$ **à** $N$ **pas ou** $(N+1)$ **points** l'ensemble des points $a + nh$, $n \in [\![0, N]\!]$ où le pas $h$ est donné par $h = (b - a)/N$.

Soit $y : [a, b] \longrightarrow \mathbb{R}$ une fonction suffisament régulière. On note $t^n = a + nh$, $n \in [\![0, N]\!]$, une **discrétisation régulière de** $[a, b]$. On appelle

- **différence finie progressive** l'approximation de $y'(t^n)$ suivante:

$$\frac{y(t^{n+1}) - y(t^n)}{h}, \quad \forall n \in [\![0, N-1]\!] \tag{5}$$

- **différence finie rétrograde** l'approximation de $y'(t^n)$ suivante:

$$\frac{y(t^n) - y(t^{n-1})}{h}, \quad \forall n \in [\![1, N]\!] \tag{6}$$

- **différence finie centrée** l'approximation de $y'(t^n)$ suivante:

$$\frac{y(t^{n+1}) - y(t^{n-1})}{2h}, \quad \forall n \in [\![1, N-1]\!] \tag{7}$$



On a démontré le lemme suivant

**Lemme 2.1**

Si $y \in \mathcal{C}^2([a, b])$ alors

$$y'(t^n) = \frac{y(t^{n+1}) - y(t^n)}{h} + \mathcal{O}(h), \quad \forall n \in [\![0, N-1]\!], \tag{8}$$

$$y'(t^n) = \frac{y(t^n) - y(t^{n-1})}{h} + \mathcal{O}(h), \quad \forall n \in [\![1, N]\!]. \tag{9}$$

Si $y \in \mathcal{C}^3([a, b])$ alors

$$y'(t^n) = \frac{y(t^{n+1}) - y(t^{n-1})}{2h} + \mathcal{O}(h^2), \quad \forall n \in [\![1, N-1]\!]. \tag{10}$$



## TD

**Exercice 3**

Soit $f \in \mathcal{C}^3([a, b]; \mathbb{R})$. On note $t^n$, $n \in [\![0, N]\!]$, une discrétisation **régulière** de $[a, b]$ de pas $h$. On note $\boldsymbol{F} \in \mathbb{R}^{N+1}$ le vecteur défini par $F_{n+1} = f(t^n)$, $\forall n \in [\![0, N]\!]$.

**Q. 1**

1. *Déterminer en fonction de $h$ et $\boldsymbol{F}$, un vecteur $\boldsymbol{V} \in \mathbb{R}^{N+1}$ vérifiant*

$$V_{n+1} = f'(t^n) + \mathcal{O}(h), \quad \forall n \in [\![0, N]\!].$$

2. *Ecrire une fonction algorithmique, nommée* Derive1Ordre1*, permettant, à partir du vecteur $\boldsymbol{F}$ et de la discrétisation régulière, de calculer le vecteur $\boldsymbol{V}$ précédent.*

**Q. 2**

1. *Connaissant uniquement $h$ et le vecteur $\boldsymbol{F}$, déterminer un vecteur $\boldsymbol{W} \in \mathbb{R}^{N+1}$ vérifiant*

$$\boldsymbol{W}_n = f'(t^n) + \mathcal{O}(h^2), \quad \forall n \in [\![0, N]\!]$$

2. *Ecrire une fonction algorithmique, nommée* Derive1Ordre2*, permettant, à partir du vecteur $\boldsymbol{F}$ et de la discrétisation régulière, de calculer le vecteur $\boldsymbol{W}$ précédent.*

## ❖ On cherche à *visualiser* l'erreur commise par les deux méthodes

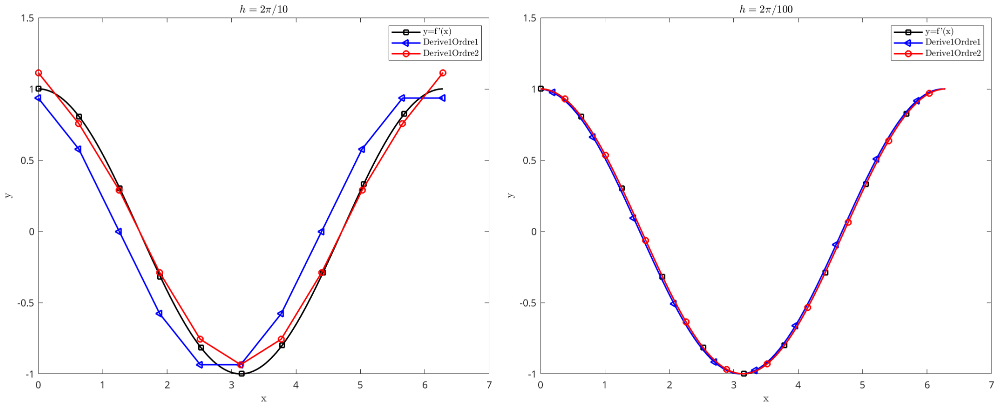



Figure: Représentation des dérivées numériques avec $f(x) = sin(x)$, $a = 0$, $b = 2\pi$. A gauche $h = \frac{2\pi}{10}$, à droite $h = \frac{2\pi}{100}$.



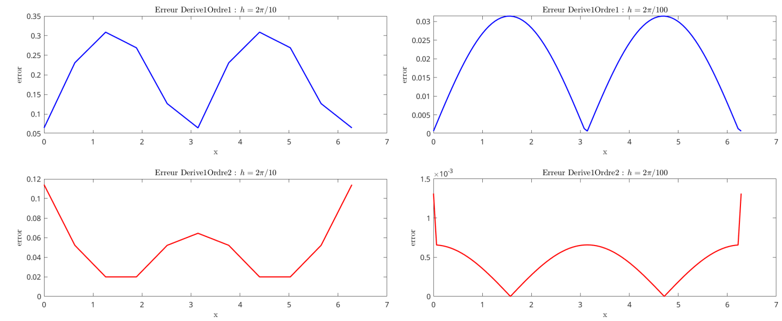



Figure: Erreur des dérivées numériques numériques avec $f(x) := sin(x)$, $a = 0$, $b = 2\pi$. A gauche $h = \frac{2\pi}{10}$, à droite $h = \frac{2\pi}{100}$.

- Dérivée première d'ordre 1 :

  $h$ a été divisé par 10 $\Longrightarrow$ l'erreur, $\mathcal{O}(h)$, divisée par 10.

- Dérivée première d'ordre 2 :

  $h$ a été divisé par 10 $\Longrightarrow$ l'erreur, $\mathcal{O}(h^2)$, divisée par 100.



### Exercice 4

On suppose écrites les fonctions Matlab/Octave Derive1Order1 et Derive1Order2 correspondant aux fonctions algorithmiques de l'exercice 3. Leurs syntaxes sont les suivantes:

V=Derive1Ordre1(h,F) et W=Derive1Ordre2(h,F)

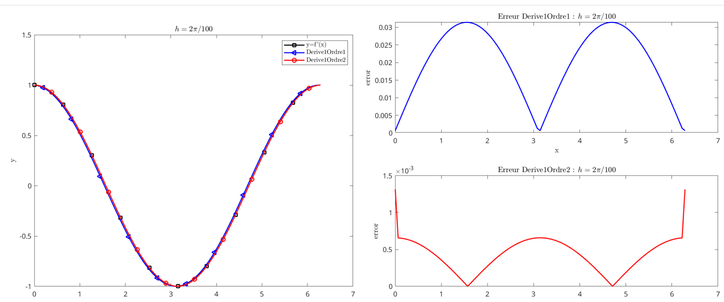



Figure: Avec $f(x) = sin(x)$, $a = 0$, $b = 2\pi$, $N = 100$, à gauche, les différentes dérivées, à droite les erreurs commises par les deux fonctions.

**Q. 1** *Ecrire un programme Matlab/Octave permettant de reproduire ces deux graphiques.*



## ❖ On cherche à *visualiser* l'ordre des deux méthodes

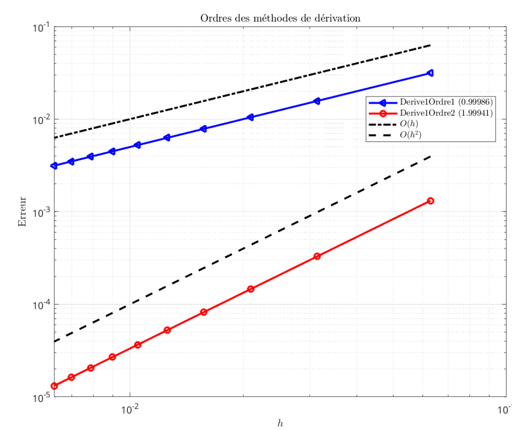


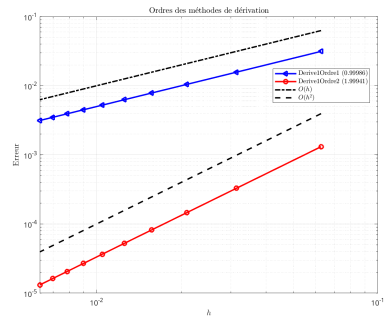

Figure: Dérivation numérique : mise en évidence de l'ordre des méthodes

Figure en échelle logarithmique : intérêt de *monter* en ordre.

## Exercice 5

Voici une figure mettant en évidence l'ordre des méthodes utilisées dans les fonctions Derive1Order1 et Derive1Order2 de l'exercice 3. On suppose écrites les fonctions Matlab/Octave Derive1Order1 et Derive1Order2 correspondant à ces deux fonctions algorithmiques et on suppose que leurs syntaxes sont les suivantes:

V=Derive1Ordre1(h,F) et W=Derive1Ordre2(h,F)




**Q. 1**

1. *Ecrire un programme Matlab/Octave permettant de calculer l'ensemble des données nécessaires à la représentation graphique de l'ordre des deux méthodes (voir figure).*
2. *A l'aide de ces données, calculer numériquement l'ordre des deux méthodes.*

Les commandes Matlab/Octave permettant de représenter des données en échelles logarithmique sont loglog, semilogx et semilogy. Elles s'utilisent globalement comme la fonction plot.

**Q. 2** *Ajouter au programme précédent le code permettant de reproduire la figure.*