Simulation Numérique 2 :

Formules de quadrature

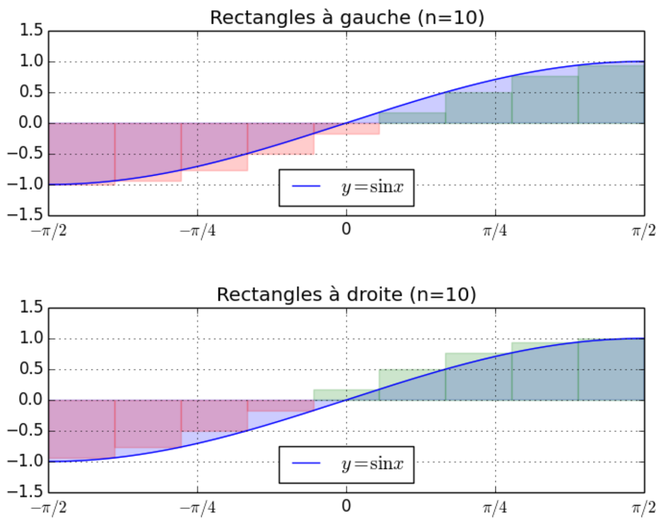
Le but de ce chapitre est d'étudier des algorithmes de calcul approché d'intégrale d'une fonction f sur un segment $[a, b]$.

I MÉTHODE DES RECTANGLES

Le cas le plus simple consiste à approcher f par une fonction constante par morceaux sur une subdivision $(a_0 = a, a_1, \dots, a_n = b)$ de $[a, b]$, c'est-à-dire constante sur chaque segment $[a_k, a_{k+1}]$ et donc d'approcher l'intégrale « aire sous la courbe » par des aires de rectangles.

$$\int_{a_k}^{a_{k+1}} f(t)\,\mathrm{d}t \approx (a_{k+1} - a_k) f(\xi_k) \quad \text{et} \quad \int_a^b f(t)\,\mathrm{d}t \approx \sum_{k=0}^{n-1} (a_{k+1} - a_k) f(\xi_k)$$

avec $\xi_k \in [a_k, a_{k+1}]$ pour tout k. Pour simplifier, on prendra $a_{k+1} - a_k = \frac{b-a}{n}$ constant.



1 Rectangles à gauche

a Formule

On choisit $\xi_k = a_k$.

$$\boxed{\int_a^b f(t)\,\mathrm{d}t \approx I_n(f) = \frac{b-a}{n}\sum_{k=0}^{n-1} f(a_k)}$$

b Algorithme

```
def rect_a_gauche(f, a, b, n):
    """valeur approchée de
l'intégrale de f entre a et b
par la méthode des rectangles
à gauche avec n sous-segments."""
    somme = f(a)
    h = (b - a) / n
    ak = a
    for k in range(n):
        ak += h
        somme += f(ak)
    return h * somme
```

Invariant d'entrée : `somme` contient $f(a_0)+\cdots+f(a_{k-1})$ et `ak` contient a_k.

Complexité (évaluations de f) : $T(n) = O(n)$.

Remarque

Pratique : cela ne coûte pas plus cher de passer de n à $n+1$ points que de passer de n à $2n$ points, en réutilisant les valeurs déjà calculées !

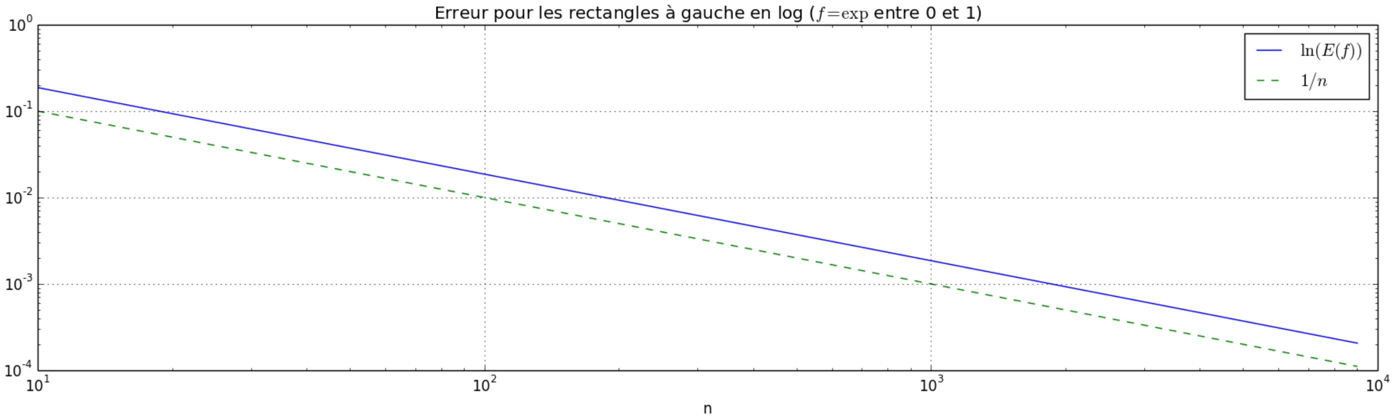
Version courte :

```
def rect_a_gauche(f, a, b, n):
    """valeur approchée de l'intégrale de f entre a et b par la
méthode des rectangles à gauche avec n sous-segments."""
    return h * sum(f(a + k * h) for k in range(n))
```

c Estimation de l'erreur

L'erreur vérifie $E(f) = \left|I_n(f) - \int_a^b f(t)\,\mathrm{d}t\right| \leqslant \frac{C}{n}$ avec $C = \frac{\sup|f'|}{2}|b-a|^2$.

Il suffit de tracer $\ln(E(f))$ (c'est au signe près le nombre de chiffres significatifs dans l'approximation de l'intégrale) en fonction de $\ln n$ pour le vérifier (on doit obtenir quelque chose s'approchant d'une droite de pente -1 !).



2 Rectangles à droite

a Formule

Cette fois, on choisit $\xi_k = a_{k+1}$.

$$\boxed{\int_a^b f(t)\,\mathrm{d}t \approx I_n(f) = \frac{b-a}{n}\sum_{k=0}^{n-1} f(a_{k+1})}$$

b Algorithme

```
def rect_a_droite(f, a, b, n):
    """valeur approchée de
l'intégrale de f entre a et b
par la méthode des rectangles
à droite avec n sous-segments."""
    somme = 0
    h = (b - a) / n
    ak = a
    for k in range(n):
        ak += h
        somme += f(ak)
    return h * somme
```

Invariant d'entrée : `somme` contient $f(a_1)+\cdots+f(a_{k-1})$ et `ak` contient a_k.

Complexité (évaluations de f) : $T(n) = O(n)$.

Version courte :

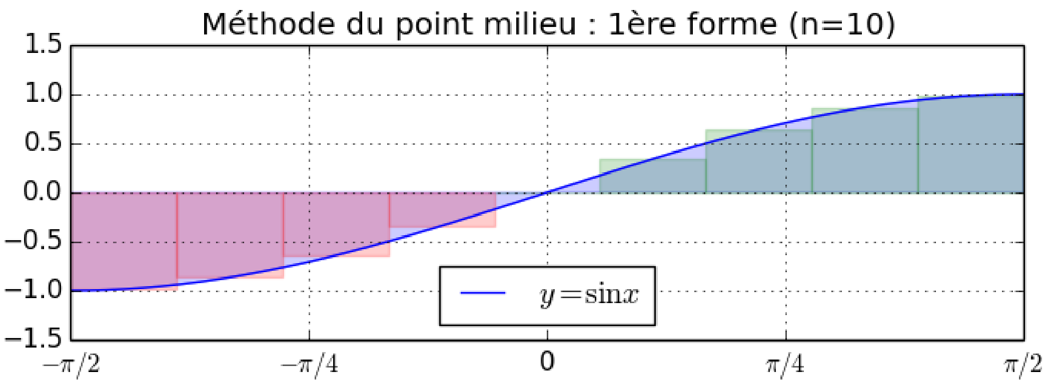
```
def rect_a_droite(f, a, b, n):
    """valeur approchée de l'intégrale de f entre a et b par la
méthode des rectangles à droite avec n sous-segments."""
    return h * sum(f(a + k * h) for k in range(1, n + 1))
```

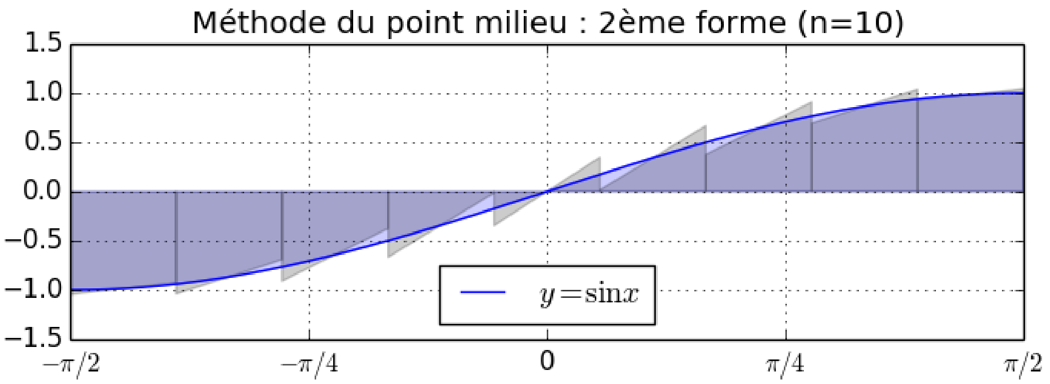
II MÉTHODE DU POINT MILIEU (HP)

1 Formule

Cette fois, on choisit $\xi_k = \dfrac{a_k + a_{k+1}}{2}$.

$$\boxed{\int_a^b f(t)\,\mathrm{d}t \approx \frac{b-a}{n}\sum_{k=0}^{n-1} f\left(\frac{a_k + a_{k+1}}{2}\right) = h\sum_{k=0}^{n-1} f\left(a_k + \frac{h}{2}\right)}$$





2 Algorithme

```
def point_milieu(f, a, b, n):
    """valeur approchée de
l'intégrale de f entre a et b
par la méthode du point
milieu avec n sous-segments."""
    somme = 0
    h = (b - a) / n
    milieu = a - h / 2
    for k in range(n):
        milieu += h
        somme += f(milieu)
    return h * somme
```

Invariant : au début de l'étape k, `somme` contient

$$f\left(\frac{a_0+a_1}{2}\right)+\cdots+f\left(\frac{a_{k-1}+a_k}{2}\right)$$

$$=f\left(a_0+\frac{h}{2}\right)+\cdots+f\left(a_{k-1}+\frac{h}{2}\right)$$

et `milieu` contient $a_k+\frac{h}{2}$.

Complexité (évaluations de f) : $T(n)=O(n)$.

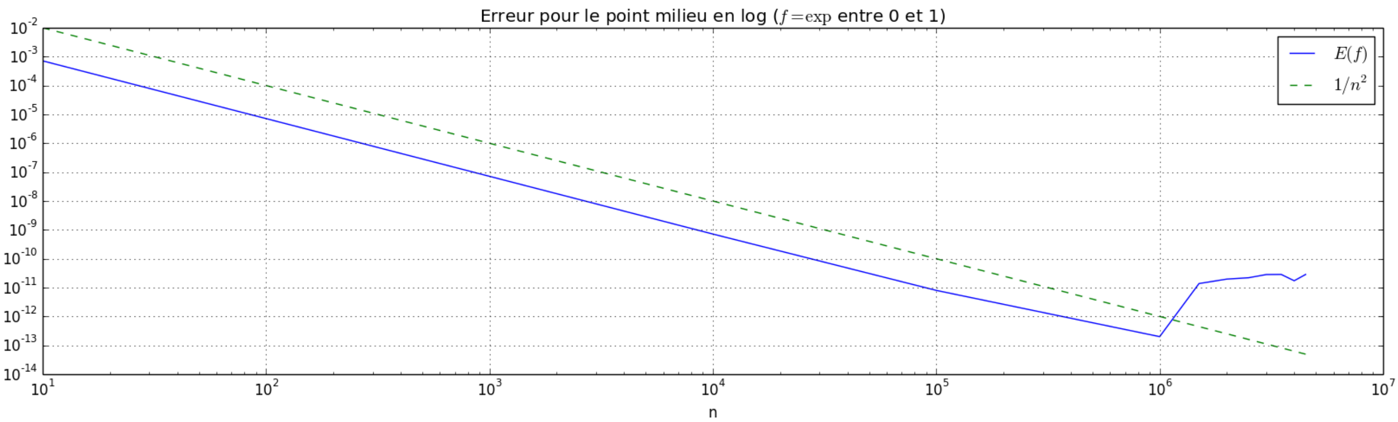
Version courte :

```
def point_milieu(f, a, b, n):
    """valeur approchée de l'intégrale de f entre a et b
par la méthode du point milieu avec n sous-segments."""
    return h * sum(f(a + (k + 1 / 2) * h) for k in range(n))
```

3 Estimation de l'erreur

L'erreur vérifie $E(f)\leqslant\frac{C}{n^2}$ avec $C=\frac{\sup\left|f''\right|}{24}|b-a|^3$.

Il suffit de tracer $\ln\left(E(f)\right)$ en fonction de $\ln n$ pour le vérifier (on doit obtenir une droite de pente -2).

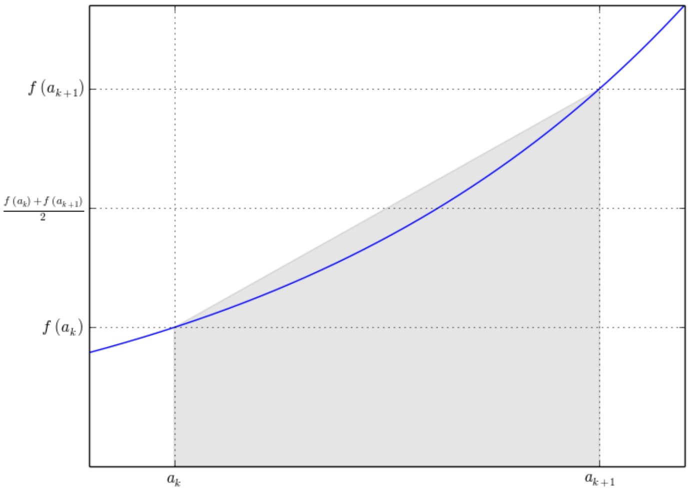


Comment expliquer le comportement à droite du graphe ?

III Méthode des trapèzes

Cette fois, on choisit d'approcher f par une droite (polynôme de degré 1) sur chaque sous-segment.

1 Formule



L'équation de la droite reliant $(a_k, f(a_k))$ à $(a_{k+1}, f(a_{k+1}))$ est

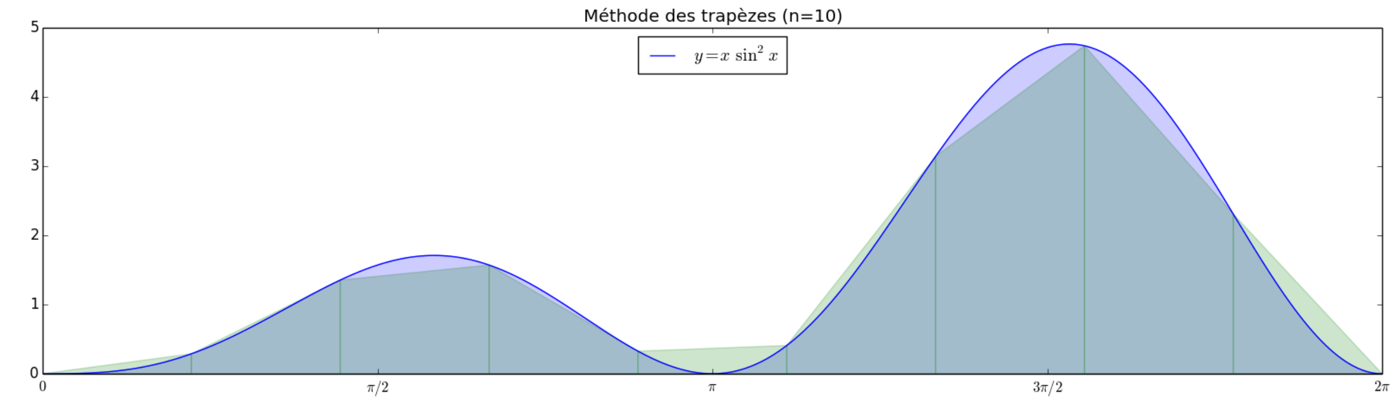
$$y = f(a_k) + \frac{f(a_{k+1}) - f(a_k)}{a_{k+1} - a_k}(x - a_k)$$

L'aire du trapèze formé avec l'axe des abscisse est facilement

$$(a_{k+1} - a_k)\frac{f(a_k) + f(a_{k+1})}{2}$$

d'où la formule de quadrature :

$$\boxed{\int_a^b f(t)\,\mathrm{d}t \approx \frac{b-a}{n}\sum_{k=0}^{n-1}\frac{f(a_k)+f(a_{k+1})}{2} = \frac{b-a}{n}\left(\frac{f(a)}{2} + \sum_{k=1}^{n-1} f(a_k) + \frac{f(b)}{2}\right)}$$



2 Algorithme

```
def trapezes(f, a, b, n):
    """valeur approchée de
l'intégrale de f entre a et b
par la méthode des trapèzes
avec n sous-segments."""
    somme = (f(a) + f(b)) / 2
    h = (b - a) / n
    ak = a
    for k in range(n - 1):
        ak += h
        somme += f(ak)
    return h * somme
```

Invariant : au début de l'étape k, `somme` contient

$$\frac{f(a)}{2} + f(a_1) + \cdots + f(a_{k-1}) + \frac{f(b)}{2}$$

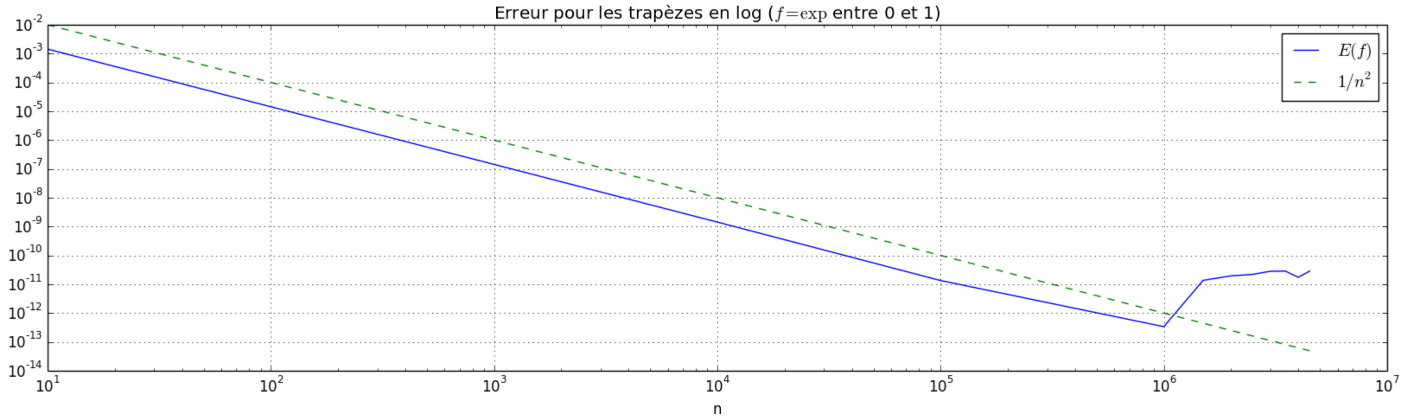
et `ak` contient a_k.

Complexité (évaluations de f) : $T(n) = O(n)$.

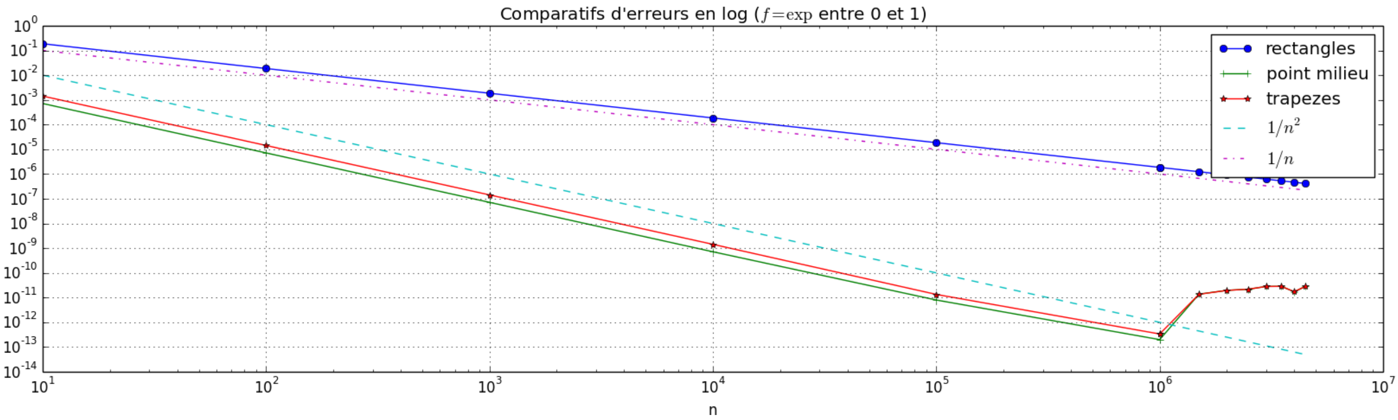
3 Estimation de l'erreur

L'erreur vérifie $E(f) \leqslant \dfrac{C}{n^2}$ avec $C = \dfrac{\sup|f''|}{12}|b-a|^3$.

Il suffit de tracer $\ln(E(f))$ en fonction de $\ln n$ pour le vérifier (on doit obtenir une droite de pente -2).



IV Comparaison



V Généralisation : méthodes de Newton-Cotes (HP)

1 Principe

On cherche à approcher l'intégrale $\int_a^b f(t)\,\mathrm{d}t$ d'une fonction continue sur un segment. On donne ici le principe général, le programme se limitant aux méthodes des rectangles et des trapèzes.

La formule de Chasles

$$\int_a^b f(t)\,\mathrm{d}t = \sum_{k=0}^{n-1} \int_{a_k}^{a_{k+1}} f(t)\,\mathrm{d}t$$

permet de faire des approximations sur des petits segments $[a_k, a_{k+1}]$ subdivisant $[a, b]$.

On peut choisir une **subdivision régulière** : $a_{k+1} - a_k$ indépendant de k (donc égal à $\dfrac{b-a}{n}$), soit $a_k = a + k\dfrac{b-a}{n}$.

Le réel $h = \dfrac{b-a}{n}$ est appelé **pas de discrétisation**.

Définition

Ces approximations dites **méthodes de quadrature élémentaires** sont de la forme

$$\int_{a_k}^{a_{k+1}} f(t)\,\mathrm{d}t \approx (a_{k+1}-a_k)\sum_{j=0}^{\ell}\omega_j f(\xi_{k,j}) = \frac{b-a}{n}\sum_{j=0}^{\ell}\omega_j f(\xi_{k,j})$$

où $\xi_{k,j} \in [a_k, a_{k+1}]$ et $\sum_{j=0}^{\ell}\omega_j = 1$ (formule exacte si f constante !).

Remarque

On remplace f sur $[a_k, a_{k+1}]$ par une somme de valeurs prises par f sur ce segment affectées d'un certain poids ω_j. C'est de l'interpolation ! On approche f par un polynôme d'interpolation de degré au plus ℓ. Les poids ω_j se calculent en intégrant les polynômes de Lagrange correspondant.

Les $\xi_{k,j}$ seront choisis régulièrement espacés dans $[a_k, a_{k+1}]$.

Pour $\ell = 0$, on retrouve les méthodes des rectangles au du point milieu, pour $\ell = 1$ la méthode des trapèzes.

Définition

La **formule de quadrature** obtenue donc est de la forme :

$$\int_a^b f(t)\,\mathrm{d}t \approx \frac{b-a}{n}\sum_{k=0}^{n-1}\sum_{j=0}^{\ell}\omega_j f(\xi_{k,j})$$

avec $\xi_{k,j} = a_k + \frac{j}{l}(a_{k+1}-a_k) = a_k + \frac{j}{l}h.$

Remarque

Le cours de mathématiques permettra de justifier que

$$\frac{b-a}{n}\sum_{k=0}^{n-1}\sum_{j=0}^{\ell}\omega_j f(\xi_{k,j}) \to \int_a^b f(t)\,\mathrm{d}t$$

lorsque $n \to +\infty$ (sommes de Riemann).

2 Degré, ordre, erreur

Définition

La méthode est dite **de degré** N si la formule approchée est exacte pour tout polynôme de degré au plus N et inexacte pour au moins un polynôme de degré $N+1$.

Remarques

R1 – Toutes les méthodes de quadrature sont de degré au moins 0 car $\sum_{j=0}^{\ell} \omega_j = 1$.

R2 – On a facilement que les méthodes des rectangles à gauche ou à droite sont de degré 0 et celle du point milieu ou des trapèzes sont de degré 1.

Définition

L'**erreur** de la méthode $\int_a^b f \approx I(f)$ est le réel $E(f) = \left| I(f) - \int_a^b f(t)\,\mathrm{d}t \right|$.

Remarque

On peut démontrer mathématiquement que si f est suffisamment régulière,

$$E(f) \leqslant \frac{C}{n^{N+1}} = C' h^{N+1}$$

où C, C' sont des constantes et N le degré de la méthode.
La méthode est alors dite d'ordre au moins $N+1$.

3 D'autres exemples

D'autres formules de quadrature de Newton-Cotes utilisées couramment :

- **Méthode de Simpson** : $\ell = 2$ (degré 3, ordre 4), $\omega_0 = \omega_2 = \frac{1}{6}$ et $\omega_1 = \frac{2}{3}$.
- **Méthode de Boole-Villarceau** : $\ell = 4$ (degré 5, ordre 6),

$$\omega_0 = \omega_4 = \frac{7}{90},\ \omega_1 = \omega_3 = \frac{16}{45},\ \omega_2 = \frac{2}{15}$$

- **Méthode de Weddle-Hardy** : $\ell = 6$ (degré 7, ordre 8),

$$\omega_0 = \omega_6 = \frac{41}{840},\ \omega_1 = \omega_5 = \frac{9}{35},\ \omega_2 = \omega_4 = \frac{9}{280},\ \omega_3 = \frac{34}{105}$$