

## Chapitre 3 – 01

### Intégration numérique

5 Février 2020

#### Savoirs et compétences :

- ❑ AN.S3 : Méthodes des rectangles et des trapèzes pour le calcul approché d'une intégrale sur un segment.
- ❑ SN.C1 : Réaliser un programme complet structuré
- ❑ SN.C2 : Étudier l'effet d'une variation des paramètres sur le temps de calcul, sur la précision des résultats, sur la forme des solutions pour des programmes d'ingénierie numérique choisis, tout en contextualisant l'observation du temps de calcul par rapport à la complexité algorithmique de ces programmes
- ❑ SN.C3 : Utiliser les bibliothèques de calcul standard
- ❑ SN.C4 : Utiliser les bibliothèques standard pour afficher les résultats sous forme graphique
- ❑ SN.C5 : Tenir compte des aspects pratiques comme l'impact des erreurs d'arrondi sur les résultats, le temps de calcul ou le stockage en mémoire.
- ❑ SN.S1 : Bibliothèques logicielles

1	<b>Rappels géométriques.</b>	2
2	<b>Méthodes géométriques.</b>	2
2.1	Méthode(s) des rectangles. . . . .	2
2.2	Méthode des trapèzes. . . . .	3
2.3	Supplément : méthode de Simpson. . . . .	4
2.4	Représentation graphique de ces différentes méthodes. . . . .	4
3	<b>Intégration numérique fournie par Python.</b>	4
4	<b>Applications</b>	6
4.1	Fonctions polynomiales . . . . .	6

## 1 Rappels géométriques.

**Propriété** L'aire d'un rectangle dont les côtés sont de longueurs  $a$  et  $b$  est  $ab$ .

**Propriété** L'aire d'un trapèze de hauteur  $h$  et de côtés de longueurs  $a$  et  $b$  est  $h \times \frac{a+b}{2}$ .

## 2 Méthodes géométriques.

L'objectif est d'approcher le réel

$$\int_a^b f,$$

où  $a$  et  $b$  sont deux réels (on supposera  $a < b$ ) et où  $f$  est une fonction réelle définie sur le segment  $[a, b]$ .

Les méthodes les plus simples reposent toutes sur le même principe.

- On subdivise le segment  $[a, b]$  en  $n$  segments :  $[x_0, x_1], \dots, [x_{n-1}, x_n]$  avec, pour tout  $0 \leq k \leq n$ ,

$$x_k = a + k \frac{b-a}{n}.$$

- Sur chaque segment  $[x_{k-1}, x_k]$ , on approche  $\int_{x_{k-1}}^{x_k} f$  par l'intégrale d'une fonction, que l'on espère proche de  $f$ .
- On renvoie la somme de ces  $n$  intégrales comme approximation de  $\int_a^b f$ .

L'objectif est dans chaque cas d'obtenir une approximation précise de l'intégrale à calculer, avec une complexité modérée. Notamment, nous utiliserons la définition suivante.

**Définition — Erreur d'approximation.** Soit  $A_n$  une approximation de  $\int_a^b f$  obtenue à partir de  $n$  segments.

L'erreur d'approximation de  $\int_a^b f$  par  $A_n$  est

$$\left| \int_a^b f - A_n \right|.$$

### 2.1 Méthode(s) des rectangles.

Cette méthode a été (ou sera bientôt) étudiée en détails dans le cours de mathématiques.

Pour chaque subdivision, on choisit un  $t_k \in [x_{k-1}, x_k]$  et l'on approche la fonction  $f$  sur  $[x_{k-1}, x_k]$  par la valeur

$f(t_k)$ , on approche donc  $\int_{x_{k-1}}^{x_k} f$  par  $\frac{b-a}{n} f(t_k)$  et l'approximation de  $\int_a^b f$  obtenue est donc

$$\frac{b-a}{n} \sum_{k=1}^n f(t_k).$$

Il y a donc autant de méthodes des rectangles que de manières de choisir ces nombres  $t_1, \dots, t_n$ . On n'en étudiera que deux.

#### 2.1.1 Méthode des rectangles à gauche.

Pour chaque  $1 \leq k \leq n$ , on choisit la borne gauche de  $[x_{k-1}, x_k]$ , soit  $x_{k-1}$ .

L'approximation de  $\int_a^b f$  calculée est donc

$$R_n = \frac{b-a}{n} \sum_{k=1}^n f\left(a + (k-1) \frac{b-a}{n}\right).$$

On code très simplement cette méthode, comme suit.

```
def Rg(f, a, b, N):
    """Approximation de l'intégrale de a à b de f
    Méthode des rectangles à gauche, N rectangles"""
    S = 0
    h = (b-a)/N
    for k in range(N):
        S = S + f(a + k*h)
    return S*h
```

### 2.1.2 Méthode des rectangles à droite.

Pour chaque  $1 \leq k \leq n$ , on choisit la borne droite de  $[x_{k-1}, x_k]$ , soit  $x_k$ .

L'approximation de  $\int_a^b f$  calculée est donc

$$R'_n = \frac{b-a}{n} \sum_{k=1}^n f\left(a + k \frac{b-a}{n}\right).$$

On code très simplement cette méthode, comme suit.

```
def Rd(f, a, b, N):
    """Approximation de l'intégrale de a à b de f
    Méthode des rectangles à droite, N rectangles"""
    S = 0
    h = (b-a)/N
    for k in range(1, N+1):
        S = S + f(a + k*h)
    return S*h
```

### 2.1.3 Propriétés communes.

On démontre dans le cours de mathématiques les propriétés suivantes.

**Propriété** Soit  $a, b \in \mathbb{R}$  avec  $a < b$ , soit  $f : [a, b] \rightarrow \mathbb{R}$ .

1. Si  $f$  est continue, alors toute approximation de  $\int_a^b f$  par une méthode des rectangles converge vers  $\int_a^b f$ .
2. Si  $f$  est lipschitzienne, alors toute méthode des rectangles a une erreur d'approximation en  $O\left(\frac{1}{n}\right)$ .

Les propriétés suivantes sont immédiates.

**Propriété** Soit  $a, b \in \mathbb{R}$  avec  $a < b$ , soit  $f : [a, b] \rightarrow \mathbb{R}$ .

1. Si  $f$  est constante, alors toute méthode des rectangles est exacte.
2. On a  $R_n - R'_n = (f(a) - f(b)) \frac{b-a}{n}$ . Si  $f(a) \neq f(b)$ , l'erreur d'approximation d'une des deux méthodes des rectangles (à gauche ou à droite) est exactement de l'ordre de  $\frac{1}{n}$ .

**R** En plus du calcul de la subdivision, le calcul d'une approximation par la méthode des rectangles avec  $n$  rectangles demande donc  $n$  appels à la fonction  $f$  et  $n$  addition (plus une multiplication de flottants).

## 2.2 Méthode des trapèzes.

Pour chaque subdivision, on approche la fonction  $f$  sur  $[x_{k-1}, x_k]$  par la corde à la courbe de  $f$  sur le segment

$[x_{k-1}, x_k]$ , on approche donc  $\int_{x_{k-1}}^{x_k} f$  par  $\frac{b-a}{2n} (f(x_{k-1}) + f(x_k))$  et l'approximation de  $\int_a^b f$  obtenue est donc

$$\begin{aligned} T_n &= \frac{b-a}{2n} \sum_{k=1}^n \left( f\left(a + (k-1) \frac{b-a}{n}\right) + f\left(a + k \frac{b-a}{n}\right) \right) \\ &= \frac{S_n + S'_n}{2} \\ &= \frac{b-a}{n} \left( \frac{f(a)}{2} + \sum_{k=1}^{n-1} f\left(a + k \frac{b-a}{n}\right) + \frac{f(b)}{2} \right). \end{aligned}$$

**R** Dans la dernière écriture, on a pris soin d'effectuer le moins d'opérations possible.

On code très simplement cette méthode, comme suit.

```
def T(f, a, b, N):
    """Approximation de l'intégrale de a à b de f
    Méthode des trapèzes, N trapèzes"""
    S = 0.5*(f(a) + f(b))
    h = (b-a)/N
    for k in range(1, N):
        S = S + f(a + k*h)
    return S*h
```

On a les propriétés suivantes.

**Propriété** Soit  $a, b \in \mathbb{R}$  avec  $a < b$ , soit  $f : [a, b] \rightarrow \mathbb{R}$ .

1. Si  $f$  est continue, alors l'approximation de  $\int_a^b f$  par la méthode des trapèzes converge vers  $\int_a^b f$ .
2. Si  $f$  est  $\mathcal{C}^2$ , alors la méthode des trapèzes a une erreur d'approximation en  $O\left(\frac{1}{n^2}\right)$ .
3. Si  $f$  est affine, alors la méthode des trapèzes est exacte.

**R** En plus du calcul de la subdivision, le calcul d'une approximation par la méthode des trapèzes avec  $n$  trapèzes demande donc  $n + 1$  appels à la fonction  $f$  et  $n + 1$  addition (plus quelques opérations sur les flottants).

### 2.3 Supplément : méthode de Simpson.

Pour chaque subdivision, on approche la fonction  $f$  sur  $[x_{k-1}, x_k]$  par la fonction polynomiale de degré 2 coïncidant avec  $f$  en  $x_{k-1}$ ,  $\frac{x_{k-1} + x_k}{2}$  et en  $x_k$ , on approche donc  $\int_{x_{k-1}}^{x_k} f$  par  $\frac{b-a}{6n} \left( f(x_{k-1}) + 4f\left(\frac{x_{k-1} + x_k}{2}\right) + f(x_k) \right)$  et l'approximation de  $\int_a^b f$  obtenue est donc

$$S_n = \frac{b-a}{6n} \sum_{k=1}^n \left( f\left(a + (k-1)\frac{b-a}{n}\right) + 4f\left(a + \left(k - \frac{1}{2}\right)\frac{b-a}{n}\right) + f\left(a + k\frac{b-a}{n}\right) \right) \\ = \frac{b-a}{6n} \left( f(a) + 2 \sum_{k=1}^{n-1} f\left(a + k\frac{b-a}{n}\right) + 4 \sum_{k=1}^n \left( a + \left(k - \frac{1}{2}\right)\frac{b-a}{n} \right) + f(b) \right).$$

**R** Dans la dernière écriture, on a pris soin d'effectuer le moins d'opérations possible.

On a les propriétés suivantes.

**Propriété** Soit  $a, b \in \mathbb{R}$  avec  $a < b$ , soit  $f : [a, b] \rightarrow \mathbb{R}$ .

1. Si  $f$  est continue, alors l'approximation de  $\int_a^b f$  par la méthode de Simpson converge vers  $\int_a^b f$ .
2. Si  $f$  est  $\mathcal{C}^4$ , alors la méthode de Simpson a une erreur d'approximation en  $O\left(\frac{1}{n^4}\right)$ .
3. Si  $f$  est polynomiale de degré au plus deux, alors la méthode de Simpson est exacte.

### 2.4 Représentation graphique de ces différentes méthodes.

On représente l'approximation de la courbe d'une fonction sur un segment, par ces quatre méthodes (voir figure 1).

## 3 Intégration numérique fournie par Python.

La bibliothèque `scipy.integrate` contient la fonction `quad`, qui met en œuvre une méthode numérique de calcul approché d'intégrales. Elle utilise la bibliothèque Fortran QUADPACK. Les méthodes numériques utilisées sont bien plus complexes que celles présentées ici, mais reposent sur la même idée : approcher l'intégrale de la

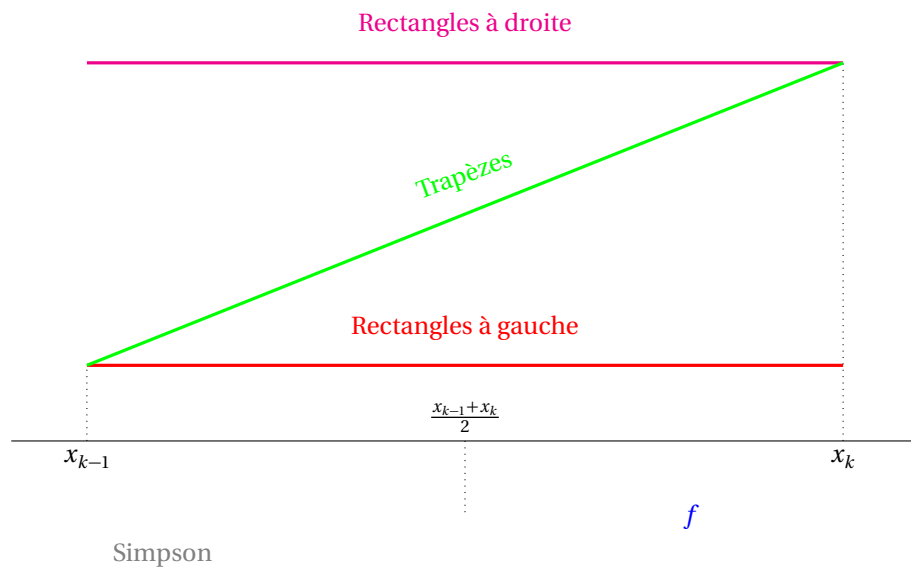


FIGURE 1 – Approximations sur  $[x_{k-1}, x_k]$  de la fonction  $f$  utilisées.

fonction par une combinaison linéaire d'évaluations de cette fonction, les points d'évaluation et les poids étant judicieusement choisis.

On remarquera que la fonction quad peut prendre en argument une précision et donne un résultat dont l'erreur d'approximation est inférieure à cette précision!

## TD

Sources :

**Savoirs et compétences :**

- ❑ AN.S3 : Méthodes des rectangles et des trapèzes pour le calcul approché d'une intégrale sur un segment.
- ❑ SN.C1 : Réaliser un programme complet structuré
- ❑ SN.C2 : Étudier l'effet d'une variation des paramètres sur le temps de calcul, sur la précision des résultats, sur la forme des solutions pour des programmes d'ingénierie numérique choisis, tout en contextualisant l'observation du temps de calcul par rapport à la complexité algorithmique de ces programmes
- ❑ SN.C3 : Utiliser les bibliothèques de calcul standard
- ❑ SN.C4 : Utiliser les bibliothèques standard pour afficher les résultats sous forme graphique
- ❑ SN.C5 : Tenir compte des aspects pratiques comme l'impact des erreurs d'arrondi sur les résultats, le temps de calcul ou le stockage en mémoire.
- ❑ SN.S1 : Bibliothèques logicielles

**4 Applications****4.1 Fonctions polynomiales**

**Q 1 :** Tracer les nuages de points correspondant, pour tout  $0 \leq p < 1000$  aux calculs de  $\int_0^1 t^p dt$  par les méthodes suivantes : rectangles à gauche, rectangles à droite, trapèzes, fonction quad de `scipy` et la valeur exacte. On utilisera 1000 rectangles ou trapèzes à chaque fois.

**Q 2 :** Produire un second graphique en faisant apparaître les erreurs d'approximations des quatre premières méthodes en utilisant une échelle appropriée.

**Q 3 :** Tracer les nuages de points correspondant, pour tout  $1 \leq n < 1000$  aux calculs de  $\int_0^1 t^2 dt$  par les méthodes suivantes : rectangles à gauche, rectangles à droite, trapèzes et la valeur exacte, en utilisant  $n$  rectangles ou trapèzes.

**Q 4 :** Produire un second graphique en faisant apparaître les erreurs d'approximations des quatre premières méthodes en utilisant une échelle appropriée.