

Calcul approché d'une intégrale

En intégration numérique, on cherche à calculer une valeur *approchée* d'une intégrale définie $I = \int_a^b f(x)dx$, où f est une fonction réelle de variable réelle définie et continue sur un intervalle fermé borné $[a ; b]$.

Si l'on connaît *explicitement* une primitive de f sur $[a ; b]$, un tel calcul ne présente aucune difficulté, à ceci près que le problème du calcul *approché* reste parfois posé.

Mais c'est bien loin d'être toujours le cas, soit qu'une telle primitive ne puisse être déterminée simplement à partir des fonctions usuelles (par exemple si f est définie par $f(x) = e^{-x^2}$), soit que la fonction f elle-même ne soit connue qu'en un nombre fini de points (par exemple en sciences expérimentales).

Dans le but de comparer les différentes méthodes proposées, nous les appliquerons systématiquement au même exemple, le calcul approché de $I = \int_0^1 \frac{4}{1+x^2} dx \dots$ La fonction est simple, la valeur de I très célèbre, puisqu'elle est égale à π . D'autres exemples, comme $\int_0^1 e^{x^2} dx$, peuvent aussi être utilisés.

Sommaire

Chapitre 15. Calcul approché d'une intégrale	335
1. Intégration numérique par interpolation	337
1.1 Formule de quadrature à $n + 1$ points	337
1.2 Intégration numérique et interpolation	337
1.3 Exactitude d'une formule de quadrature	338
1.4 Une fonction pour le calcul des polynômes L_i	339
1.5 Les formules de Newton-Cotes	340
1.6 Une première fonction pour le calcul des coefficients	341
1.7 Une autre fonction pour le calcul des coefficients	343
1.8 Calcul d'une intégrale par interpolation	346
1.9 Évaluation de l'erreur	347
1.10 Vers les méthodes composites	347

2.	Méthode des rectangles	348
2.1	Principe de la méthode	348
2.2	Méthode composite des rectangles	348
2.3	Une fonction pour la méthode des rectangles	349
2.4	Illustration graphique	351
2.5	Méthode des rectangles avec une fonction de classe C^1	355
2.6	Vers une amélioration de la méthode des rectangles	355
3.	La méthode des rectangles avec point médian	359
3.1	Principe de la méthode	359
3.2	Évaluation de l'erreur	361
3.3	Méthode composite des rectangles avec point médian	362
3.4	Écriture de fonctions	363
4.	Méthode des trapèzes	366
4.1	Principe de la méthode	366
4.2	Majoration de l'erreur due à la méthode	366
4.3	Méthode composite des trapèzes	369
4.4	Écriture d'une fonction	370
4.5	Illustration graphique	372
5.	Méthode de Simpson	373
5.1	Principe de la méthode	373
5.2	Majoration de l'erreur due à la méthode	376
5.3	Méthode composite de Simpson	379
5.4	Écriture d'une fonction	380
5.5	Illustration graphique	381
6.	La méthode de Romberg	384
6.1	Développement limité de la méthode des trapèzes	384
6.2	Application à une accélération de convergence	384
6.3	Mise en œuvre sur le tableur	386

1. Intégration numérique par interpolation

1.1 Formule de quadrature à $n + 1$ points

Soit f une fonction continue sur un intervalle $[a ; b]$.

D'un point de vue très général, le principe de l'intégration numérique consiste à remplacer le calcul de $I = \int_a^b f(x) dx$ par celui de la somme *finie* :

$$\sum_{k=0}^n A_n^k f(x_k) = A_n^0 f(x_0) + A_n^1 f(x_1) + \dots + A_n^n f(x_n)$$

où les x_i désignent $n + 1$ réels de l'intervalle $[a ; b]$.

Dans cette dernière formule, les coefficients A_n^k ne dépendent pas de la fonction f ¹.

Définition

Une telle formule est appelée *formule de quadrature à $n + 1$ points*.

1.2 Intégration numérique et interpolation

On sait qu'on peut approcher la fonction f par un polynôme d'interpolation g .

Une des méthodes classiques d'intégration numérique consiste à remplacer le calcul de $I = \int_a^b f(x) dx$ par celui de $\int_a^b g(x) dx$, intégrale d'un polynôme, ceci pour au moins deux raisons :

l'intégration des polynômes est très simple et ne nécessite que les quatre opérations élémentaires ;

il arrive fréquemment, pour les courbes expérimentales par exemple, qu'on ne connaisse f qu'en certains de ses points x_i .

Supposons donc que g soit le polynôme d'interpolation de Lagrange aux points d'abscisse :

$$x_0 = a < x_1 < \dots < x_{n-1} < x_n = b.$$

g est donc le polynôme de degré n qui vérifie pour tout entier naturel i inférieur ou égal à n :

$$g(x_i) = f(x_i).$$

Or on sait comment exprimer ce polynôme. Il peut s'écrire :

$$g(x) = \sum_{i=0}^n (L_i(x) \times f(x_i)) \quad \text{où} \quad L_i(x) = \frac{\prod_{\substack{k=0 \\ k \neq i}}^n (x - x_k)}{\prod_{\substack{k=0 \\ k \neq i}}^n (x_i - x_k)}.$$

Rappelons que $L_i(x_i) = 1$ et $L_i(x_j) = 0$ lorsque $i \neq j$.

¹ On réutilise donc les mêmes coefficients lorsqu'on travaille avec une autre fonction.

Par conséquent :

$$\int_a^b g(x) dx = \int_a^b \sum_{j=0}^n (L_j(x) \times f(x_j)) dx = \sum_{i=0}^n \left(\int_a^b L_i(x) dx \right) f(x_i) = \sum_{i=0}^n A_i^n f(x_i)$$

en posant $A_i^n = \int_a^b L_i(x) dx$.

Les coefficients A_i^n ne dépendent pas de la fonction f , seulement des points d'interpolation choisis.

Conformément à ce que nous avons dit plus haut, remplacer la fonction f par le polynôme qui l'interpole aux points x_i , $0 \leq i \leq n$, conduit bien à une formule de quadrature à $n + 1$ points, dite formule de quadrature à $n + 1$ points **de type interpolation**.

Formule de quadrature à $n + 1$ points de type interpolation

Une égalité du type $I = \int_a^b f(x) dx \approx \int_a^b g(x) dx$, où g est un polynôme interpolant f sur l'intervalle $[a ; b]$, est appelée formule de quadrature à $n + 1$ points de type *interpolation*.

Reste évidemment à choisir le degré du polynôme interpolateur, et les x_i « au mieux »².

1.3 Exactitude d'une formule de quadrature

En remplaçant la fonction f par son polynôme interpolateur g , on commet une erreur qu'il est essentiel de connaître, et qu'on cherchera à majorer. Lorsque la fonction f est elle-même un polynôme de degré au plus n , elle est égale à son polynôme interpolateur, et la formule d'intégration numérique est exacte. En d'autres termes, une formule de quadrature à $n + 1$ points de type interpolation est forcément exacte au moins sur l'ensemble des polynômes de degré n .

Plus généralement, on définit la notion d'exactitude d'une formule de quadrature, pas forcément de type interpolation d'ailleurs, de la façon suivante :

Une formule de quadrature est dite *exacte* sur un ensemble V si l'intégrale d'une fonction de V est égale exactement à sa valeur obtenue par la formule de quadrature considérée.
On dit qu'elle est d'ordre n si elle est exacte sur l'ensemble des polynômes de degré n .

On dispose alors du résultat suivant :

Théorème

Une formule de quadrature à $n + 1$ points est exacte sur l'espace vectoriel $\mathcal{P}_n$ des polynômes de degré au plus n (donc d'ordre n) si et seulement si elle est du type interpolation à $n + 1$ points.

Démonstration

Considérons donc une formule de quadrature *exacte* sur l'espace vectoriel $\mathcal{P}_n$ des polynômes de degré au plus n . On a donc :

$$\int_a^b P(x) dx = \sum_{k=0}^n A_n^k P(x_k) \text{ pour } P \in \mathcal{P}_n.$$

En particulier l'égalité a lieu pour le polynôme L_j , ce qui donne :

$$\int_a^b L_j(x) dx = \sum_{k=0}^n A_n^k L_j(x_k) = A_n^j.$$

² En particulier, il n'est pas dit ici que les x_i doivent être régulièrement espacés.

Par suite, $I = \int_a^b f(x) dx = \sum_{k=0}^n \left(\int_a^b L_k(x) dx \right) f(x_k) = \int_a^b \left(\sum_{k=0}^n L_k(x) f(x_k) \right) dx$, ce qui prouve que la formule de quadrature considérée est de type interpolation à $n + 1$ points.

Réciproquement, si la formule est de type interpolation à $n + 1$ points, alors quel que soit le polynôme P de degré au plus n , P est confondu avec son polynôme d'interpolation à $n + 1$ points. La formule est donc exacte sur $\mathcal{P}_n$, l'espace vectoriel des polynômes de degré au plus n .

1.4 Une fonction pour le calcul des polynômes L_i

Nous travaillerons avec des formules de quadrature de type interpolation. Nous aurons donc besoin de calculer les polynômes L_i qui interviennent dans la détermination du polynôme d'interpolation de Lagrange. On rappelle qu'ils sont définis par :

$$L_i(x) = \frac{\prod_{\substack{k=0 \\ k \neq i}}^n (x - x_k)}{\prod_{\substack{k=0 \\ k \neq i}}^n (x_i - x_k)} \quad \text{où les } x_i \text{ sont les abscisses des points d'interpolation.}$$

En entrée, on dispose de la liste des abscisses d'interpolation, depuis x_0 jusqu'à x_n ³, ainsi que l'indice i pour lequel on veut calculer L_i . On commence par fabriquer le numérateur de L_i : on calcule le produit de tous les $x - x_k$ en utilisant une instruction de liste pour que le calcul soit effectué en une seule fois ; enfin il faut penser à retirer de ce produit le facteur $x - x_i$.

Le dénominateur est tout simplement égal à ce que l'on obtient en remplaçant dans le numérateur x par x_i .

On obtient alors la fonction suivante :

```
calcul_li
Define LibPriv calcul_li(lx,i)=
Func
Local p
product(x-lx)
x-lx[i+1] → p
Return p / p[x=lx[i+1]]
EndFunc
```

Les résultats sont conformes à ceux que l'on attend :

$\text{calcul_li}(\{1,2,4,7\},0)$	$\frac{-(x-7) \cdot (x-4) \cdot (x-2)}{18}$
$\text{calcul_li}(\{1,2,4,7\},1)$	$\frac{(x-7) \cdot (x-4) \cdot (x-1)}{10}$
$\text{calcul_li}(\{1,2,4,7\},2)$	$\frac{-(x-7) \cdot (x-2) \cdot (x-1)}{18}$
$\text{calcul_li}(\{1,2,4,7\},3)$	$\frac{(x-4) \cdot (x-2) \cdot (x-1)}{90}$
$\frac{-(x-7) \cdot (x-4) \cdot (x-2)}{18} \Big _{x=\{1,2,4,7\}}$	$\{1,0,0,0\}$

³ Attention, les listes sont numérotées à partir de 1.

1.5 Les formules de Newton-Cotes

Les formules de Newton-Cotes sont des formules de quadrature de type interpolation, faisant intervenir une subdivision *régulière* de l'intervalle $[a ; b]$:

$$x_0 = a, x_1 = a + h, x_2 = a + 2h, \dots, x_n = b = a + nh \text{ avec } h = \frac{b-a}{n}.$$

Elles recouvrent tous les cas habituellement utilisés en intégration approchée (rectangle, trapèze, Simpson).

Elles sont dites de type *fermé*, car les extrémités de l'intervalle sont dans la subdivision.

Étudions quelques cas, selon le degré n du polynôme interpolateur.

- **Tout d'abord quand $n = 1$** , on interpole la fonction par le polynôme de degré 1 qui prend les mêmes valeurs que f en a et b .

Conformément à ce que nous avons écrit plus haut, nous pouvons écrire :

$$\int_a^b g(x) dx = A_0^1 f(a) + A_1^1 f(b) \text{ avec } A_0^1 = \int_a^b L_0(x) dx \text{ et } A_1^1 = \int_a^b L_1(x) dx.$$

Il reste à faire le calcul de $A_0^1 = \int_a^b L_0(x) dx$ et $A_1^1 = \int_a^b L_1(x) dx$.

$calcul_li(\{a,b\},0)$	$\frac{x-b}{a-b}$
$\int_a^b \left(\frac{x-b}{a-b} \right) dx$	$\frac{-(a-b)}{2}$
$e:=calcul_li(\{a,b\},1): \int_a^b (e) dx$	$\frac{-(a-b)}{2}$

On a donc :

$$A_0^1 = \frac{b-a}{2} \text{ et } A_1^1 = \frac{b-a}{2}.$$

Concrètement, on retrouve la méthode des trapèzes sur l'intervalle $[a, b]$, pour laquelle on approxime l'intégrale par :

$$\int_a^b g(x) dx = A_0^1 f(a) + A_1^1 f(b) = \frac{b-a}{2} (f(a) + f(b))$$

Poursuivons avec $n = 2$: on interpole cette fois la fonction par le trinôme du second degré qui prend les mêmes valeurs que f en a , $\frac{a+b}{2}$ et b .

On écrit cette fois :

$$\int_a^b g(x) dx = A_0^2 f(a) + A_1^2 f\left(\frac{a+b}{2}\right) + A_2^2 f(b)$$

$$\text{avec } A_0^2 = \int_a^b L_0(x) dx, A_1^2 = \int_a^b L_1(x) dx \text{ et } A_2^2 = \int_a^b L_2(x) dx.$$

⁴ On reconnaît la méthode de Simpson.

Calculons les coefficients avec notre TI-Nspire :

$e:=calcul_li\left(\left\{a,\frac{a+b}{2},b\right\},0\right):\int_a^b (e)dx$	$\frac{-(a-b)}{6}$
$e:=calcul_li\left(\left\{a,\frac{a+b}{2},b\right\},1\right):\int_a^b (e)dx$	$\frac{-2\cdot(a-b)}{3}$
$e:=calcul_li\left(\left\{a,\frac{a+b}{2},b\right\},2\right):\int_a^b (e)dx$	$\frac{-(a-b)}{6}$

On a donc :

$$A_0^2 = \frac{b-a}{6}, \quad A_1^2 = \frac{2(b-a)}{3} \quad \text{et} \quad A_2^2 = \frac{b-a}{6}.$$

On retrouve la formule de Simpson, qui affirme que :

$$\int_a^b g(x)dx = \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right).$$

Dans les A_k^n que nous venons de calculer, apparaît systématiquement le facteur $b-a$. Cela conduit à poser $B_k^n = \frac{A_k^n}{b-a}$, si bien que :

$$B_0^1 = B_1^1 = \frac{1}{2}; \quad B_0^2 = B_2^2 = \frac{1}{6} \quad \text{et} \quad B_1^2 = \frac{2}{3}.$$

Nous nous bornerons dans la suite à ne calculer que les coefficients B_k^n .

Avec $n = 3$, on obtient les coefficients suivants :

$$B_0^3 = B_3^3 = \frac{1}{8} \quad \text{et} \quad B_1^3 = B_2^3 = \frac{3}{8}$$

comme le montrent les écrans suivants :

$e:=calcul_li\left(\text{seq}\left(a+k\cdot\frac{b-a}{3},k,0,3\right),0\right):\frac{1}{b-a}\cdot\int_a^b (e)dx$	$\frac{1}{8}$
$e:=calcul_li\left(\text{seq}\left(a+k\cdot\frac{b-a}{3},k,0,3\right),1\right):\frac{1}{b-a}\cdot\int_a^b (e)dx$	$\frac{3}{8}$
$e:=calcul_li\left(\text{seq}\left(a+k\cdot\frac{b-a}{3},k,0,3\right),2\right):\frac{1}{b-a}\cdot\int_a^b (e)dx$	$\frac{3}{8}$
$e:=calcul_li\left(\text{seq}\left(a+k\cdot\frac{b-a}{3},k,0,3\right),3\right):\frac{1}{b-a}\cdot\int_a^b (e)dx$	$\frac{1}{8}$

1.6 Une première fonction pour le calcul des coefficients

Évidemment, on pourrait poursuivre avec le tableur, en écrivant une fonction qui donne directement la liste des coefficients correspondant à un entier donné.

On garde le même principe que dans les calculs que nous venons de faire : une instruction **seq** génère la liste des abscisses régulièrement espacées. Une boucle **For** calcule chacun des coefficients et les mémorise dans une liste.

coeff	2/8
Define LibPriv coeff (n)=	
Func	
Local l,res,i,e	
seq($a+k \cdot \frac{b-a}{n}$,k,0,n)→l:{ }→res	
For i,0,n	
calcul_li(l,i)→e	
$\frac{1}{b-a} \cdot \int_a^b (e)dx \rightarrow e$	
augment(res,{e})→res	
EndFor	
Return res	
EndFunc	
coeff(1)	$\left\{ \frac{1}{2}, \frac{1}{2} \right\}$
coeff(2)	$\left\{ \frac{1}{6}, \frac{2}{3}, \frac{1}{6} \right\}$
coeff(3)	$\left\{ \frac{1}{8}, \frac{3}{8}, \frac{3}{8}, \frac{1}{8} \right\}$
coeff(4)	$\left\{ \frac{7}{90}, \frac{16}{45}, \frac{2}{15}, \frac{16}{45}, \frac{7}{90} \right\}$

On peut remarquer que les temps de calcul pour des valeurs de n plus grandes deviennent assez rapidement importants : 2 minutes 15 pour **coeff(50)** et presque un quart d'heure pour **coeff(70)** sur mon ordinateur portable.

On peut alors construire une feuille de calcul : partant de A1 mis à 0, on saisit l'instruction **augment({A1+1},coeff(A1+1))** dans la zone grisée de la colonne B, puis on la recopie successivement dans les zones grisées des colonnes C, D, etc. On obtient les résultats suivants :

A	B	C	D	E	F	G	H	I	J	K
•	=augr	=augr	=augr	=augr	=augr	=augr	=augr	=augr	=augr	=augr
1	0	1	2	3	4	5	6	7	8	9
2		1/2	1/6	1/8	7/90	19/288	41/840	751/17280	989/28350	2857/89600
3		1/2	2/3	3/8	16/45	25/96	9/35	3577/17280	2944/14175	15741/89600
4			1/6	3/8	2/15	25/144	9/280	49/640	-464/14175	27/2240
5				1/8	16/45	25/144	34/105	2989/17280	5248/14175	1209/5600
6					7/90	25/96	9/280	2989/17280	-454/2835	2889/44800
7						19/288	9/35	49/640	5248/14175	2889/44800
8							41/840	3577/17280	-464/14175	1209/5600
9								751/17280	2944/14175	27/2240
10									989/28350	15741/89600
11										2857/89600
12										
13										
14										
15										
16										
17										
18										
B	=augment({A1+1},coeff(A1+1))									

Ce tableau nous permet d'accéder rapidement aux méthodes d'intégration approchée qui suivent la célèbre méthode de Simpson.

Le cas $n = 4$ par exemple s'appelle la méthode de Villarceau et conduit à la formule :

$$(b-a) \left(\frac{7}{90} f(a) + \frac{16}{45} f\left(a + \frac{b-a}{4}\right) + \frac{2}{15} f\left(a + \frac{b-a}{2}\right) + \frac{16}{45} f\left(a + \frac{3(b-a)}{4}\right) + \frac{7}{90} f(b) \right)$$

Le cas $n = 6$ conduit à la méthode de Hardy.

À partir de $n = 8$, on observe que des signes – apparaissent dans les coefficients, ce qui est une gêne dans le calcul approché, avec le risque possible d'apparition d'erreurs de cancellation (perte importante de chiffres significatifs par différence de nombres très voisins).

Une autre remarque s'impose à l'examen de ce tableau : les coefficients présentent une symétrie, à la manière des coefficients binomiaux :

$B_k^n = B_{n-k}^n$, pour tout entier naturel n non nul et pour tout entier naturel k compris entre 1 et n , ce que nous prouverons un peu plus loin.

1.7 Une autre fonction pour le calcul des coefficients

- Revenons à l'étude générale. En faisant intervenir les coefficients $B_k^n = \frac{A_k^n}{b-a}$, mis en évidence sur la feuille de calcul précédente, la formule de quadrature s'écrit :

$$\int_a^b f(x) dx \approx \sum_{i=0}^n A_i^n f(a+ih) = (b-a) \sum_{i=0}^n B_i^n f(a+ih)$$

en posant $h = \frac{b-a}{n}$.

Comment peuvent s'exprimer plus simplement les coefficients B_i^n ?

$$\text{On sait que } B_i^n = \frac{1}{b-a} A_i^n = \frac{1}{b-a} \int_a^b L_i(x) dx \text{ avec } L_i(x) = \frac{\prod_{\substack{k=0 \\ k \neq i}}^n (x - x_k)}{\prod_{\substack{k=0 \\ k \neq i}}^n (x_i - x_k)}.$$

Calculons cette dernière intégrale par changement de variable en posant $y = \frac{x-a}{h}$ ou $x = a + yh$, si bien que $hdy = dx$.

D'une part :

$$\prod_{\substack{k=0 \\ k \neq i}}^n (x - x_k) = \prod_{\substack{k=0 \\ k \neq i}}^n (a + yh - a - kh) = \prod_{\substack{k=0 \\ k \neq i}}^n (h(y - k)) = h^n \prod_{\substack{k=0 \\ k \neq i}}^n (y - k)$$

... et d'autre part :

$$\begin{aligned} \prod_{\substack{k=0 \\ k \neq i}}^n (x_i - x_k) &= \prod_{\substack{k=0 \\ k \neq i}}^n (a + ih - a - kh) = h^n \prod_{\substack{k=0 \\ k \neq i}}^n (i - k) \\ &= h^n \times i \times (i-1) \times \dots \times (i - (i-1)) \times (i - (i+1)) \times (i - (i+2)) \times \dots \times (i - n) \\ &= h^n \times i! \times (-1) \times (-2) \times \dots \times (-(n-i)) = h^n \times (-1)^{n-i} \times i! \times (n-i)! \end{aligned}$$

si bien que

$$L_i(x) = \frac{\prod_{\substack{k=0 \\ k \neq i}}^n (x - x_k)}{\prod_{\substack{k=0 \\ k \neq i}}^n (x_i - x_k)} = \frac{h^n \prod_{\substack{k=0 \\ k \neq i}}^n (y - k)}{h^n \times (-1)^{n-i} \times i! \times (n-i)!} = \frac{\prod_{\substack{k=0 \\ k \neq i}}^n (y - k)}{(-1)^{n-i} \times i! \times (n-i)!}$$

On peut passer maintenant au changement de variable dans le calcul de l'intégrale pour obtenir l'expression suivante du coefficient B_i^n :

$$B_i^n = \frac{1}{b-a} \times \frac{1}{(-1)^{n-i} i! (n-i)!} \int_0^n \prod_{\substack{k=0 \\ k \neq i}}^n (y - k) h dy = \frac{(-1)^{n-i}}{i! (n-i)! n} \int_0^n \prod_{\substack{k=0 \\ k \neq i}}^n (y - k) dy$$

Cette dernière expression permet de comprendre pourquoi **l'on obtient un nombre réel, indépendant de a et de b .**

D'autre part, elle permet aussi de prouver la remarque, que nous avons faite plus haut, **sur l'égalité $B_i^n = B_{n-i}^n$.**

En effet :

$$B_{n-i}^n = \frac{(-1)^{n-(n-i)}}{(n-i)! i! n} \int_0^n \prod_{k=0, k \neq n-i}^n (y - k) dy = \frac{(-1)^i}{(n-i)! i! n} \int_0^n \prod_{k=0, k \neq i}^n (y - (n-k)) dy$$

que l'on peut transformer en posant $u = n - y$, d'où $du = -dy$. On peut alors écrire :

$$B_{n-i}^n = \frac{(-1)^i}{(n-i)! i! n} \int_n^0 \prod_{k=0, k \neq i}^n (-(u - k)) (-du) = \frac{(-1)^{n+i}}{(n-i)! i! n} \int_0^n \prod_{k=0, k \neq i}^n (u - k) du$$

tandis que $B_i^n = \frac{(-1)^{n-i}}{(n-i)! i! n} \int_0^n \prod_{k=0, k \neq i}^n (u - k) du$.

Ces deux dernières expressions sont bien les mêmes car $(-1)^{n+k} = (-1)^{n-k} \dots$ le quotient des deux vaut 1.

On retrouve évidemment avec cette expression les résultats obtenus précédemment, avec un calcul un peu plus simple.

Ainsi pour la méthode du trapèze où $n = 1$, les calculs conduisent à :

$$B_0^1 = \frac{-1}{0! \times 1! \times 1} \int_0^1 (y-1) dy = - \left[\frac{y^2}{2} - y \right]_0^1 = \frac{1}{2} \quad \text{aussi égal à } B_1^1 \text{ d'après la symétrie des coefficients.}$$

The screenshot shows a TI-Nspire calculator window with the title bar '*Non enregistré'. The main display shows the expression $-\int_0^1 (y-1) dy$ and the result $\frac{1}{2}$ on the right side of the screen.

Avec trois points, et la méthode de Simpson, on peut écrire :

$$B_0^2 = B_2^2 = \frac{1}{0! \times 2! \times 2} \int_0^2 (y-1)(y-2) dy = \frac{1}{4} \int_0^2 (y^2 - 3y + 2) dy$$

$$= \frac{1}{4} \left[\frac{y^3}{3} - \frac{3y^2}{2} + 2y \right]_0^2 = \frac{1}{4} \left(\frac{8}{3} - 6 + 4 \right) = \frac{1}{4} \times \frac{2}{3} = \frac{1}{6}$$

$$B_1^2 = \frac{-1}{1! \times 1! \times 2} \int_0^2 y(y-2) dy = -\frac{1}{2} \int_0^2 (y^2 - 2y) dy$$

$$= -\frac{1}{2} \left[\frac{y^3}{3} - y^2 \right]_0^2 = -\frac{1}{2} \times -\frac{4}{3} = \frac{2}{3}$$

1.1	*Non enregistré	
$\frac{1}{2! \cdot 2} \cdot \int_0^2 ((y-1) \cdot (y-2)) dy$	$\frac{1}{6}$	
$\frac{-1}{1! \cdot 2} \cdot \int_0^2 (y \cdot (y-2)) dy$	$\frac{2}{3}$	

- Automatisons ces calculs par l'écriture d'une nouvelle fonction, plus rapide que la précédente, pour le calcul des coefficients B_i^n .

```

calc_b
Define LibPriv calc_b(n,i)=
Func
Local k,e

$$\prod_{k=0}^n (y-k)$$


$$\frac{y-i}{y-i} \rightarrow e$$

Return  $\frac{(-1)^{n-i}}{i! \cdot (n-i)! \cdot n} \int_0^n (e) dy$ 
EndFunc

```

Remarquons que la variable y est une variable formelle : elle ne doit pas être déclarée dans **Local**, ni subir une affectation, faute de quoi un message d'erreur serait renvoyé.

On retrouve évidemment les résultats précédents, en deux minutes presque instantanés pour $n = 50$ ou douze minutes pour $n = 70$... mais sans tenir compte de la propriété de symétrie des coefficients... S'il n'est pas spectaculaire, le gain est malgré tout bon à prendre !

```

seq(calc_b(50,k),k,0,50)
{
7826703834477197385469928351367488672438750881894451, 340149326215550713758404130454
1949501792461778315086524181832024641248452402013536256, 487375448115444578771631045458
}
seq(calc_b(70,k),k,0,70)
{
461876181997889678318319711467968526923419946147027628497328644875450144241, 3539387
170556987250568425716768498350834758317418680912413228903451059449881427968000, 5942752
}

```

1.8 Calcul d'une intégrale par interpolation

Nous disposons maintenant de tous les éléments pour écrire un *programme*⁵ donnant le calcul approché de n'importe quelle intégrale en utilisant le polynôme d'interpolation du degré que l'on veut et des abscisses régulièrement espacées.

On utilise cette fois l'instruction **Request** pour définir la fonction, les bornes de l'intervalle d'intégration et le degré du polynôme d'interpolation : f est une variable globale, non déclarée dans **Local**⁶, alors que a et b sont des variables locales.

La liste des abscisses des points d'interpolation est alors générée avec une fonction **seq**.

Il reste à construire la liste des coefficients B_i^n en se servant de la propriété de symétrie : **linf** récupère les premiers coefficients et **lsup** les derniers, symétriques des premiers. Attention si n est pair, cette liste possède un nombre impair d'éléments : un coefficient central est à rajouter.

```
nc
Define LibPriv nc()=
Prgm
Local a,b,n,l1,l2,linf,lsup,r
Request "f(x) = " f(x)
Request "a = " a
Request "b = " b
Request "degré du polynôme d'interpolation..." n
seq(a+k*(b-a)/n,k,0,n)→l1
{l1}→linf:{l1}→lsup
For i,0,int((n-1)/2)
  calc_b(n,i)→r
  augment(linf,{r})→linf:augment({r},lsup)→lsup
EndFor
If mod(n,2)=0 Then
  augment(linf,{calc_b(n,n/2)})→linf
EndIf
augment(linf,lsup)→l2
sum(f(l1)·l2·(b-a)→res
Disp res
EndPrgm
```

Les résultats que l'on obtient sont intéressants mais pas si spectaculaires que cela, eu égard au nombre faramineux de calculs effectués : on obtient avec un polynôme de degré 80 une valeur approchée de π avec 43 décimales exactes :

```
nc()
f(x) = 4/(x^2+1)
a = 0
b = 1
degré du polynôme d'interpolation... 80
1472668468466407801977734338689128213287675506532327639795279486555177953883313353776915
468764932583999589048979902774114159993121224656338290391573440345835572021094736599636
Terminé
division_dec(divex2(res,50) "3.14159265358979323846264338327950288419716938037088"
```

⁵ Une fois n'est pas coutume !

⁶ Une valeur par défaut – le contenu de f s'il existe – sera proposée dans la fenêtre de saisie de l'instruction **Request**.

1.9 Évaluation de l'erreur

D'une manière générale, on peut admettre le théorème suivant, qui permet de préciser le comportement asymptotique de l'erreur commise dans le cas d'une méthode de Newton-Cotes.

Théorème

Si le nombre de points d'interpolation est $n + 1$, dans les formules de Newton-Cotes vues précédemment, avec n pair, l'erreur est en $h^{n+3} = \left(\frac{b-a}{n}\right)^{n+3}$ et avec n impair, elle est en $h^{n+2} = \left(\frac{b-a}{n}\right)^{n+2}$.

On a aussi le résultat suivant :

Les formules de Newton-Cotes à $n + 1$ points sont exactes sur l'ensemble des polynômes de degré n lorsque n est impair, et $n + 1$ lorsque n est pair.

... ce qui explique en particulier que la méthode de Simpson soit vraie non seulement pour les polynômes de degré 2 mais aussi pour les polynômes de degré 3.

Bref, il y a donc avantage à utiliser les formules de Newton-Cotes avec n pair (méthode de Simpson par exemple, où l'on interpole la fonction à intégrer par un polynôme de degré 2, dont l'erreur est en h^5 tandis que la méthode qui la suit utilisant un polynôme de degré 3 est aussi en h^5).

Ceci étant, le coefficient qui intervient devant le terme d'erreur peut être grand et dégrader la qualité de la convergence. Ainsi dans notre calcul, nous avons obtenu 43 décimales exactes alors que

$$\left(\frac{1}{70}\right)^{73} \approx 7,2 \times 10^{-154}.$$

1.10 Vers les méthodes composites

L'interpolation de Lagrange, selon la fonction ou la subdivision utilisée, peut être sujette à des effets de bord (phénomènes de Runge). La convergence, quand n grandit, du polynôme interpolateur vers la fonction qu'il interpole, n'a pas forcément lieu. Ce qui est fâcheux quand on cherche à estimer une intégrale ! Bref, outre que les calculs demandés sont très lourds, on peut en plus découvrir quelques surprises.

C'est pourquoi, plutôt que d'augmenter le degré du polynôme d'interpolation, on préfère utiliser des méthodes de quadrature de type *composites*, éventuellement couplées à la méthode de Romberg⁷, qui permettent à moindre frais, d'obtenir assez facilement des résultats précis. On partage l'intervalle

$[a ; b]$ en n petits sous-intervalles de longueur $h = \frac{b-a}{n}$, sur chacun desquels on interpole la fonction par un polynôme de faible degré.

Nous utiliserons dans toute la suite du chapitre les notations suivantes :

$$\begin{array}{ll} x_0 = a = a + 0 \times h & \dots \\ x_1 = a + 1 \times h & x_k = a + k \times h \\ x_2 = a + 2 \times h & \dots \\ \dots & x_n = a + n \times h = a + n \frac{b-a}{n} = b \end{array}$$

⁷ C'est une méthode d'accélération de convergence à partir de la méthode composite des trapèzes.

2. Méthode des rectangles

2.1 Principe de la méthode

Commençons par supposer que la fonction f est continue et croissante sur l'intervalle $[a ; b]$. Remarquons que pour une fonction continue et décroissante sur $[a ; b]$, les résultats qui suivent pourraient être transposés point par point.

On peut alors encadrer f par les deux fonctions *constantes* g_1 et g_2 définies sur $[a ; b]$ par $g_1(x) = f(a)$ et $g_2(x) = f(b)$.

Théorème

Si f est une fonction continue et croissante sur l'intervalle $[a, b]$, alors :

$$(b-a)f(a) \leq I \leq (b-a)f(b).$$

Démonstration

En effet, et c'est presque immédiat, comme f est croissante sur $[a ; b]$, on peut écrire pour tout x de cet intervalle :

$$f(a) \leq f(x) \leq f(b).$$

En intégrant cette inégalité entre a et b , on en déduit l'inégalité cherchée.

L'interprétation graphique est immédiate, si l'on raisonne sur une fonction f positive : son intégrale, qui est l'aire sous la courbe entre a et b , est encadrée par les aires de deux rectangles, d'où le nom de la méthode.

2.2 Méthode composite des rectangles

Ce que nous venons de décrire n'est autre qu'une méthode de quadrature par interpolation, où l'on remplace la fonction à intégrer par un polynôme de degré 0. Évidemment, la précision est grossière et la méthode composite est toute indiquée pour obtenir un résultat plus précis.

Découpons donc l'intervalle $[a ; b]$ en n sous-intervalles de longueur $h = \frac{b-a}{n}$. On obtient le résultat suivant, valable encore pour une fonction croissante sur $[a ; b]$:

Théorème

Si f est une fonction continue et croissante sur l'intervalle $[a, b]$, alors :

$$R'_n = \frac{b-a}{n} \sum_{i=0}^{n-1} f(x_i) \leq I \leq \frac{b-a}{n} \sum_{i=0}^{n-1} f(x_{i+1}) = R''_n.$$

Démonstration

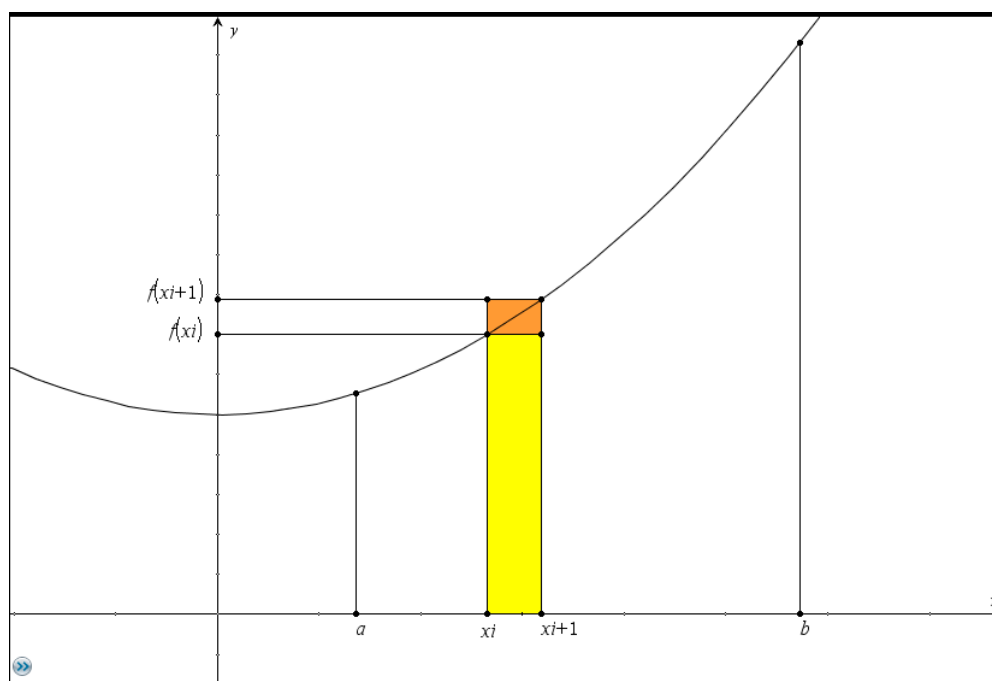
Sur l'intervalle $[x_i, x_{i+1}]$, on a, d'après le théorème du début :

$$(x_{i+1} - x_i) f(x_i) = \frac{b-a}{n} f(x_i) \leq \int_{x_i}^{x_{i+1}} f(x) dx \leq (x_{i+1} - x_i) f(x_{i+1}) = \frac{b-a}{n} f(x_{i+1})$$

Par suite, en sommant depuis $i = 0$, jusqu'à $n - 1$, on peut écrire :

$$\sum_{i=0}^{n-1} \left(\frac{b-a}{n} f(x_i) \right) \leq \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x) dx \leq \sum_{i=0}^{n-1} \left(\frac{b-a}{n} f(x_{i+1}) \right)$$

soit $\frac{b-a}{n} \sum_{i=0}^{n-1} f(x_i) \leq \int_a^b f(x) dx \leq \frac{b-a}{n} \sum_{i=0}^{n-1} f(x_{i+1})$, qui est bien la formule annoncée.



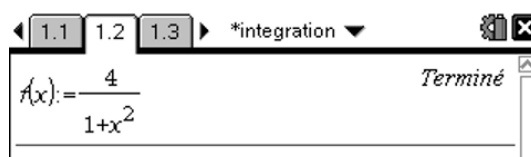
Lorsque n tend vers l'infini, la convergence vers I est assurée... d'après la définition même de l'intégrale de Riemann. En effet, chacune des sommes R'_n et R''_n n'est autre qu'une somme de Riemann, correspondant à la subdivision considérée.

Lorsque la fonction est continue et décroissante sur $[a ; b]$, les inégalités sont inversées⁸ : dans le cas où la fonction est monotone, la méthode des rectangles présente donc le (gros) avantage de donner un encadrement de I , ce qui permet facilement de donner un sens aux résultats renvoyés.

Cas où la fonction n'est pas monotone sur l'intervalle $[a ; b]$: les encadrements précédents n'ont plus lieu, mais la convergence de (R'_n) et de (R''_n) vers I est assurée !

2.3 Une fonction pour la méthode des rectangles

La programmation de cette méthode est très simple. On fait le choix de définir au préalable la fonction à étudier dans une fenêtre **Calculs** :



Une stratégie possible, parmi quelques autres, consiste à mémoriser au préalable dans une variable s la somme de tous les $f(x_i)$ pour i allant de 0 jusqu'à n . Pour le calcul de R'_n , il faut retirer $f(b) = f(x_n)$ de cette variable s tandis que pour celui de R''_n , il faut retirer $f(a) = f(x_0)$... en n'oubliant pas de multiplier par h , le pas de la subdivision.

⁸ Lorsque la fonction est croissante sur $[a, b]$, l'aire des rectangles à gauche minore l'intégrale tandis que l'aire des rectangles à droite la majore. C'est exactement l'inverse lorsque la fonction est décroissante.

Le résultat est par exemple renvoyé dans une liste : on obtient ainsi deux valeurs qui encadrent l'intégrale cherchée dans le cas où la fonction f est monotone.

Le code est donné ci-après :

```

rect
Define rect(a,b,n)=
Func
Local h,s,i
 $\frac{b-a}{n} \rightarrow h:0 \rightarrow s$ 
For i,0,n
   $s+f(a+i \cdot h) \rightarrow s$ 
EndFor
Return {  $h \cdot (s-f(b))$ ,  $h \cdot (s-f(a))$  }
EndFunc

```

Voici quelques-uns des résultats obtenus avec notre fonction. Les deux nombres de la liste encadrent π , comme prévu puisque la fonction f de notre exemple est décroissante sur $[0 ; 1]$. Les rectangles à gauche donnent un majorant de I et les rectangles à droite un minorant.

$f(x)$	$\frac{4}{x^2+1}$
$rect(0.,1,10)$	{ 3.23992598891, 3.03992598891 }
$rect(0.,1,100)$	{ 3.15157598692, 3.13157598692 }
$rect(0.,1,1000)$	{ 3.14259248692, 3.14059248692 }
$rect(0.,1,10000)$	{ 3.14169265192, 3.14149265192 }
$rect(0.,1,100000)$	{ 3.14160265357, 3.14158265357 }

Le découpage de l'intervalle en 10 000 ou 100 000 tranches allonge fortement le temps de calcul, notamment sur la calculatrice, mais ne conduit pas à un gain de précision spectaculaire, d'autant aussi qu'avec 100 000 itérations risque de s'accroître le cumul des erreurs d'arrondi ! La méthode n'est donc pas recommandée pour obtenir une valeur approchée de π très précise⁹.

Revenons sur le problème important du contrôle de l'erreur. Supposons pour fixer les idées que l'on choisisse de renvoyer comme résultat la valeur moyenne $\bar{R}_n = \frac{R'_n + R''_n}{2}$, comme le montre l'instruction suivante :

```

mean(rect(0.,1,100000))
3.1415926535694

```

Toujours en supposant que la fonction f est monotone sur l'intervalle $[a ; b]$, on peut majorer l'erreur commise :

$$|I - \bar{R}_n| \leq \frac{|R''_n - R'_n|}{2} = \frac{b-a}{2n} |f(b) - f(a)|.$$

Ainsi pour avoir un résultat $\bar{R}_n$ à 10^{-6} près avec notre exemple de référence, on devrait prendre :

$$\frac{b-a}{n} |f(b) - f(a)| = \frac{1-0}{n} (f(0) - f(1)) = \frac{2}{n} \leq 10^{-6}$$

soit $n \geq 2 \times 10^6 = 2\,000\,000 \dots$

⁹ On remarquera aussi qu'on a indiqué comme entrée 0. pour forcer le calcul en mode approché. Un calcul exact, étant donnée la précision de la méthode, ne présente aucun intérêt.

... c'est beaucoup, voire inaccessible pour une calculatrice comme la nôtre ! Ou pour tout outil de calcul approché d'ailleurs qui risque d'accumuler au cours des 2 000 000 de calculs les erreurs d'arrondis...

2.4 Illustration graphique

Il nous reste à étudier la possibilité d'une illustration graphique de la méthode des rectangles, pédagogiquement indispensable ! On sait qu'on ne dispose pas véritablement d'instructions de programmation graphique avec TI-Nspire, pour le moment du moins. Mais comme on l'a déjà fait précédemment, on peut s'en sortir avec des nuages de points, en mode relié, judicieusement choisis pour décrire les rectangles nécessaires à chaque étape.

De quelles pages aurons-nous besoin dans le classeur ?

Tout d'abord une page **Calculs**, dans laquelle nous définissons comme d'habitude la fonction f ;

une page **Tableur & Listes**, qui servira à piloter de façon dynamique la page de géométrie ;

enfin une page **Graphiques**... pour visualiser le résultat obtenu, et piloter le découpage de l'intervalle à l'aide d'un curseur... D'ores et déjà, on peut demander dans cette page la représentation graphique de f , sans se préoccuper pour le moment du réglage de la fenêtre que l'on gèrera par programme. De plus on stocke les extrémités des axes, en les faisant apparaître si nécessaire, dans les variables **xmin**, **xmax**, **ymin**, **ymax**.

... et des fenêtres de programmation, autant que nécessaire !

Préparons d'abord le terrain... Une fois la fonction définie, on peut demander par programme un réglage automatique de la fenêtre... autant se décharger le plus possible des tâches fastidieuses !

C'est l'objet du programme qui suit, programme et non fonction car il faut modifier des variables *globales* –ici **xmax**, **xmin**, **ymax** et **ymin**– ce qu'une fonction ne peut pas faire. Les paramètres d'entrée sont **a** et **b**, les bornes de l'intervalle d'étude.

Attention à bien faire figurer l'axe des abscisses dans la fenêtre puisqu'il est nécessaire aux constructions à venir : si le maximum de la fonction est strictement négatif, il faut le remplacer par 0 et si le minimum est strictement positif, il faut aussi le remplacer par 0.

```
fenetre_f
Define LibPriv fenetre_f(a,b)=
Prgm
Local m1,m2,i
© réglage de la fenêtre
a-0.1*(b-a)→xmin
b+0.1*(b-a)→xmax
fnc\maximum(f(x),a,b)→m2:fnc\minimum(f(x),a,b)→m1
max(m2,0)→m2:min(m1,0)→m1
m1-(m2-m1)/10→ymin:m2+(m2-m1)/10→ymax
EndPrgm
```

Demandons l'exécution de ce programme. On obtient bien un cadrage correct – horizontalement et verticalement – de notre représentation graphique à partir de l'intervalle $[a, b]$ choisi. Enfin, pour simplifier cette phase de préparation, on peut regrouper toutes les instructions et le programme précédent sur une seule ligne, comme le montre l'écran ci-dessous :

```
f(x)=4/(1+x^2):a:=0:b:=1:fenetre_f(a,b) Terminé
```

Poursuivons notre travail, en écrivant trois fonctions, construites sur le même modèle, qui retourneront les listes des abscisses et ordonnées des nuages de points correspondant à la méthode des rectangles à gauche et à droite¹⁰.

Des fonctions cette fois... car elles devront être utilisées dans le tableur pour réagir dynamiquement.

Décrivons la construction attendue. À chaque passage dans la boucle, il faudrait créer le rectangle à gauche dont les sommets ont pour coordonnées :

$$(a+i \times h ; 0), (a+i \times h ; f(a+i \times h)), (a+(i+1) \times h ; f(a+i \times h)) \text{ et } (a+(i+1) \times h, 0).$$

On peut donc écrire deux fonctions, la première qui donnera la liste de toutes les abscisses en ajoutant à chaque étape les valeurs

$$a+i \times h, a+i \times h, a+(i+1) \times h \text{ et } a+(i+1) \times h,$$

la seconde qui donnera la liste des ordonnées avec les valeurs

$$0, f(a+i \times h), f(a+i \times h) \text{ et } 0.$$

Ci-dessous le texte, très simple, de ces deux fonctions (**x_rectgau** et **y_rectgau**). Il s'agit simplement de constituer une liste d'abscisses suivant les règles ci-dessus et les ordonnées qui correspondent. De plus, il n'est pas nécessaire de réécrire intégralement la deuxième fonction : on repart d'une copie de la première, *mutatis mutandis*.

x_rectgau	5/7	y_rectgau	5/7
Define LibPriv x_rectgau (a,b,n)=		Define LibPriv y_rectgau (a,b,n)=	
Func		Func	
Local h,l,i		Local h,l,i	
$\frac{b-a}{n} \rightarrow h$		$\frac{b-a}{n} \rightarrow h$	
{ } → l		{ } → l	
For i,0,n-1		For i,0,n-1	
augment(l,{a+i·h,a+i·h,a+(i+1)·h,a+(i+1)·h})→l		augment(l,{0,f(a+i·h),f(a+i·h),0})→l	
EndFor		EndFor	
Return l		Return l	
EndFunc		EndFunc	

Idem avec les rectangles à droite : les abscisses ne changeant pas –inutile de réécrire une fonction pour la liste des abscisses–, il faut juste modifier la liste des ordonnées en conséquence. Les points à joindre ont cette fois pour coordonnées :

$$(a+i \times h ; 0), (a+i \times h ; f(a+(i+1) \times h)), (a+(i+1) \times h ; f(a+(i+1) \times h)) \text{ et } (a+(i+1) \times h ; 0).$$

D'où la fonction **y_rectdroi** ci-après.

y_rectdroi	5/7
Define LibPriv y_rectdroi (a,b,n)=	
Func	
Local h,l,i	
$\frac{b-a}{n} \rightarrow h$	
{ } → l	
For i,0,n-1	
augment(l,{0,f(a+(i+1)·h),f(a+(i+1)·h),0})→l	
EndFor	
Return l	
EndFunc	

Pour fixer les idées, dans la page **Calculs**, on obtient par exemple les résultats suivants :

¹⁰ Trois fonctions car la liste des abscisses sert à la fois pour les rectangles à gauche et à droite...

$f(x) = \frac{4}{1+x^2} : a:=0 : b:=1 : fenetre_f(a,b)$	Terminé
$x_rectgau(a,b,3)$	$\left\{ 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3}, \frac{2}{3}, \frac{2}{3}, 1, 1 \right\}$
$y_rectgau(a,b,3)$	$\left\{ 0, 4, 4, 0, 0, \frac{18}{5}, \frac{18}{5}, 0, 0, \frac{36}{13}, \frac{36}{13}, 0 \right\}$
$y_rectdroi(a,b,3)$	$\left\{ 0, \frac{18}{5}, \frac{18}{5}, 0, 0, \frac{36}{13}, \frac{36}{13}, 0, 0, 2, 2, 0 \right\}$

Mais c'est dans le tableur que nous allons utiliser ces listes... sur le modèle de la page qui suit. Rien à ajouter à ce que l'on voit, sauf rappeler qu'en A1 est définie la variable n , et que les cellules B1 et B2 sont liées aux variables a et b , déjà définies dans la page **Calculs**. Ne pas oublier non plus de nommer les listes obtenues, **xx1**, **yy1** et **yy2** comme indiqué ci-après. Ces noms vont permettre de demander le tracé des nuages de points, c'est-à-dire, on le sait, des rectangles à gauche et à droite. On peut aussi dans la cellule G1 demander l'affichage du calcul approché de l'intégrale avec les paramètres choisis : il suffit d'appeler la fonction **rect(a,b,n)** et de demander la moyenne de la liste obtenue. Nous stockons cette valeur dans la variable **res**.

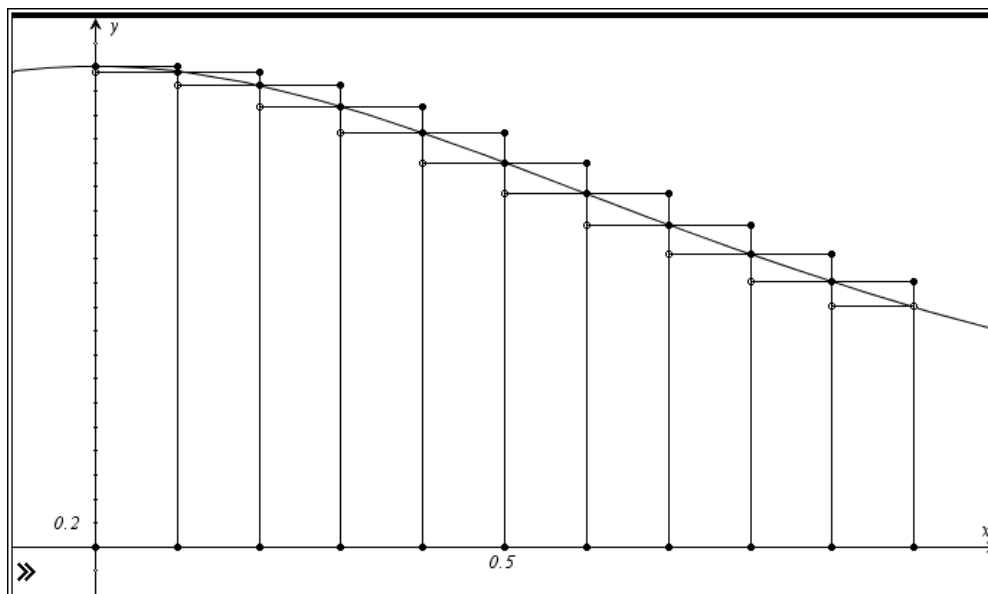
	A	B	C _{xx1}	D _{yy1}	E _{yy2}	F	G	H
			=x_rectgau('a','b','n')	=y_rectgau('a','b','n')	=y_rectdroi('a','b','n')			
1	10.	0	0.	0	0		3.13992598891	
2		1	0.	4.	3.9603960396			
3			0.1	4.	3.9603960396			
4			0.1	0	0			
5			0.1	0	0			
6			0.1	3.9603960396	3.84615384615			
7			0.2	3.9603960396	3.84615384615			
8			0.2	0	0			
9			0.2	0	0			
10			0.2	3.84615384615	3.66972477064			
11			0.3	3.84615384615	3.66972477064			
12			0.3	0	0			
13			0.3	0	0			
14			0.3	3.66972477064	3.44827586207			
15			0.4	3.66972477064	3.44827586207			
16			0.4	0	0			
17			0.4	0	0			
18			0.4	3.44827586207	3.2			
G1	res:=mean(rect(a,b,n))							

Dans la fenêtre **Graphiques**, il reste à représenter deux nuages de points :

l'un avec pour abscisses les éléments de la liste **xx1** et pour ordonnées ceux de la liste **yy1** (en mode relié –voir dans Attributs–, ce nuage visualisera les rectangles à gauche) ;

l'autre avec pour abscisses les éléments de la liste **xx1** et pour ordonnées ceux de la liste **yy2**, qui représentera les rectangles à droite.

On obtient alors le résultat suivant, qui donne une illustration intéressante de la méthode des rectangles. Il reste à rendre tout ceci plus convivial pour que l'utilisation de ce graphique soit la plus simple possible.

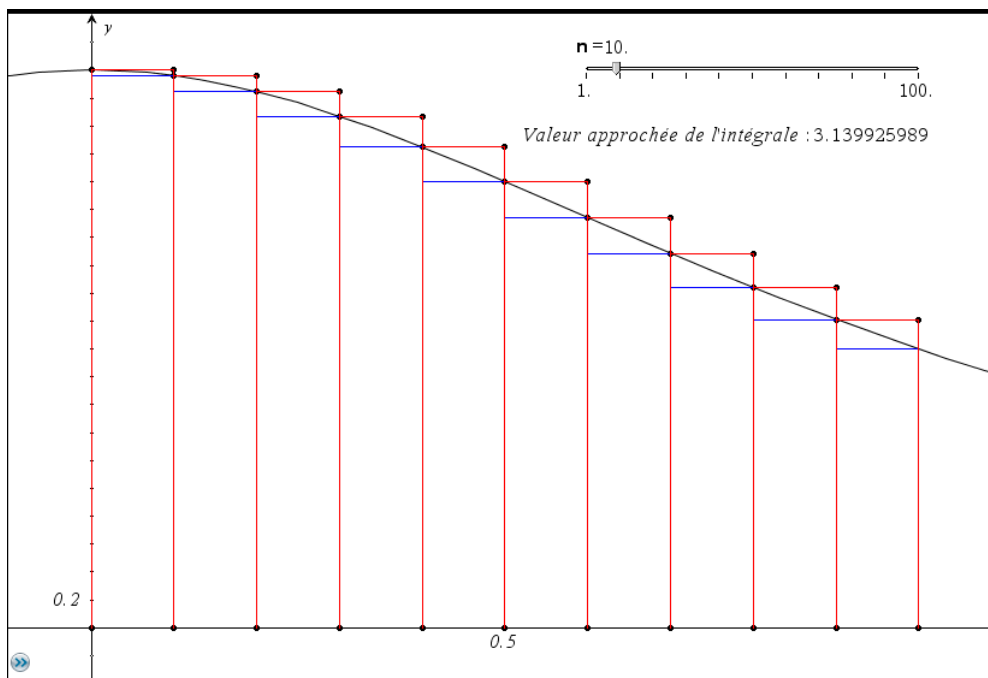


Quelles améliorations peuvent-elles être apportées ?

Tout d'abord, on peut *piloter* la valeur de n au moyen d'un curseur, par pas entier de 1 à 100. Une modification de la valeur de n relancera le re-calculation¹¹, donc en particulier l'appel des fonctions qui fabriquent les nouvelles listes. La fenêtre de géométrie est ainsi mise à jour de façon dynamique.

Ensuite, on peut utiliser la couleur (sur le logiciel) pour distinguer les rectangles à droite et à gauche.

Enfin on peut demander l'affichage de la valeur approchée de l'intégrale correspondant aux paramètres choisis en demandant l'affichage de la variable **res** dans l'application **Graphiques**... Un clic droit sur la valeur renvoyée permet d'accéder au menu Attributs... pour demander quelques décimales supplémentaires. On peut ancrer cette valeur au texte : *Valeur approchée de l'intégrale*.



¹¹ C'est là où le caractère « dynamique » du tableur est appréciable !

Ce modèle de classeur pourra être réutilisé pour d'autres méthodes de calcul approché d'intégrales, notamment la méthode des trapèzes. Après un « enregistrer sous », il ne restera plus qu'à adapter à la méthode étudiée, par quelques modifications somme toute mineures, la construction qui vient d'être faite. La méthode de Simpson toutefois relèvera d'un traitement un peu différent pour l'illustration graphique.

2.5 Méthode des rectangles avec une fonction de classe C^1

- On ne suppose plus que la fonction f est monotone sur $[a ; b]$, mais simplement de classe C^1 . On a vu que l'on perdait alors l'encadrement du paragraphe précédent. On peut se contenter de ne travailler qu'avec les rectangles à droite ou à gauche : on remplace donc la fonction f par la fonction constante égale à $f(a)$ (ou $f(b)$) sur l'intervalle $[a ; b]$.

Un calcul d'erreur peut cependant être mené, en posant $M_1 = \sup_{x \in [a, b]} |f'(x)|$ et en utilisant l'inégalité des accroissements finis. En effet, pour tout x appartenant à $[a ; b]$, on a :

$$|f(x) - f(a)| \leq M_1 |x - a|.$$

Par suite :

$$\begin{aligned} \left| \int_a^b f(x) dx - (b-a) f(a) \right| &= \left| \int_a^b (f(x) - f(a)) dx \right| \leq \int_a^b |f(x) - f(a)| dx \\ &\leq \int_a^b M_1 |x - a| dx = M_1 \int_a^b (x - a) dx = M_1 \left[\frac{x^2}{2} - ax \right]_a^b \\ &= M_1 \left(\frac{b^2}{2} - ab - \frac{a^2}{2} + a^2 \right) = \frac{M_1}{2} (b-a)^2 \end{aligned}$$

On a bien sûr un résultat analogue en travaillant avec la valeur $f(b)$.

Cette formule, avec un coefficient M_1 lié à la dérivée première, rappelle aussi le fait que la méthode des rectangles est d'ordre 0, autrement dit qu'elle est exacte pour les polynômes de degré 0... En effet, dans ce cas, la dérivée première précisément est nulle sur $[a ; b]$... ainsi que l'erreur commise...

- Pour la méthode composite des rectangles, l'inégalité précédente appliquée à un intervalle $[x_i ; x_{i+1}]$ de la subdivision donne :

$$\left| \int_{x_i}^{x_{i+1}} f(x) dx - \frac{b-a}{n} f(x_i) \right| \leq \frac{M_1}{2} (x_{i+1} - x_i)^2 = \frac{M_1}{2} \left(\frac{b-a}{n} \right)^2$$

En sommant sur tous les intervalles, on a :

$$\left| \int_a^b f(x) dx - \frac{b-a}{n} \sum_{i=0}^{n-1} f(x_i) \right| \leq \frac{M_1}{2n} (b-a)^2.$$

2.6 Vers une amélioration de la méthode des rectangles¹²

- Restons dans les hypothèses de la fin du paragraphe précédent... car rien ne nous oblige à remplacer f sur l'intervalle $[x_i ; x_{i+1}]$ par une fonction constante égale à $f(x_i)$ ou à $f(x_{i+1})$. On peut prendre n'importe quel $f(c_i)$, où c_i est une valeur de $[x_i ; x_{i+1}]$.

Peut-être existe-t-il un choix plus pertinent avec une erreur moins importante ?

¹² Le classeur proposé ici repose sur une idée originale de Philippe Fortin. Merci à lui !

Le classeur précédent peut être adapté pour que l'on puisse observer ce qui se passe quand la valeur de c_i varie. La variable k pilote ce changement : $k = 0$ correspond à la méthode des rectangles à gauche, soit $c_i = x_i$, et $k = 1$ à la méthode des rectangles à droite, soit $c_i = x_{i+1}$; les autres valeurs permettent d'obtenir un c_i strictement compris entre x_i et x_{i+1} . Les principales modifications sont données ci-après.

Le paramètre k apparaît maintenant dans la fonction **rect** :

```

rect
Define rect(a,b,n,k)=
Func
Local h,s,i
 $\frac{b-a}{n} \rightarrow h; 0 \rightarrow s$ 
For i,0,n-1
 $s + f(a + (i+k) \cdot h) \rightarrow s$ 
EndFor
Return h · s
EndFunc

```

... mais aussi dans la fonction **y_rectgau**, tandis que **x_rectgau** n'a pas à être modifié :

"x_rectgau" enregistrement effectué	y_rectgau
<pre> Define LibPriv x_rectgau(a,b,n)= Func Local h,l,i $\frac{b-a}{n} \rightarrow h$ { } → l For i,0,n-1 augment(l, { $a+i \cdot h, a+i \cdot h, a+(i+1) \cdot h, a+(i+1) \cdot h$ }) → l EndFor Return l EndFunc </pre>	<pre> Define LibPriv y_rectgau(a,b,n,k)= Func Local h,l,i $\frac{b-a}{n} \rightarrow h$ { } → l For i,0,n-1 augment(l, { 0, $f(a+(i+k) \cdot h), f(a+(i+k) \cdot h), 0$ }) → l EndFor Return l EndFunc </pre>

Il suffit en fait de modifier tous les $f(a + ih)$ en $f(a + (i + k)h)$

La feuille de calcul est modifiée en conséquence : plus besoin de gérer les rectangles à droite, on peut supprimer la colonne.

On peut aussi ajouter :

la cellule B4 que l'on met à 0, pour commencer, et que l'on mémorise dans la variable k ,

ce qui permet de modifier les appels des fonctions **y_rectgau** et **rect**, en incluant cette variable k ,

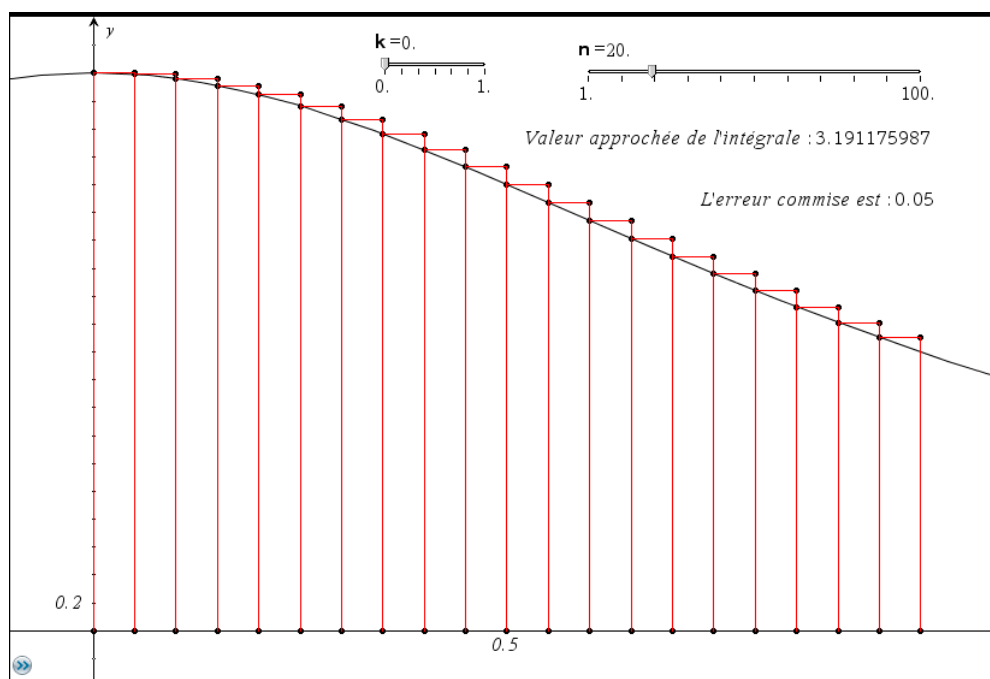
le calcul de l'erreur en E4, stocké dans une variable **erreur**,

la capture des variables k , d'une part, et **erreur**, d'autre part, dans les colonnes H et I, nommées respectivement **hh** et **ii**.

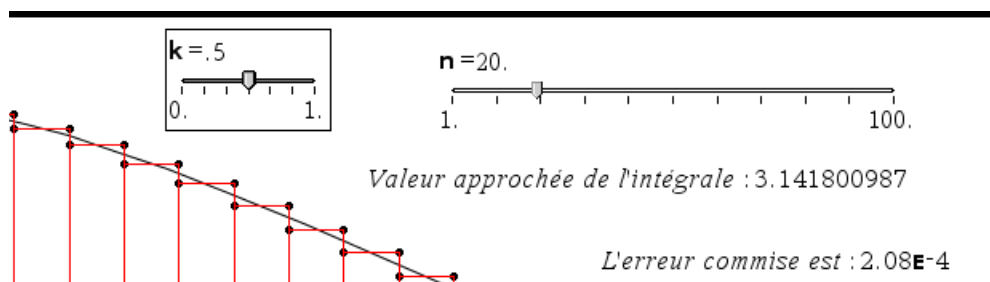
A	B	Cxx1	Dyy1	Eyy2	F	G	Hhh	Iii	J
		=x_rect	=y_rect	=y_rectdroi			=capture('k,1)	=capture(erreur,1)	
1	20.	0	0.	0		3.18922694789	0.02	0.0476342943	
2		1	0.3999...	3.9900249...		erreur			
3			0.05	3.999...		0.0476342943			
4	0.02		0.05	0					
5			0.05	0					
6			0.05	3.989...		3.9603960...			
7			0.1	3.989...		3.9603960...			
8			0.1	0					
9			0.1	0					
10			0.1	3.959...		3.9119804...			
11			0.15	3.959...		3.9119804...			
12			0.15	0					
13			0.15	0					
14			0.15	3.910...		3.8461538...			
15			0.2	3.910...		3.8461538...			
16			0.2	0					

G3 erreur: $\int_a^b (f(x)) dx$

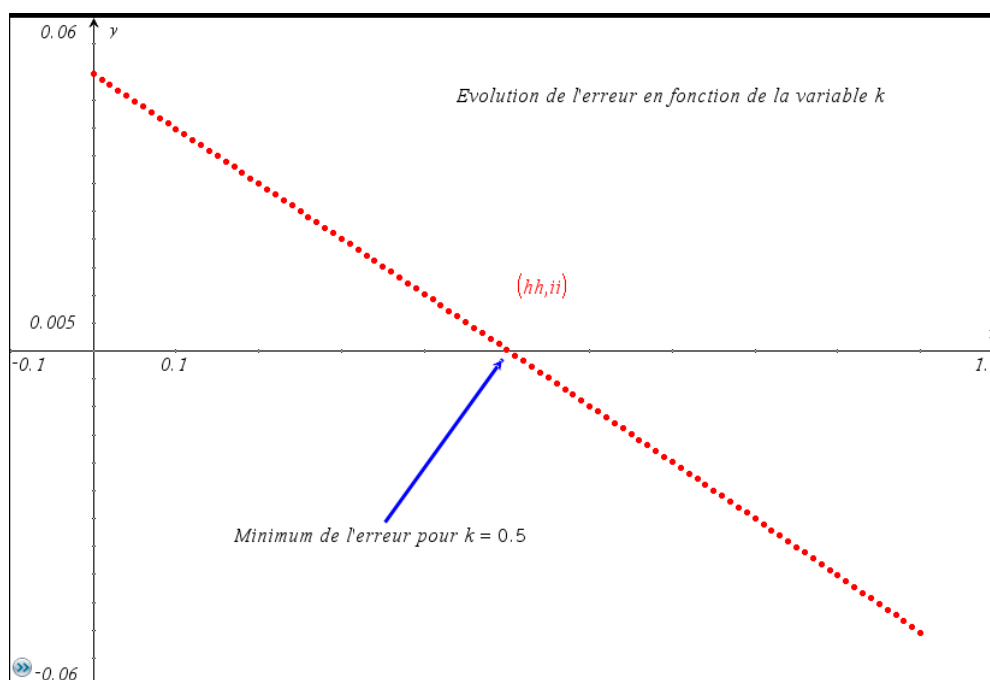
Enfin un curseur pour la variable k est ajouté à la page **Graphiques** (pas de 0,01), ainsi que l'affichage de la variable **erreur**.



Prenons par exemple $n = 20$ et faisons varier les valeurs de k depuis 0 jusqu'à 1 en animant le curseur. On constate que l'erreur présente un minimum lorsque l'on choisit $k = 0,5$ (voir les rectangles...).



La représentation graphique du nuage de points (en abscisse les valeurs de k , en ordonnée les valeurs de l'erreur qui correspond) est aussi très explicite :



- Reprenons le calcul d'erreur fait plus haut, pour la méthode des rectangles appliquées à un intervalle $[a ; b]$ tout entier, en prenant un réel $c \in [a ; b]$ plutôt qu'une des bornes de l'intervalle. On peut alors écrire :

$$\left| \int_a^b f(x) dx - (b-a) f(c) \right| = \left| \int_a^b (f(x) - f(c)) dx \right| \leq \int_a^b |f(x) - f(c)| dx$$

Comme précédemment, on est amené à utiliser l'inégalité des accroissements finis. En posant $M_1 = \sup_{x \in [a, b]} |f'(x)|$, on arrive à :

$$|f(x) - f(c)| \leq M_1 |x - c| \text{ pour tout } x \text{ de } [a ; b].$$

Par conséquent :

$$\left| \int_a^b f(x) dx - (b-a) f(c) \right| \leq \int_a^b |f(x) - f(c)| dx \leq M_1 \int_a^b |x - c| dx.$$

Bref, tout repose sur le calcul de l'intégrale $\int_a^b |x - c| dx$:

$$\int_a^b |x - c| dx = \int_a^c |x - c| dx + \int_c^b |x - c| dx = \int_a^c (c - x) dx + \int_c^b (x - c) dx$$

et pour s'épargner des calculs sans intérêt, la TI-Nspire peut prendre le relais :

$\int_a^c (c-x) dx + \int_c^b (x-c) dx$	$\frac{a^2}{2} - a \cdot c + \frac{b^2}{2} - b \cdot c + c^2$
$\text{expand}\left(\frac{a^2}{2} - a \cdot c + \frac{b^2}{2} - b \cdot c + c^2, c\right)$	$c^2 - c \cdot (a+b) + \frac{a^2 + b^2}{2}$

On arrive finalement à :

$$\left| \int_a^b f(x) dx - (b-a) f(c) \right| \leq M_1 \left(c^2 - c(a+b) + \frac{a^2 + b^2}{2} \right).$$

c peut prendre n'importe quelle valeur entre a et b . On retrouve notamment avec $c = a$ l'erreur calculée précédemment, qui est la même que pour $c = b$.

$c^2 - c \cdot (a+b) + \frac{a^2+b^2}{2} \rightarrow e(c)$	Terminé
$\text{factor}(e(a))$	$\frac{(a-b)^2}{2}$
$\text{factor}(e(b))$	$\frac{(a-b)^2}{2}$

Cette erreur étant une fonction du second degré de la variable c , on sait qu'elle présente un minimum pour $c = -\frac{-(a+b)}{2 \times 1} = \frac{a+b}{2}$... c'est-à-dire pour le point médian... Ce minimum vaut alors :

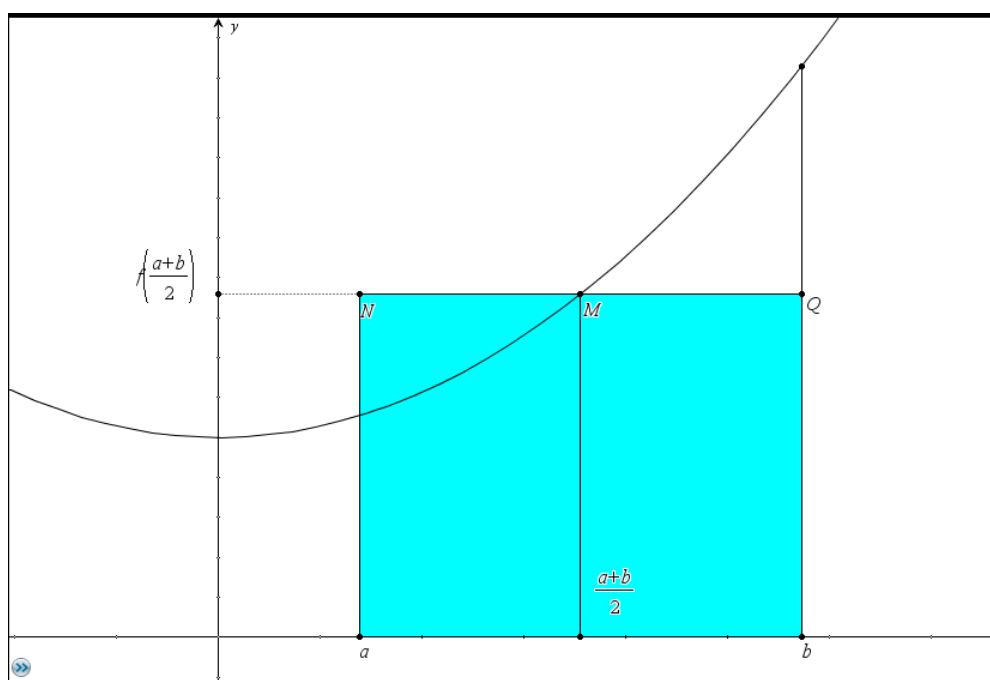
$$\left| \text{factor}\left(e\left(\frac{a+b}{2}\right)\right) \right| = \frac{(a-b)^2}{4}$$

c'est-à-dire une erreur deux fois moindre que la précédente¹³. C'est dire l'intérêt qu'il y a à choisir ce point médian plutôt qu'un autre.

3. La méthode des rectangles avec point médian

3.1 Principe de la méthode

Les remarques précédentes nous conduisent à la méthode des rectangles *avec point médian* : on remplace la fonction f sur l'intervalle $[a ; b]$ par la fonction constante g égale à la valeur de f au point médian $\frac{a+b}{2}$. Autrement dit pour tout x de $[a ; b]$, on a $g(x) = f\left(\frac{a+b}{2}\right)$. Comme le montre le dessin suivant dans le cas où la fonction est positive, on remplace encore une fois l'intégrale cherchée par le calcul de l'aire d'un rectangle.



¹³ En fait beaucoup moins lorsque la fonction est de classe C^2 , comme le montrera la suite.

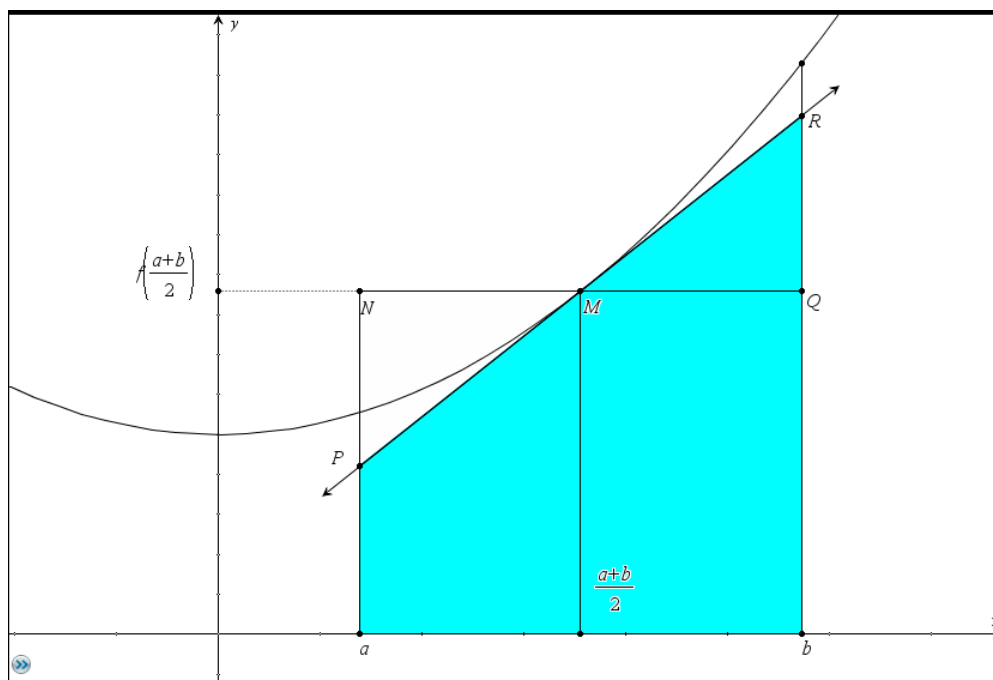
Soit f une fonction définie et continue sur un intervalle $[a, b]$.

On approxime alors I par $R = (b-a) f\left(\frac{a+b}{2}\right)$.

On conçoit que l'approximation obtenue ici soit un peu meilleure en général, par une certaine compensation des aires, qu'avec le rectangle à gauche, qui minorerait dans notre exemple l'intégrale et le rectangle à droite qui la majorerait.

Donnons quelques précisions supplémentaires dans le cas, fréquent, où la fonction f est dérivable en $c = \frac{a+b}{2}$. L'aire du rectangle obtenu avec le point médian est alors exactement la même que celle du

trapèze ci-dessous, obtenu à partir de la tangente à la courbe au point d'abscisse $c = \frac{a+b}{2}$.



C'est immédiat si l'on a affaire à une fonction positive sur $[a; b]$: les triangles MNP et MQR sont isométriques et ont donc la même aire ; à la façon d'un puzzle, on passe du trapèze au rectangle en mettant le triangle MQR dans le triangle MNP .

Un raisonnement général s'applique aussi. En effet, considérons la droite de pente m^{14} passant par le point de coordonnées $\left(\frac{a+b}{2}, f\left(\frac{a+b}{2}\right)\right)$. Cette droite a pour équation $y = m\left(x - \frac{a+b}{2}\right) + f\left(\frac{a+b}{2}\right)$.

L'aire (algébrique) du trapèze limité par cette droite, l'axe des abscisses et les droites d'équation $x = a$ et $x = b$ est :

$$\begin{aligned} \int_a^b \left(m\left(x - \frac{a+b}{2}\right) + f\left(\frac{a+b}{2}\right) \right) dx &= \int_a^b m\left(x - \frac{a+b}{2}\right) dx + (b-a) f\left(\frac{a+b}{2}\right) \\ &= \int_{-(b-a)/2}^{(b-a)/2} mu \, du + (b-a) f\left(\frac{a+b}{2}\right) = (b-a) f\left(\frac{a+b}{2}\right) \end{aligned}$$

¹⁴ m est un nombre réel quelconque, pas forcément égal au nombre dérivé de la fonction au point d'abscisse $(a+b)/2$.

c'est-à-dire précisément la valeur de la méthode des rectangles avec point médian.

Ce que montre aussi la TI-Nspire :

Remarquons que lorsque la fonction est dérivable et convexe sur $[a ; b]$, la courbe représentative de f est située au dessus de sa tangente au point d'abscisse $\frac{a+b}{2}$: par conséquent, l'approximation R par la méthode des rectangles avec point médian est inférieure ou égale à la valeur I de l'intégrale. C'est le contraire si la fonction est concave.

3.2 Évaluation de l'erreur

Le calcul que nous avons fait précédemment reste valable dans le cas où la fonction est de classe C^1 sur $[a ; b]$. On peut aller plus loin si la fonction est de classe C^2 . C'est l'objet du théorème qui suit :

Théorème

Si la fonction f est de classe C^2 sur $[a ; b]$, et si M_2 désigne le maximum de $|f''|$ sur $[a ; b]$, alors $|I - R| \leq \frac{(b-a)^3}{24} M_2$.

Cette majoration qui fait intervenir la dérivée seconde montre que la méthode des rectangles avec point médian est donc d'ordre 1 : elle est exacte pour toutes les polynômes de degré 1 – les fonctions affines – dont la dérivée seconde est nulle !

Démonstration

Reprenons le calcul d'erreur fait précédemment en posant $c = \frac{a+b}{2}$. Nous avons obtenu :

$$I - R = \int_a^b f(x) dx - (b-a) f(c) = \int_a^b (f(x) - f(c)) dx.$$

et nous nous étions appuyés sur l'inégalité des accroissements finis. On peut aller plus loin car la fonction f est de classe C_2 ... par exemple avec la formule de Taylor-Lagrange appliquée à l'intervalle d'extrémités x et c :

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

où d appartient à $]c ; x[$ ou $]x ; c[$.

Si bien que :

$$I - R = \int_a^b (x-c) f'(c) dx + \frac{1}{2} \int_a^b (x-c)^2 f''(d) dx.$$

Remarquons que le premier terme de cette somme est nul, précisément par le choix judicieux du point médian... En effet :

$$\int_a^b (t-c) f'(c) dt = f'(c) \left[\frac{(t-c)^2}{2} \right]_a^b = 0$$

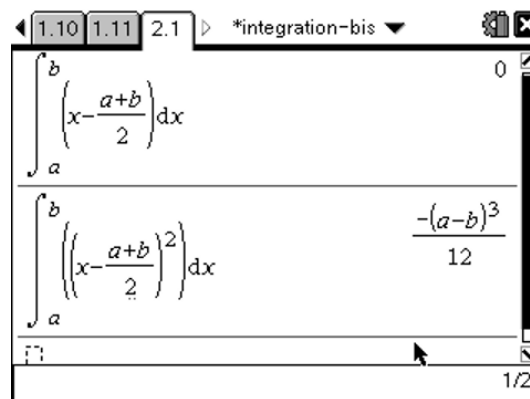
Par suite :

$$|I - R| \leq \frac{1}{2} \left| \int_a^b (x-c)^2 f''(d) dx \right| \leq \frac{M_2}{2} \int_a^b (x-c)^2 dx = \frac{M_2}{2} \left[\frac{(x-c)^3}{3} \right]_a^b.$$

Reste à finir le dernier calcul...

$$\begin{aligned} \left[\frac{(x-c)^3}{3} \right]_a^b &= \frac{\left(b - \frac{a+b}{2}\right)^3}{3} - \frac{\left(a - \frac{a+b}{2}\right)^3}{3} = \frac{\left(\frac{b-a}{2}\right)^3}{3} - \frac{\left(\frac{a-b}{2}\right)^3}{3} \\ &= \frac{\left((b-a)^3 - (a-b)^3\right)}{24} = \frac{2(b-a)^3}{24} = \frac{(b-a)^3}{12} \end{aligned}$$

Remarquons que ces résultats sont donnés par TI-Nspire :



En conclusion, on a bien prouvé que :

$$|I - R| \leq \frac{M_2 (b-a)^3}{24}.$$

3.3 Méthode composite des rectangles avec point médian

Théorème

Si f est une fonction de classe C_2 sur l'intervalle $[a ; b]$ et si M_2 désigne le maximum de $|f''|$ sur $[a ; b]$, alors on peut approximer I par $R_n = \frac{b-a}{n} \sum_{k=0}^{n-1} f\left(\frac{x_k + x_{k+1}}{2}\right)$.

L'erreur commise est majorée par $\frac{(b-a)^3}{24n^2} M_2$.

Démonstration

La convergence de la suite (R_n) vers I est immédiate : c'est la convergence d'une somme de Riemann vers l'intégrale qu'elle définit.

Appliquons le résultat précédent pour chacun des intervalles $[x_k ; x_{k+1}]$, pour k variant entre 0 et $n-1$. On obtient :

$$\left| \int_{x_k}^{x_{k+1}} f(x) dx - (x_{k+1} - x_k) f\left(\frac{x_k + x_{k+1}}{2}\right) \right| \leq \frac{(x_{k+1} - x_k)^3}{24} M_2$$

qui peut s'écrire :

$$-\frac{(b-a)^3}{24n^3}M_2 \leq \int_{x_k}^{x_{k+1}} f(x)dx - \frac{b-a}{n}f\left(\frac{x_k+x_{k+1}}{2}\right) \leq \frac{(b-a)^3}{24n^3}M_2.$$

En ajoutant membre à membre toutes ces inégalités – il y en a n –, on arrive à :

$$-\frac{(b-a)^3}{24n^2}M_2 \leq \int_a^b f(x)dx - \frac{b-a}{n} \sum_{k=0}^{n-1} f\left(\frac{x_k+x_{k+1}}{2}\right) \leq \frac{(b-a)^3}{24n^2}M_2$$

... c'est bien le résultat auquel on doit parvenir !

Remarquons que nous avons là affaire à une méthode de Newton-Cotes de type *ouvert* car les extrémités de l'intervalle $[a ; b]$ ne sont pas utilisées dans l'interpolation.

3.4 Écriture de fonctions

On peut bien sûr programmer cette méthode comme ci-dessous.

```
rectm
Define rectm(a,b,n)=
Func
Local h,s,i
b-a → h:0 → s
n
For i,1,n
s+f(a+h/2) → s
a+h → a
EndFor
Return h·s
EndFunc
```

On obtient par exemple les résultats suivants, évidemment plus précis que ceux de la méthode des rectangles à gauche ou à droite :

$f(x)$	$\frac{4}{x^2+1}$
rectm(0.,1,10)	3.142425985
rectm(0.,1,100)	3.14160098692
rectm(0.,1,1000)	3.14159273692
rectm(0.,1,10000)	3.14159265442
rectm(0.,1,100000)	3.14159265361
rectm(0.,1,1000000)	3.1415926536

Au bout de combien d'itérations obtiendra-t-on une erreur moindre que 10^{-6} ?

D'après le majorant de l'erreur donné plus haut, on a tout d'abord besoin de calculer le maximum de la dérivée seconde de f sur l'intervalle $[0 ; 1]$. Heureusement, il est donné par la calculatrice... ouf !

$f(x)$	$\frac{4}{x^2+1}$
fonc\maximum $\left(\left \frac{d^2}{dx^2}(f(x)) \right , 0, 1 \right)$	8

Enfin, d'après la majoration précédente, pour avoir une erreur inférieure ou égale à 10^{-6} , il suffit cette fois que :

$$\frac{(b-a)^3}{24n^2} M_2 = \frac{8}{24n^2} < 10^{-6} \text{ soit } 3n^2 > 10^6 \text{ ou encore } n > \sqrt{\frac{10^6}{3}} \approx 577,350.$$

Bref, à partir de 578, l'erreur commise est majorée par 10^{-6} . À comparer avec les 2 000 000 de la méthode précédente !

Ce que montre d'ailleurs l'écran suivant :

<code>rectm(0.,1,578)</code>	3.14159290303
<code>3.1415929030282-π</code>	0.000000249438
<code>2.494384E-7</code>	

Évidemment, le plus simple est d'incorporer le calcul de l'erreur dans la fonction elle-même. Peu de modifications à faire sur notre fonction initiale :

l'erreur e que l'on souhaite doit être proposée comme entrée au programme, plutôt que la valeur de n ;

l'appel à la fonction maximum de notre bibliothèque **func** permet de calculer le coefficient du

théorème $\frac{(b-a)^3}{24n^2} M_2$;

la boucle **For** devient une boucle **While**, dont on sort dès que $\frac{(b-a)^3}{24n^2} M_2$ est inférieur à l'erreur souhaitée ;

la valeur de n obtenue provoque alors le calcul de l'intégrale selon la méthode du point médian ;

on peut aussi, à titre d'information, faire afficher la valeur de n .

On obtient le code suivant :

```

rectm1
Define rectm1(a,b,e)=
Func
Local n,k
 $\frac{(b-a)^3}{24} \cdot \text{func} \setminus \text{maximum} \left( \left| \frac{d^2}{dx^2}(f(x)) \right|, a, b \right) \rightarrow k$ 
1 → n
While  $\frac{k}{n^2} > e$ 
  n+1 → n
EndWhile
Disp n
Return rectm(a,b,n)
EndFunc

```

Les résultats sont donnés dans l'écran qui suit :

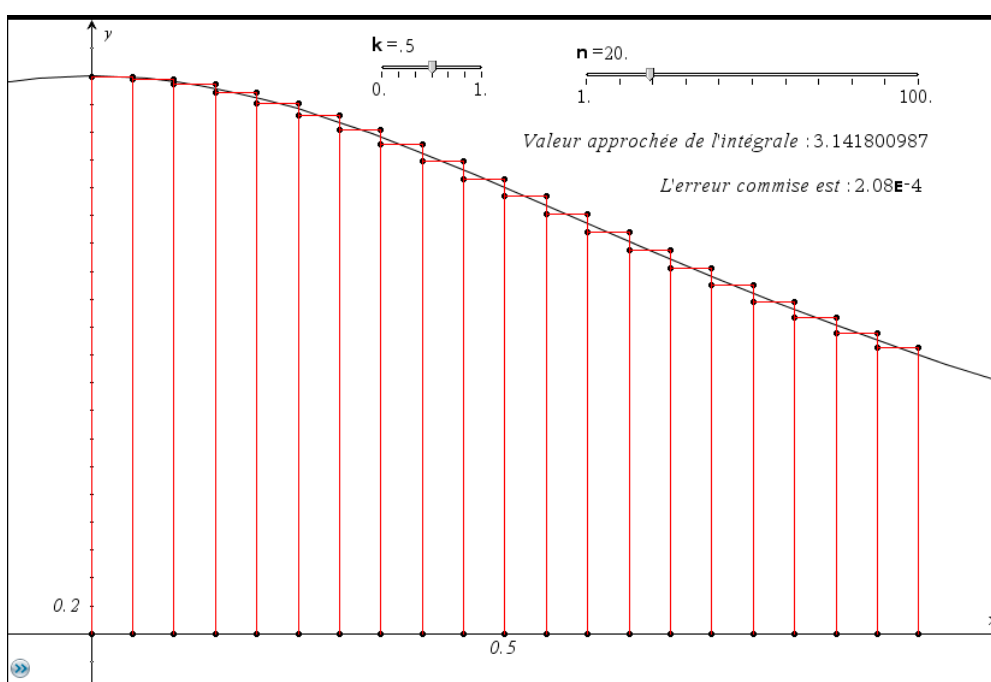
$rectm1(0.,1,10^{-6})$	578
	3.14159290303
$rectm1(0.,1,10^{-7})$	1826
	3.14159267858
$rectm1(0.,1,10^{-8})$	5774
	3.14159265608
$rectm1(0.,1,10^{-9})$	18258
	3.14159265378
$rectm1(0.,1,10^{-10})$	57736
	3.14159265342
	5/5

Les résultats renvoyés sont instructifs : on peut remarquer que pour diviser l'erreur par 10 (ce qui revient grosso-modo à gagner un chiffre significatif), il faut faire environ trois fois plus d'itérations. C'est pour le moins une performance relativement médiocre, notamment si l'on vise des précisions très petites (attention aussi à la propagation des erreurs d'arrondi si n est grand !).

Tout s'explique à partir des formules d'erreurs obtenues. En effet, si l'on pose $e_n = \frac{1}{12n^2}$, il est clair

que $\frac{e_p}{e_n} = \left(\frac{n}{p}\right)^2$... ce quotient vaut 1/10 lorsque $\frac{n}{p} = \sqrt{10} \approx 3,16$ soit $n \approx 3,16p$.

Enfin rappelons que nous avons déjà obtenu l'illustration graphique de cette méthode dans le paragraphe précédent.



Les performances restent encore modestes et pour les améliorer, on va chercher à interpoler la fonction f sur l'intervalle $[a ; b]$ par des polynômes de degré de plus en plus grand : le degré 1 conduit à la méthode des trapèzes et le degré 2 à la méthode de Simpson.

Rien n'interdit évidemment de poursuivre avec des polynômes de degré encore plus grand mais on a vu dans le premier paragraphe que ce n'était pas le meilleur choix pour atteindre une précision intéressante. On se tournera plutôt vers la méthode de Romberg – c'est une accélération de convergence – qui donnera une précision intéressante à moindre frais.

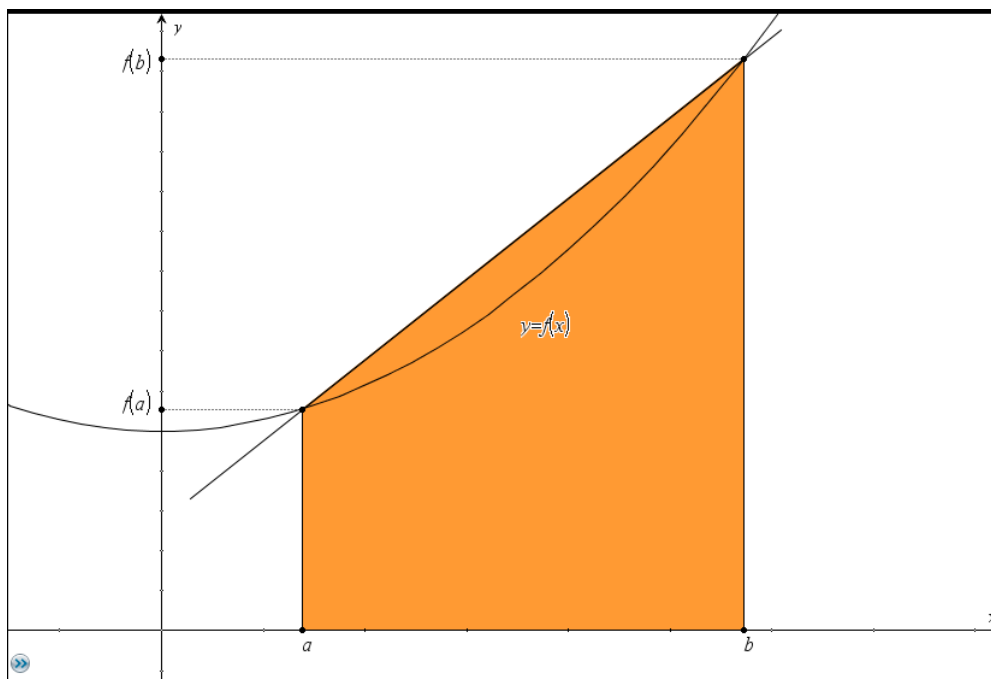
4. Méthode des trapèzes

4.1 Principe de la méthode

On se base sur une interpolation par un polynôme de degré 1. On remplace la fonction f par la fonction affine g qui coïncide avec f en a et en b , autrement dit par :

$$g(x) = \frac{f(b) - f(a)}{b - a}(x - a) + f(a).$$

On approxime alors le nombre I par $\int_a^b g(x) dx = (b - a) \frac{f(a) + f(b)}{2} = T$, c'est-à-dire, dans le cas où la fonction f est positive, par l'aire du trapèze rectangle dont les sommets ont pour coordonnées $(a, 0)$, $(a, f(a))$, $(b, 0)$ et $(b, f(b))$.



4.2 Majoration de l'erreur due à la méthode

Théorème

Si la fonction f est de classe C^2 sur l'intervalle $[a ; b]$ et si M_2 désigne le maximum de $|f''|$ sur $[a ; b]$, alors $|I - T| \leq \frac{(b - a)^3}{12} M_2$.

Démonstration

Commençons par voir à quoi peut bien ressembler $I - T$:

$$\begin{aligned} E = I - T &= \int_a^b f(x) dx - \int_a^b \left(\frac{f(b) - f(a)}{b - a} (x - a) + f(a) \right) dx \\ &= \int_a^b \left(f(x) - f(a) - \frac{f(b) - f(a)}{b - a} (x - a) \right) dx. \end{aligned}$$

C'est ce que l'on cherche à intégrer qui doit guider... Or :

$$f(x) - f(a) - \frac{f(b) - f(a)}{b - a} (x - a) = (x - a) \left(\frac{f(x) - f(a)}{x - a} - \frac{f(b) - f(a)}{b - a} \right) \text{ pour } x \neq a,$$

avec un deuxième facteur qui est bien de la forme $\varphi(x) - \varphi(b)$ ¹⁵ en posant :

$$\varphi(x) = \frac{f(x) - f(a)}{x - a}.$$

Le problème, c'est qu'on doit avoir une fonction définie et continue sur $[a; b]$... et notre fonction φ n'est pas définie en a . Qu'à cela ne tienne posons $\varphi(a) = f'(a)$ et par définition de la dérivée, on peut bien affirmer que la fonction φ est continue sur $[a; b]$.

D'autre part, elle est clairement dérivable sur $]a; b]$. Peu importe ce qui se passe en a , cela suffit pour appliquer le théorème des accroissements finis. Pour tout x de $[a; b]$, il existe donc un réel c de $]x; b[$ tel que :

$$\varphi(x) - \varphi(b) = (x - b)\varphi'(c)$$

ou encore pour $x \neq a$:

$$\frac{f(x) - f(a)}{x - a} = \frac{f(b) - f(a)}{b - a} + (x - b)\varphi'(c)$$

soit en multipliant par $x - a$ non nul des deux côtés :

$$f(x) - f(a) = \frac{f(b) - f(a)}{b - a} (x - a) + (x - a)(x - b)\varphi'(c).$$

Il est par ailleurs immédiat de vérifier que cette dernière égalité est encore vraie pour $x = a$: on trouve de chaque côté du symbole « = » le réel 0.

Il n'en faut pas plus pour majorer l'erreur E . Déroulons maintenant les calculs :

$$E = \int_a^b \left(f(x) - f(a) - \frac{f(b) - f(a)}{b - a} (x - a) \right) dx = \int_a^b (x - a)(b - x)\varphi'(c) dx$$

Par suite :

$$\begin{aligned} |E| &\leq \left| \int_a^b (x - a)(b - x)\varphi'(c) dx \right| \leq |\varphi'(c)| \int_a^b (x - a)(b - x) dx \\ &\leq \sup_{x \in [a, b]} |\varphi'(x)| \times \int_a^b (x - a)(b - x) dx \end{aligned}$$

¹⁵ Forme particulièrement agréable pour appliquer le théorème des accroissements finis.

Occupons-nous de chacun des facteurs.

Calculons la dérivée de φ pour x différent de a . Avec les règles habituelles de calcul, il vient :

$$\varphi'(x) = \frac{f'(x)(x-a) - (f(x) - f(a))}{(x-a)^2} = \frac{f(a) - f(x) - (a-x)f'(x)}{(x-a)^2}$$

qui peut s'écrire, au moyen de la formule de Taylor-Lagrange :

$$\varphi'(x) = \frac{\frac{(a-x)^2}{2} f''(d)}{(x-a)^2} = \frac{f''(d)}{2} \text{ où } d \in]a; x[.$$

Il en ressort que $|\varphi'(x)| \leq \frac{M_2}{2}$ pour tout x de l'intervalle $]a; b]$. Il en est de même de $\sup_{x \in]a, b]} |\varphi'(x)|$.

Enfin il reste maintenant à évaluer l'intégrale $\int_a^b (x-a)(b-x)dx$... un jeu d'enfant, auquel la TI-Nspire accepte de jouer :

$$\int_a^b ((x-a) \cdot (b-x)) dx = \frac{-(a^3 - 3a^2b + 3ab^2 - b^3)}{6}$$

$$\text{factor}\left(\frac{-(a^3 - 3a^2b + 3ab^2 - b^3)}{6}\right) = \frac{-(a-b)^3}{6}$$

Le calcul au demeurant n'est guère délicat, même à la main. Procédons par changement de variable en posant $X = x - c$ où $c = \frac{a+b}{2}$; donc $dX = dx$.

L'intégrale devient :

$$\int_a^b (x-a)(b-x)dx = \int_{a-c}^{b-c} (X+c-a)(b-X-c)dX = \int_{-h}^h (X+h)(h-X)dX$$

en posant $h = \frac{b-a}{2}$. Poursuivons le calcul :

$$\begin{aligned} \int_a^b (x-a)(b-x)dx &= \int_{-h}^h (h^2 - X^2)dX = 2h^3 - \left[\frac{X^3}{3} \right]_{-h}^h = 2h^3 - \frac{2h^3}{3} = \frac{4h^3}{3} \\ &= \frac{4(b-a)^3}{24} = \frac{(b-a)^3}{6} \end{aligned}$$

En conclusion, on peut écrire que

$$|E| \leq \sup_{x \in]a, b]} |\varphi'(x)| \int_a^b (x-a)(b-x)dt \leq \frac{M_2}{2} \frac{(b-a)^3}{6} \leq \frac{M_2(b-a)^3}{12}$$

ce que nous avons annoncé... Le calcul n'est pas fondamentalement délicat une fois que l'idée de début a été bien comprise ! Il faut juste être patient¹⁶...

¹⁶ Une démonstration intéressante, particulièrement courte et facile à mémoriser, est proposée par Daniel Perrin sur son site :

<http://www.math.u-psud.fr/~perrin/CAPES/analyse/inte%CC%81grales-e%CC%81qua-diff/calcul-approche-d%27integrale.pdf>

Le principe de la démonstration se généralise à la méthode de Simpson.

4.3 Méthode composite des trapèzes

Théorème

Si la fonction f est de classe C_2 sur l'intervalle $[a; b]$ et si M_2 désigne le maximum de $|f''|$ sur $[a; b]$, alors on peut approximer I par :

$$T_n = \frac{b-a}{n} \sum_{k=0}^{n-1} \frac{f(x_k) + f(x_{k+1})}{2}.$$

L'erreur commise est majorée par $\frac{(b-a)^3}{12n^2} M_2$.

Démonstration

Appliquons le résultat précédent pour chacun des intervalles $[x_k, x_{k+1}]$, pour k variant entre 0 et $n-1$. On obtient :

$$-\frac{(b-a)^3}{12n^3} M_2 \leq \int_{x_k}^{x_{k+1}} f(x) dx - \frac{b-a}{n} \frac{f(x_k) + f(x_{k+1})}{2} \leq \frac{(b-a)^3}{12n^3} M_2.$$

En ajoutant membre à membre toutes ces inégalités – il y en a n –, on arrive à :

$$-\frac{(b-a)^3}{12n^2} M_2 \leq \int_a^b f(x) dx - \frac{b-a}{n} \sum_{k=0}^{n-1} \frac{f(x_k) + f(x_{k+1})}{2} \leq \frac{(b-a)^3}{12n^2} M_2$$

... c'est bien le résultat auquel on doit parvenir !

Remarquons que :

$$T_n = \frac{b-a}{n} \sum_{k=0}^{n-1} \frac{f(x_k) + f(x_{k+1})}{2} = \frac{1}{2} \left(\frac{b-a}{n} \sum_{k=0}^{n-1} f(x_k) + \frac{b-a}{n} \sum_{k=0}^{n-1} f(x_{k+1}) \right) = \frac{1}{2} (R_n^i + R_n^r).$$

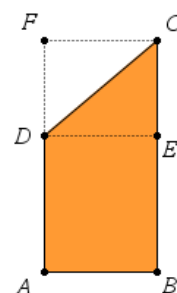
• Liens avec la méthode des rectangles

La méthode des trapèzes n'est autre que la moyenne entre la méthode des rectangles à gauche et des rectangles à droite. En particulier, ceci garantit que la suite (T_n) converge vers l'intégrale I .

Géométriquement, cela s'interprète simplement. Si on note a_1 l'aire du rectangle $ABED$ et a_2 l'aire du rectangle $ABCF$, il est clair que l'aire du trapèze $ABCD$ vaut :

$$a_1 + \frac{a_2 - a_1}{2} = \frac{a_1 + a_2}{2}$$

c'est-à-dire la moyenne des aires des rectangles...



Y-a-t-il aussi un lien avec la méthode des rectangles avec point médian ?

On dispose du résultat suivant : $T_{2n} = \frac{1}{2} (T_n + R_n)$, dont la vérification ne pose pas de problème.

Il suffit d'écrire les égalités. On a :

$$\begin{aligned}
\frac{1}{2}(T_n + R_n) &= \frac{b-a}{2n} \sum_{k=0}^{n-1} \frac{f(x_k) + f(x_{k+1})}{2} + \frac{b-a}{2n} \sum_{k=0}^{n-1} f\left(\frac{x_k + x_{k+1}}{2}\right) \\
&= \frac{b-a}{2n} \sum_{k=0}^{n-1} \frac{f(x_k) + f(x_{k+1})}{2} + \frac{b-a}{2n} \sum_{k=0}^{n-1} \frac{f(x'_k) + f(x'_k)}{2} \\
&= \frac{b-a}{2n} \sum_{k=0}^{n-1} \left(\frac{f(x_k) + f(x'_k)}{2} + \frac{f(x'_k) + f(x_{k+1})}{2} \right) \\
&= T_{2n}
\end{aligned}$$

où x'_k désigne $\frac{x_k + x_{k+1}}{2} \dots$

En d'autres termes, on peut calculer T_{2n} lorsque l'on connaît R_n et T_n .

Remarquons aussi que si la fonction est convexe, on a les inégalités :

$$T_n \geq I \geq R_n$$

(les inégalités ont lieu dans l'autre sens si la fonction est concave).

La méthode des rectangles avec point médian et la méthode des trapèzes donnent un encadrement de l'intégrale cherchée, ce qui est toujours intéressant pour les calculs approchés.

4.4 Écriture d'une fonction

La méthode peut être programmée sans difficulté. On peut facilement repartir d'une des fonctions déjà écrites.

```

trapeze
Define LibPriv trapeze(a,b,n)=
Func
Local h,s,i
(b-a)/n → h:0 → s
For i,1,n
s + (f(a)+f(a+h))/2 → s
a+h → a
EndFor
Return h*s
EndFunc

```

On obtient par exemple les résultats suivants, à peine moins précis que ceux de la méthode des rectangles avec point médian, dont on a vu qu'elle était aussi d'ordre 1 et qu'elle pouvait se ramener à une espèce de méthode des trapèzes :

<code>trapeze(0.,1,10)</code>	3.13992598891
<code>trapeze(0.,1,100)</code>	3.14157598692
<code>trapeze(0.,1,1000)</code>	3.14159248692
<code>trapeze(0.,1,10000)</code>	3.14159265192
<code>trapeze(0.,1,100000)</code>	3.14159265359
<code>trapeze(0.,1,1000000)</code>	3.1415926536

On peut vérifier que la moyenne des rectangles à gauche et à droite redonne exactement la même chose.

<code>mean(rect(0.,1,10))</code>	3.13992598891
<code>mean(rect(0.,1,100))</code>	3.14157598692
<code>mean(rect(0.,1,1000))</code>	3.14159248692
<code>mean(rect(0.,1,10000))</code>	3.14159265192

Reprenons notre calcul d'erreur « standard » : au bout de combien d'itérations obtiendra-t-on une erreur moindre que 10^{-6} ?

On sait déjà que $|f''(x)| = 8$. Enfin, d'après la majoration précédente, pour avoir une erreur inférieure ou égale à 10^{-6} , il suffit cette fois que :

$$\frac{(b-a)^3}{12n^2} M_2 = \frac{8}{12n^2} < 10^{-6} \text{ soit } \frac{3}{2} n^2 > 10^6 \text{ ou encore } n > \sqrt{\frac{10^6}{1,5}} \approx 816,50.$$

À partir de 817, l'erreur commise est majorée par 10^{-6} . C'est une performance un peu moins bonne que celle de la méthode des rectangles avec point médian. Elle est toutefois du même ordre de grandeur.

On peut aussi utiliser les inégalités précédentes, $R_n \leq I \leq T_n$, valables pour tout entier naturel n , pour avoir un encadrement de l'intégrale cherchée :

<code>{rectm(0.,1,p),trapeze(0.,1,p)} → integ(p)</code>	Terminé
<code>integ(10)</code>	{ 3.142425985, 3.13992598891 }
<code>integ(100)</code>	{ 3.14160098692, 3.14157598692 }
<code>integ(1000)</code>	{ 3.14159273692, 3.14159248692 }
<code>integ(10000)</code>	{ 3.14159265442, 3.14159265192 }

Le dernier résultat renvoyé permet d'être sûr des décimales 3,14159265.

Il reste, comme on l'a fait précédemment à intégrer le calcul d'erreur dans la fonction elle-même. Cette nouvelle fonction se construit de façon analogue à celle qu'on a construite pour la méthode des rectangles avec point médian.

<code>trapeze1</code>	2/8
Define trapeze1 (a,b,e)=	
Func	
Local n,k	
$\frac{(b-a)^3}{12} \cdot \text{func} \backslash \text{maximum} \left(\left \frac{d^2}{dx^2}(f(x)) \right , a, b \right) \rightarrow k$	
1 → n	
While $\frac{k}{n^2} > e$	
n+1 → n	
EndWhile	
Disp n	
Return <code>trapeze</code> (a,b,n)	
EndFunc	

Les performances sont légèrement inférieures à la méthode des rectangles avec point médian, comme on peut le supposer en regardant le majorant de l'erreur.

$\text{trapezeI}(0,1,10^{-6})$	817
	3.1415924039
$\text{trapezeI}(0,1,10^{-7})$	2582
	3.14159262859
$\text{trapezeI}(0,1,10^{-8})$	8165
	3.14159265109
$\text{trapezeI}(0,1,10^{-9})$	25820
	3.14159265346
$\text{trapezeI}(0,1,10^{-10})$	81650
	3.14159265341
	5/5

Pour terminer, l'approche graphique peut être adaptée sans problème depuis la méthode des rectangles avec point médian. Nous ne détaillerons pas plus avant les écrans qui suivent.

4.5 Illustration graphique

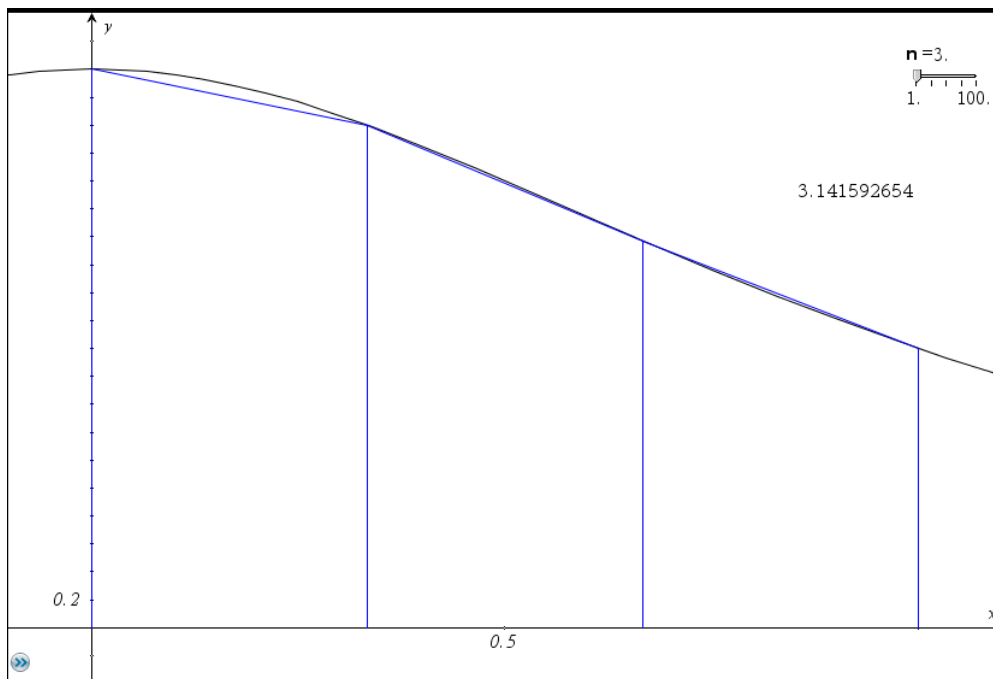
Sans plus de commentaire, si l'on veut faire vite, on peut repartir de ce que l'on a fait pour la méthode des rectangles. C'est même un peu plus simple : au lieu de deux rectangles à tracer à chaque étape, il n'y a plus qu'un trapèze dont les sommets ont pour coordonnées sont :

$$(a+i \times h ; 0), (a+i \times h ; f(a+i \times h)), (a+(i+1) \times h ; f(a+(i+1) \times h) \text{ et } (a+(i+1) \times h ; 0)$$

Le programme qui donne la liste des abscisses ne change donc pas; celui qui donne la liste des ordonnées est à peine différent.

x_trapeze	5/8	y_trapeze	8/8
Define LibPriv x_trapeze (a,b,n)=		Define LibPriv y_trapeze (a,b,n)=	
Func		Func	
Local h,l,i		Local h,l,i,c	
$\frac{b-a}{n} \rightarrow h$		$\frac{b-a}{n} \rightarrow h$	
$\{\} \rightarrow l$		$\{\} \rightarrow l$	
For i,1,n		For i,1,n	
augment(l,{a,a+h,a+h}) $\rightarrow l$		augment(l,{0,f(a),f(a+h),0}) $\rightarrow l$	
$a+h \rightarrow a$		$a+h \rightarrow a$	
EndFor		EndFor	
Return l		Return l	
EndFunc		EndFunc	

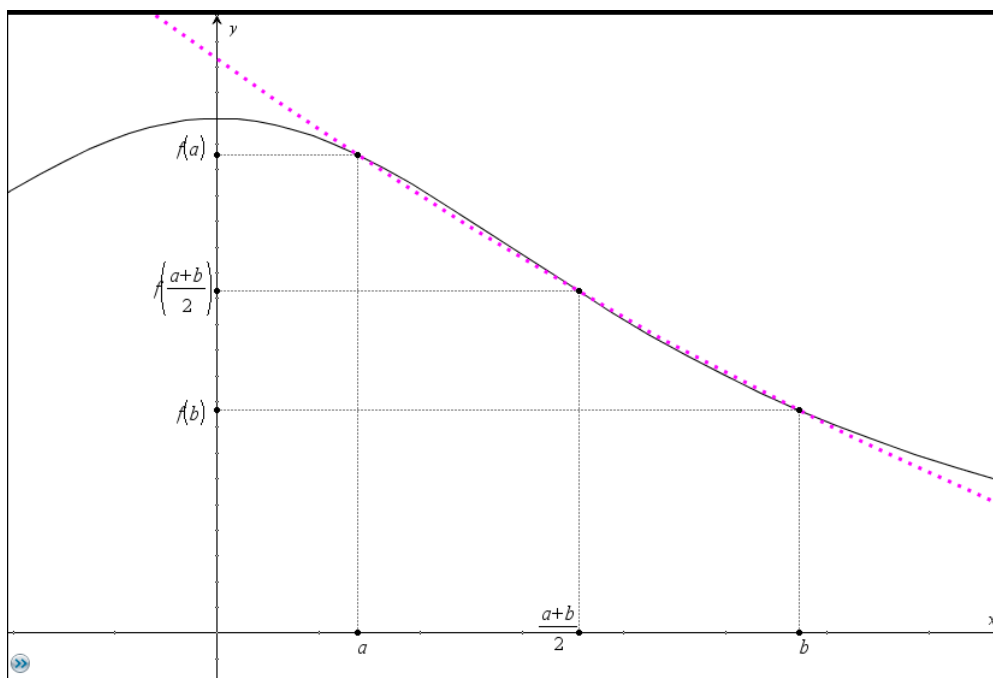
La représentation graphique de la méthode peut alors ressembler à ce qui suit :



5. Méthode de Simpson

5.1 Principe de la méthode

Interpolons maintenant la fonction f par la fonction polynôme g de degré 2 qui coïncide avec f aux points a , b et $\frac{a+b}{2}$. En pointillé sur le dessin qui suit, est tracée la courbe représentative de g , fonction polynôme du second degré, qui passe par les points de coordonnées $(a, f(a))$, $(b, f(b))$ et $\left(\frac{a+b}{2}, f\left(\frac{a+b}{2}\right)\right)$. Nous avons juste besoin de préciser ce que vaut l'intégrale d'une telle fonction du second degré.



• Détermination de la fonction g

Sa détermination effective n'est pas utile pour le calcul approchée de l'intégrale mais nous nous servirons de ce résultat pour l'illustration graphique.

Nous chercherons la fonction g sous la forme¹⁷ $g(x) = A(x-c)^2 + B(x-c) + C$ en posant $c = \frac{a+b}{2}$.

Traduisons les conditions pour déterminer les constantes A , B et C :

$$\begin{cases} f(a) = g(a) = A(a-c)^2 + B(a-c) + C = A\left(\frac{a-b}{2}\right)^2 + B\left(\frac{a-b}{2}\right) + C \\ f(b) = g(b) = A(b-c)^2 + B(b-c) + C = A\left(\frac{b-a}{2}\right)^2 + B\left(\frac{b-a}{2}\right) + C \\ f(c) = g(c) = C = f\left(\frac{a+b}{2}\right) \end{cases}$$

La dernière égalité donne immédiatement $C = f\left(\frac{a+b}{2}\right) = f(c)$.

Pour obtenir A et B , il reste à résoudre le système :

$$\begin{cases} A\left(\frac{a-b}{2}\right)^2 + B\left(\frac{a-b}{2}\right) + f(c) = f(a) \\ A\left(\frac{a-b}{2}\right)^2 - B\left(\frac{a-b}{2}\right) + f(c) = f(b) \end{cases}$$

On en déduit, successivement par addition, puis soustraction membre à membre :

$$A = \frac{2(f(a) + f(b) - 2f(c))}{(a-b)^2}$$

$$B = \frac{f(a) - f(b)}{a-b}$$

Si bien que la fonction g est définie par :

$$g(x) = \frac{2(f(a) + f(b) - 2f(c))}{(a-b)^2}(x-c)^2 + \frac{f(a) - f(b)}{a-b}(x-c) + f(c)$$

en rappelant que $c = \frac{a+b}{2}$.

Remarquons que ces calculs peuvent être menés avec la TI-Nspire :

$\frac{a+b}{2} \rightarrow c$	$\frac{a+b}{2}$
$\alpha \cdot (x-c)^2 + \beta \cdot (x-c) + \gamma \rightarrow f(x)$	Terminé
solve $\left(\begin{cases} g(a)=f(a) \\ g(b)=f(b) \\ g(c)=f(c) \end{cases}, \alpha, \beta, \gamma \right)$	$\alpha = \frac{-2 \cdot \left(2 \cdot g\left(\frac{a+b}{2}\right) - g(a) - g(b) \right)}{(a-b)^2}$ and $\beta = \frac{g(a) - g(b)}{a-b}$ and $\gamma = g\left(\frac{a+b}{2}\right)$

¹⁷ Attention à la forme très particulière de g selon les puissances de $x-c$

- **Valeur de l'intégrale** $S = \int_a^b g(x)dx \perp$

Nous en aurons par contre besoin pour la mise en œuvre effective de la méthode de Simpson. Il se trouve que l'intégrale ne dépend que de $g(a)$, $g(b)$ et $g\left(\frac{a+b}{2}\right)$, autrement dit de la valeur de la fonction en seulement trois réels. C'est ce que l'on appelle parfois la formule des trois niveaux. On a le résultat suivant :

Théorème : formule des trois niveaux

Soit g une fonction polynôme de degré au plus 2. Alors :

$$\int_a^b g(x)dx = \frac{b-a}{6} \left(g(a) + 4g\left(\frac{a+b}{2}\right) + g(b) \right).$$

Démonstration

Plutôt que de faire un calcul général d'emblée, nous allons nous appuyer sur des propriétés de linéarité.

Tout d'abord, la formule est vraie si la fonction g est affine.

En effet, d'une part, il est clair que $\int_a^b g(x)dx = (b-a) \frac{g(a)+g(b)}{2}$.

D'autre part, puisque g est une fonction affine, on sait que $g\left(\frac{a+b}{2}\right) = \frac{1}{2}g(a) + \frac{1}{2}g(b)$ et l'on peut écrire :

$$\begin{aligned} \frac{b-a}{6} \left(g(a) + 4g\left(\frac{a+b}{2}\right) + g(b) \right) &= \frac{b-a}{6} (g(a) + 2g(a) + 2g(b) + g(b)) \\ &= \frac{b-a}{2} (g(a) + g(b)) \end{aligned}$$

ce qui prouve l'égalité.

Montrons maintenant que la formule est vraie pour la fonction carré.

On a en effet :

$$\int_a^b x^2 dx = \frac{b^3 - a^3}{3}$$

tandis que :

$$\begin{aligned} \frac{b-a}{6} \left(g(a) + 4g\left(\frac{a+b}{2}\right) + g(b) \right) &= \frac{b-a}{6} \left(a^2 + 4\left(\frac{a+b}{2}\right)^2 + b^2 \right) \\ &= \frac{b-a}{6} (a^2 + a^2 + 2ab + b^2 + b^2) \\ &= \frac{b-a}{6} (2a^2 + 2ab + 2b^2) \\ &= \frac{b-a}{3} (a^2 + ab + b^2) = \frac{b^3 - a^3}{3} \end{aligned}$$

Par combinaison linéaire, l'égalité annoncée est bien vraie pour tout polynôme de degré au plus 2.

Le calcul formel TI-Nspire peut appuyer notre démonstration, oserons-nous dire *faire* la démonstration ?

$g(x) := a \cdot x^2 + \beta \cdot x + \gamma$	Terminé
$\text{expand} \left(\int_a^b (g(x)) dx \right)$	$\frac{-a^3 \cdot a}{3} - \frac{a^2 \cdot \beta}{2} - a \cdot \gamma + \frac{b^3 \cdot a}{3} + \frac{b^2 \cdot \beta}{2} + b \cdot \gamma$
$\text{expand} \left(\frac{b-a}{6} \cdot \left(g(a) + 4 \cdot g\left(\frac{a+b}{2}\right) + g(b) \right) \right)$	$\frac{-a^3 \cdot a}{3} - \frac{a^2 \cdot \beta}{2} - a \cdot \gamma + \frac{b^3 \cdot a}{3} + \frac{b^2 \cdot \beta}{2} + b \cdot \gamma$

Ajoutons que la propriété est aussi vraie pour $g(x) = x^3 \dots$ et donc pour tout polynôme de degré 3... Nous nous contenterons cette fois d'une *démonstration* « assistée » par TI-Nspire :

$g(x) := x^3$	Terminé
$\text{expand} \left(\int_a^b (g(x)) dx \right)$	$\frac{b^4}{4} - \frac{a^4}{4}$
$\text{expand} \left(\frac{b-a}{6} \cdot \left(g(a) + 4 \cdot g\left(\frac{a+b}{2}\right) + g(b) \right) \right)$	$\frac{b^4}{4} - \frac{a^4}{4}$
$g(x) := a \cdot x^3 + \beta \cdot x^2 + \gamma \cdot x + \delta$	Terminé
$\text{expand} \left(\int_a^b (g(x)) dx \right) = \text{expand} \left(\frac{b-a}{6} \cdot \left(g(a) + 4 \cdot g\left(\frac{a+b}{2}\right) + g(b) \right) \right)$	true

Mais la formule n'est plus vraie pour $g(x) = x^4$. Il vaut quand même mieux s'en assurer ! Fini le bel ordonnancement !

$g(x) := x^4$	Terminé
$\text{expand} \left(\int_a^b (g(x)) dx \right)$	$\frac{b^5}{5} - \frac{a^5}{5}$
$\text{expand} \left(\frac{b-a}{6} \cdot \left(g(a) + 4 \cdot g\left(\frac{a+b}{2}\right) + g(b) \right) \right)$	$\frac{-5 \cdot a^5}{24} + \frac{a^4 \cdot b}{24} - \frac{a^3 \cdot b^2}{12} + \frac{a^2 \cdot b^3}{12} - \frac{a \cdot b^4}{24} + \frac{5 \cdot b^5}{24}$

• Méthode de Simpson

Nous sommes en mesure de donner tous les éléments de la méthode de Simpson. Avec les notations habituelles, on est donc amené à approximer le nombre $I = \int_a^b f(x) dx$ par :

$$\int_a^b g(x) dx = \frac{b-a}{6} \left(g(a) + 4g\left(\frac{a+b}{2}\right) + g(b) \right) = \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right),$$

puisque g est le polynôme d'interpolation de f de degré 2 aux points a , b et $\frac{a+b}{2}$.

5.2 Majoration de l'erreur due à la méthode

Théorème

Si la fonction f est de classe C^4 sur l'intervalle $[a ; b]$ et si M_4 désigne le maximum de $|f^{(4)}|$ sur $[a ; b]$, alors $|I - S| \leq \frac{(b-a)^5}{2880} M_4$.

Comme précédemment, on peut remarquer que cette majoration, qui fait intervenir la dérivée quatrième, montre que la méthode de Simpson est d'ordre 3 : autrement dit, elle est exacte pour tous les polynômes de degré 3, dont la dérivée quatrième est nulle !

Démonstration

Commençons par déterminer ce que vaut $I - S$. Si on appelle F une primitive de f sur $[a ; b]$, on peut écrire :

$$\begin{aligned} I - S &= \int_a^b f(t) dt - \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right) \\ &= F(b) - F(a) - \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right). \end{aligned}$$

Il est intéressant dans cette formule de faire apparaître la symétrie autour de $c = \frac{a+b}{2}$.

En posant $h = \frac{b-a}{2}$, l'égalité précédente s'écrit :

$$I - S = F(c+h) - F(c-h) - \frac{h}{3} (f(c-h) + 4f(c) + f(c+h)).$$

Soit alors M le nombre réel tel que :

$$F(c+h) - F(c-h) - \frac{h}{3} (f(c-h) + 4f(c) + f(c+h)) = Mh^5.$$

Observons que M est bien toujours défini. Il vaut :

$$\frac{F(c+h) - F(c-h) - \frac{h}{3} (f(c-h) + 4f(c) + f(c+h))}{h^5} = M.$$

Passons maintenant à une fonction. Pour t appartenant à $[0 ; h]$, on définit la fonction G par :

$$G(t) = F(c+t) - F(c-t) - \frac{t}{3} (f(c-t) + 4f(c) + f(c+t)) - Mt^5.$$

Il est clair que G est continue et dérivable sur $[0 ; h]$.

Par ailleurs, d'après la définition de M et celle de G ¹⁸, on peut écrire :

$$\text{d'une part, } G(h) = F(c+h) - F(c-h) - \frac{h}{3} (f(c-h) + 4f(c) + f(c+h)) - Mh^5 = 0,$$

$$\text{d'autre part, } G(0) = F(c) - F(c) - \frac{0}{3} (f(c) + 4f(c) + f(c)) - M \times 0^5 = 0.$$

D'après le théorème de Rolle¹⁹ sur $[0 ; h]$, il existe un réel c_1 de l'intervalle $]0 ; h[$ tel que $G'(c_1) = 0$.

Calculons donc $G'(t)$ sur $[0 ; h]$.

¹⁸ On pourrait même ajouter que c'est étudié pour... comme dans le sketch de Fernand Raynaud...

¹⁹ Tout concourt depuis le début à fabriquer une fonction qui permette l'application de ce célèbre théorème.

$$\begin{aligned}
 G'(t) &= f(c+t) + f(c-t) - \frac{1}{3}(f(c-t) + 4f(c) + f(c+t)) - \frac{t}{3}(f'(c+t) - f'(c-t)) - 5Mt^4 \\
 &= \frac{2}{3}f(c+t) + \frac{2}{3}f(c-t) - \frac{4}{3}f(c) - \frac{t}{3}(f'(c+t) - f'(c-t)) - 5Mt^4
 \end{aligned}$$

On peut encore poursuivre l'application du théorème de Rolle sur l'intervalle $[0 ; c_1]$ car $G'(0) = 0 = G'(c_1)$. Les hypothèses sont bien remplies et l'on peut déduire l'existence d'un réel c_2 de l'intervalle $]0 ; c_1[$ tel que $G''(c_2) = 0$.

Calculons donc $G''(t)$ sur $[0 ; h]$. Tous calculs faits, on arrive à :

$$\begin{aligned}
 G''(t) &= \frac{2}{3}f'(c+t) - \frac{2}{3}f'(c-t) - \frac{1}{3}(f'(c+t) - f'(c-t)) - \frac{t}{3}(f''(c+t) + f''(c-t)) - 20Mt^3 \\
 &= \frac{1}{3}f'(c+t) - \frac{1}{3}f'(c-t) - \frac{t}{3}(f''(c+t) + f''(c-t)) - 20Mt^3
 \end{aligned}$$

Pourquoi s'arrêter en si bon chemin ? Comme $G''(0) = 0 = G''(c_2)$, d'après le théorème de Rolle sur l'intervalle $[0 ; c_2]$, il existe donc un réel c_3 de l'intervalle $]0 ; c_2[$ tel que $G^{(3)}(c_3) = 0$.

Or :

$$\begin{aligned}
 G^{(3)}(t) &= \frac{1}{3}f''(c+t) + \frac{1}{3}f''(c-t) - \frac{1}{3}(f''(c+t) + f''(c-t)) - \frac{t}{3}(f^{(3)}(c+t) - f^{(3)}(c-t)) - 60Mt^2 \\
 &= -\frac{t}{3}(f^{(3)}(c+t) - f^{(3)}(c-t)) - 60Mt^2
 \end{aligned}$$

Inutile d'aller plus loin. L'expression obtenue est suffisamment simple. On peut alors écrire :

$$G^{(3)}(c_3) = -\frac{c_3}{3}(f^{(3)}(c_3+t) - f^{(3)}(c_3-t)) - 60Mc_3^2 = 0$$

d'où l'on peut déduire :

$$M = -\frac{1}{180} \frac{f^{(3)}(c+c_3) - f^{(3)}(c-c_3)}{c_3} = -\frac{1}{180} \frac{2c_3 f^{(4)}(c_4)}{c_3} = -\frac{f^{(4)}(c_4)}{90}$$

où $c_4 \in]c-c_3 ; c+c_3[$, d'après le théorème des accroissements finis appliqué à la fonction $f^{(3)}$ sur l'intervalle $[c-c_3 ; c+c_3]$.

Il en résulte immédiatement que :

$$|M| \leq \frac{1}{90} M_4$$

et donc :

$$\begin{aligned}
 |I - S| &= \left| F(c+h) - F(c-h) - \frac{h}{3}(f(c-h) + 4f(c) + f(c+h)) \right| \\
 &= |Mh^5| \leq \frac{M_4}{90} \frac{(b-a)^5}{32} = \frac{M_4(b-a)^5}{2880}
 \end{aligned}$$

C'est bien ce que l'on cherchait à prouver²⁰ !

²⁰ Voir comme plus haut pour la méthode des trapèzes la démonstration de Daniel Perrin sur son site.

5.3 Méthode composite de Simpson

Théorème

Si la fonction f est de classe C_4 sur l'intervalle $[a ; b]$ et si M_4 désigne le maximum de $|f^{(4)}|$ sur $[a ; b]$, alors on peut approximer I par :

$$S_n = \frac{b-a}{6n} \sum_{k=0}^{n-1} \left(f(x_k) + 4f\left(\frac{x_k + x_{k+1}}{2}\right) + f(x_{k+1}) \right).$$

L'erreur commise est majorée par $\frac{(b-a)^5}{2880n^4} M_4$.

Démonstration

Appliquons le résultat précédent pour chacun des intervalles $[x_k, x_{k+1}]$, pour k variant entre 0 et $n-1$. On obtient :

$$-\frac{(b-a)^5}{2880n^5} M_4 \leq \int_{x_k}^{x_{k+1}} f(x) dx - \frac{b-a}{6n} \frac{f(x_k) + 4f\left(\frac{x_k + x_{k+1}}{2}\right) + f(x_{k+1})}{2} \leq \frac{(b-a)^5}{2880n^5} M_4.$$

En ajoutant membre à membre toutes ces inégalités – il y en a n –, on arrive à :

$$-\frac{(b-a)^5}{2880n^4} M_4 \leq \int_a^b f(x) dx - \frac{b-a}{6n} \sum_{k=0}^{n-1} \frac{f(x_k) + 4f\left(\frac{x_k + x_{k+1}}{2}\right) + f(x_{k+1})}{2} \leq \frac{(b-a)^5}{2880n^4} M_4$$

... c'est bien le résultat auquel on doit parvenir !

• Liens avec les méthodes précédentes

Avec les notations habituelles, pour tout entier $n \geq 1$, on a $S_n = \frac{T_n + 2R_n}{3}$.

Démonstration

En effet, on peut le vérifier pour $n = 1$ c'est-à-dire pour la méthode à un pas. C'est alors un calcul élémentaire. On sait que :

$$T_1 = (b-a) \frac{f(a) + f(b)}{2} \text{ et } R_1 = (b-a) f\left(\frac{a+b}{2}\right).$$

En conséquence :

$$\begin{aligned} \frac{T_1 + 2R_1}{3} &= \frac{1}{3} \left((b-a) \frac{f(a) + f(b)}{2} + 2 \times (b-a) f\left(\frac{a+b}{2}\right) \right) = \frac{b-a}{3} \frac{f(a) + f(b) + 4f\left(\frac{a+b}{2}\right)}{2} \\ &= \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right) = S_1 \end{aligned}$$

ce qui est exactement la formule de Simpson. La relation est donc vraie aussi pour n'importe quel découpage de l'intervalle $[a ; b]$.

Ajoutons à ce résultat que, du fait que $T_{2n} = \frac{1}{2}(T_n + R_n)$, on en déduit que :

$$\frac{4T_{2n} - T_n}{3} = \frac{4 \frac{T_n + R_n}{2} - T_n}{3} = \frac{T_n + 2R_n}{3} = S_n$$

Or il se trouve, comme nous le verrons plus loin, que l'égalité $S_n = \frac{4T_{2n} - T_n}{3}$ correspond à la première accélération de convergence de la méthode de Romberg.

5.4 Écriture d'une fonction

Ci-dessous une fonction qui renvoie le résultat donné par la méthode de Simpson :

```

simpson
Define LibPriv simpson(a,b,n)=
Func
Local h,s,i
(b-a)/n → h:0 → s
For i,1,n
  f(a)+4·f(a+h/2)+f(a+h)
  s+————→ s
      6
  a+h → a
EndFor
Return h·s
EndFunc
  
```

Avec notre exemple, on obtient des résultats dont la convergence vers π est spectaculaire, bien plus qu'avec aucune des méthodes vues précédemment :

$f(x)$	$\frac{4}{x^2+1}$
$\text{simpson}(0.,1,10)$	3.14159265297
$\text{simpson}(0.,1,100)$	3.14159265359
$\text{simpson}(0.,1,1000)$	3.14159265359
$\pi - 3.1415926535894$	4.E-13
4.E-13	

Reprenons notre calcul d'erreur de référence. La calculatrice nous donne

$$\text{fonc}\backslash\text{maximum}\left(\left|\frac{d^4}{dx^4}(f(x))\right|,0,1\right)$$

Par conséquent, pour avoir une erreur inférieure ou égale à 10^{-6} , il suffit d'avoir cette fois :

$$\frac{(b-a)^5}{2880n^4} M_4 = \frac{96}{2880n^4} < 10^{-6} \text{ soit } 30n^4 > 10^6 \text{ et } n > 13,512.$$

À partir de $n = 14$, l'erreur est plus petite que 10^{-6} . La comparaison avec les méthodes précédentes est édifiante.

En intégrant le calcul de l'erreur dans la fonction, on arrive à :

```
"simpson1" enregistrement effectué
Define LibPriv simpson1(a,b,e)=
Func
Local n,k

$$\frac{(b-a)^5}{2880} \cdot \text{fonc} \backslash \text{maximum} \left( \left| \frac{d^4}{dx^4} (f(x)) \right|, a, b \right) \rightarrow k$$

1 → n
While  $\frac{k}{n^4} > e$ 
  n+1 → n
EndWhile
Disp n
Return simpson(a,b,n)
EndFunc
```

Avec par exemple :

```
simpson1(0.,1,10-6)
14
3.14159265351
```

5.5 Illustration graphique

Elle est un peu plus délicate, sans pour autant être trop technique pour peu que l'on soit progressif.

Le plus simple est de commencer par ce qui demeure, ou qui change peu. Tout d'abord, les deux fonctions qui permettent de construire le nuage de points... et dans le tableur les deux colonnes, nommées **xx** et **yy** pour en demander la représentation graphique. Nous nous contenterons de placer les points de coordonnées :

$$(a+i \times h ; f(a+i \times h)), (a+i \times h/2 ; f(a+i \times h/2)), (a+(i+1) \times h ; f(a+i \times h))$$

sans cependant les relier.

x_simpson	0/8	y_simpson	8/8
Define LibPriv x_simpson (a,b,n)=		Define LibPriv y_simpson (a,b,n)=	
Func		Func	
Local xx,h,i		Local yy,h,i	
{ } → xx		{ } → yy	
$\frac{b-a}{n} \cdot 1. \rightarrow h$		$\frac{b-a}{n} \cdot 1. \rightarrow h$	
For i,1,n		For i,1,n	
augment(xx, {a, a+ $\frac{h}{2}$, a+h}) → xx		augment(yy, {f(a), f(a+ $\frac{h}{2}$), f(a+h)}) → yy	
a+h → a		a+h → a	
EndFor		EndFor	
Return xx		Return yy	
EndFunc		EndFunc	

Ce qui change... On travaille avec des paraboles, d'où la nécessité d'écrire une fonction qui récupère l'équation de cette parabole.

Facile car nous l'avons déterminée plus haut²¹... C'est ce que fait la fonction suivante :

```

interpol2
Define LibPriv interpol2(a,b)=
Func
Local c
a+b
2 → c
Return expand( (2*(f(a)+f(b))-2*f(c)) / (b-a)^2 * (x-c)^2 + (f(b)-f(a)) / (b-a) * (x-c) + f(c) )
EndFunc

```

Le reste se passe dans le tableur : on retrouve dans les colonnes A à D la structure habituelle. S'y ajoute :

en E, les abscisses des points de la subdivision, les milieux étant exclus ;

en F, l'appel de la fonction précédente pour récupérer la formule qui donne l'équation de la parabole sur la subdivision considérée ;

en G1, on tape : **=when(e1<x<e2,f1,0)**, pour n'avoir qu'une parabole limitée au petit intervalle sur lequel on travaille ;

enfin, on recopie vers le bas (jusque G30 par exemple, mais il est inutile de mettre trop de cellules) la formule en G1 ;

on peut aussi dans H1 calculer la valeur renvoyée par la méthode de Simpson pour les valeurs choisies de *a*, *b* et *n*, puis mémoriser ce résultat dans une variable **res**.

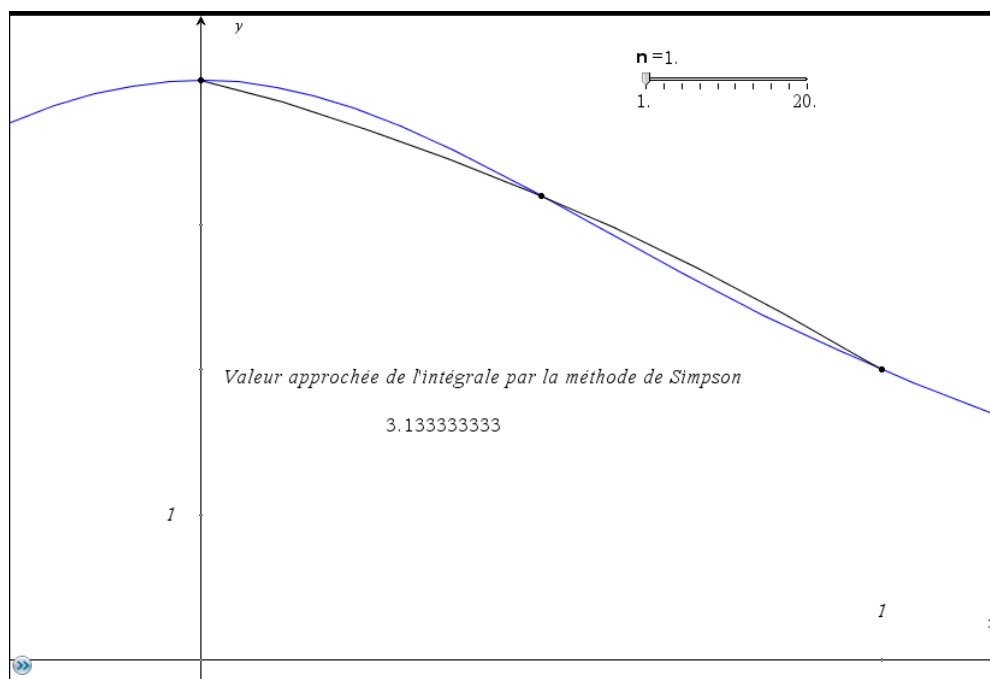
Il reste à nommer **jj** la colonne G, et à demander dans **Graphiques** dans **f2** la représentation graphique de **jj**, (plus exactement **left(jj,n)**), i.e. de chacune des fonctions de la liste **jj**.

A	B	C _{xx}	D _{yy}	E	F _{ff}	G _{jj}	H	I
♦		=x _{sin}	=y _{sims}	=seq('a+k*(b-a)/n,k,0,n)				
1	3	0	0	4	0	-612*x...	piecewise(4...	829597/264069
2		1	0.16...	3.89189...	1/3	-36*x^...	piecewise(5...	
3			0.33...	3.6	2/3	684*x^...	piecewise(3...	
4			0.33...	3.6	1	_	piecewise(_...	
5			0.5	3.2		_	piecewise(_...	
6			0.66...	2.76923...		_	piecewise(_...	
7			0.66...	2.76923...		_	piecewise(_...	
8			0.83...	2.36065...		_	piecewise(_...	
9			1.	2.		_	piecewise(_...	
10						_	piecewise(_...	
11						_	piecewise(_...	
12						_	piecewise(_...	
13						_	piecewise(_...	
14						_	piecewise(_...	
15						_	piecewise(_...	
16						_	piecewise(_...	
17						_	piecewise(_...	
18						_	piecewise(_...	
G1	=when(e1<x<e2,f1)							

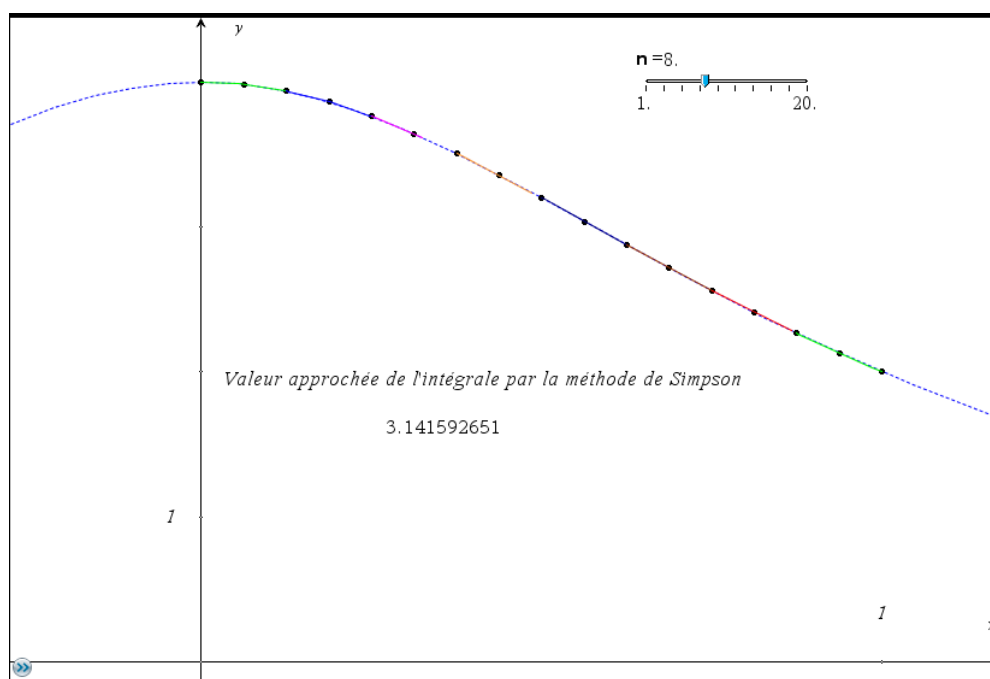
²¹ Plus précisément au paragraphe 5.1. C'est même la seule raison pour laquelle nous l'avons fait : on a vu qu'on n'en a pas besoin pour le calcul des intégrales grâce à la formule des trois niveaux.

Il reste à visualiser la fenêtre graphique. Évidemment, les paraboles deviennent très vite extrêmement proches de la courbe représentative de f – qu'on peut mettre en pointillé pour bien la distinguer des autres.

Comme précédemment, on peut mettre un curseur pour piloter la valeur de n ; la valeur approchée correspondante peut être affichée avec la variable **res**.



Les courbes deviennent peu visibles dès que n est trop grand²². Cette visualisation est intéressante lorsque n prend de toutes petites valeurs, pour montrer la qualité de l'approximation de la courbe représentative de f par des paraboles.



²² Avec parfois quelques petits problèmes d'affichage, mais cela illustre quand même bien la méthode de Simpson.

6. La méthode de Romberg

Cette méthode consiste à appliquer à la méthode des trapèzes le procédé d'accélération de la convergence de Richardson²³. Plutôt que d'augmenter le degré du polynôme d'interpolation, c'est plutôt cette méthode qu'il faut mettre en œuvre lorsque l'on souhaite avoir un résultat précis.

6.1 Développement limité de la méthode des trapèzes

Nous admettrons le résultat suivant²⁴ :

Théorème

Si f est de classe C^{2k} sur un segment $[a ; b]$ et si $T(h)$ désigne l'approximation de $I = \int_a^b f(x)dx$ par la méthode des trapèzes pour un pas égal à $h = \frac{b-a}{n}$, alors on a le développement limité suivant en 0 :

$$T(h) = I + a_1 h^2 + a_2 h^4 + \dots + a_{k-1} h^{2(k-1)} + O(h^{2k})$$

où les coefficients a_i ne dépendent ni de h , ni de n .

6.2 Application à une accélération de convergence

Nous allons nous servir de ce développement pour accélérer la convergence de la méthode des trapèzes.

Le résultat précédent nous permet d'écrire :

$$T(h) = I + a_1 h^2 + a_2 h^4 + \dots + a_{k-1} h^{2(k-1)} + O(h^{2k})$$

La convergence vers I sera plus rapide si l'on parvient par un calcul algébrique à éliminer le terme en h^2 de cette expression. Or, si l'on considère :

$$T\left(\frac{h}{2}\right) = I + a_1 \frac{h^2}{4} + a_2 \frac{h^4}{16} + \dots + a_{k-1} \frac{h^{2(k-1)}}{4^{k-1}} + O(h^{2k})$$

rien n'est plus facile que d'éliminer ce terme en h^2 ! Il suffit de considérer $4T\left(\frac{h}{2}\right) - T(h)$ ce qui donne :

$$4T\left(\frac{h}{2}\right) - T(h) = 3I + O(h^4) \text{ donc } \frac{4T\left(\frac{h}{2}\right) - T(h)}{3} = I + O(h^4)$$

L'approximation de I par $\frac{4T\left(\frac{h}{2}\right) - T(h)}{3}$ est donc meilleure que celle obtenue avec $T(h)$, car l'erreur commise est en $O(h^4)$.

C'est une première accélération, que nous connaissons déjà. Revenons à l'écriture avec n : $T(h)$ n'est autre que T_n et $T\left(\frac{h}{2}\right)$ que T_{2n} . On a donc :

²³ Déjà rencontré pour accélérer la méthode d'Archimède de calcul approché de π .

²⁴ Il se déduit de la formule d'Euler-Mac-Laurin...

$$\frac{4T\left(\frac{h}{2}\right) - T(h)}{3} = \frac{4T_{2n} - T_n}{3}$$

qui n'est autre que le calcul de S_n obtenu par la méthode de Simpson. C'est une accélération convaincante... mais déjà connue. Mais rien n'interdit de poursuivre cette idée.

Introduisons les notations suivantes :

$$\text{pour } m \text{ entier } \geq 0, r_{0,m} = T\left(\frac{b-a}{2^m}\right) \text{ (méthode des trapèzes) ;}$$

$$\text{pour } m \text{ entier } \geq 1, r_{1,m} = \frac{4r_{0,m} - r_{0,m-1}}{3} \text{ (première accélération qui donne la méthode de Simpson).}$$

On peut présenter les résultats dans un tableau : la première colonne est constituée par les résultats de la méthode des trapèzes où l'on a successivement partagé l'intervalle $[a ; b]$ en 1, 2, 4, 8, etc. sous-intervalles ; la seconde est constituée de ce que l'on obtient à la suite de la première accélération.

Nombre de sous-intervalles	1 ^{ère} étape	2 ^e étape	3 ^e étape
1	$r_{0,0}$		
2	$r_{0,1}$	$r_{1,1} = \frac{4r_{0,1} - r_{0,0}}{3}$	
4	$r_{0,2}$	$r_{1,2} = \frac{4r_{0,2} - r_{0,1}}{3}$	$r_{2,2} = \frac{16r_{1,2} - r_{1,1}}{15}$
8	$r_{0,3}$	$r_{1,3} = \frac{4r_{0,3} - r_{0,2}}{3}$	$r_{2,3} = \frac{16r_{1,3} - r_{1,2}}{15}$

Quelle serait l'étape suivante de cette accélération ? Comment fabriquer une troisième colonne sur le même principe ?

Supposons donc que l'on ait obtenu, avec un pas $h = \frac{b-a}{2^{m-1}}$:

$$r_{1,m-1} = I + a'_2 \frac{h^4}{4} + a'_3 \frac{h^6}{16} + O(h^8) \quad (\text{les coefficients ne sont plus les mêmes})$$

Avec un pas de $\frac{h}{2} = \frac{b-a}{2^m}$, on peut écrire :

$$r_{1,m} = I + a'_2 \frac{h^4}{64} + a'_3 \frac{h^6}{1024} + O(h^8).$$

Pour encore accélérer la convergence, fabriquons la colonne des $r_{2,m}$ à partir de la deuxième ligne.

Pour faire disparaître les termes en h^4 , on peut travailler avec $r_{2,m} = \frac{16r_{1,m} - r_{1,m-1}}{15}$ pour $m \geq 2$: d'où la troisième colonne du tableau.

Celle qui vient après serait par un raisonnement analogue définie par :

$$r_{3,m} = \frac{64r_{2,m} - r_{2,m-1}}{63} \text{ pour } m \geq 3.$$

Plus généralement, on aurait $r_{n+1,m} = \frac{4^{n+1}r_{n,m} - r_{n,m-1}}{4^{n+1} - 1}$ pour $m \geq n + 1$.

Quelques précisions : une analyse un peu rapide laisserait croire que l'on obtient des valeurs de plus en plus précises de π en se déplaçant vers la droite dans le tableau. Ce n'est pas forcément vrai et rien n'indique que par exemple que $r_{1,1}$ soit plus précis que $r_{0,0}$ ou $r_{0,1}$. Il faut raisonner globalement : ce sont les suites obtenues dans chaque colonne qui vont avoir une convergence vers π plus rapide. Dans la 1^{ère} colonne, le terme d'erreur est en $O(h^2)$, dans la seconde, le terme est en $O(h^4)$ et plus généralement dans la n^e , en $O(h^{2^n})$. Ceci garantit un comportement *asymptotique* mais ne préjuge en rien de ce qui se passe pour les *premiers termes*.

6.3 Mise en œuvre sur le tableau

Le tableur est particulièrement bien adapté à ce type d'accélération. Dans la colonne A, on peut saisir les puissances de 2, donnant le nombre d'intervalles du découpage.

Dans la colonne B, on renvoie le résultat de la méthode des trapèzes, correspondant à la valeur de n de la colonne A.

Enfin dans les colonnes qui suivent, on saisit les différentes formules d'accélération de la méthode de Romberg (voir par exemple sur l'écran ci-dessous ce qui a été saisi en C2).

A	B	C	D	E	F	G	H	I	J	K
◆ =seq(2^k,k,0,8)										
1	1	3								
2	2	31/10	47/15							
3	4	5323/...	8011/25...	6677/21...						
4	8	1018...	1529166...	127430...	4014052...					
5	16	1720...	2582019...	430336...	6777800...	288056...				
6	32	1672...	1930355...	209121...	6587339...	559923...	422420...			
7	64	1901...	3708076...	181768...	9733701...	318217...	792503...	1959...		
8	128	3194...	7854627...	399276...	1257722...	534531...	911376...	3291...	26958...	
9	256	5097...	9557751...	796479...	2508909...	106628...	265404...	6565...	53777...	58738...
10										
11										
12										
13										
14										
15										
16										
17										
C2	= (4*B2-B1)/3									

Si l'on veut des valeurs approchées, pour se rendre compte de la qualité de la convergence, il faut mettre **approx** dans la colonne A.

A	B	C	D	E	F	
◆ =approx(seq(2^k						
1	1.	3.				
2	2.	3.1	3.1333333333			
3	4.	3.13117647059	3.14156862745	3.14211764706		
4	8.	3.13898849449	3.14159250246	3.14159409413	3.14158578376	
5	16.	3.14094161204	3.14159265123	3.14159266114	3.1415926384	3.14159266528
6	32.	3.14142989318	3.14159265355	3.14159265371	3.14159265359	3.14159265365
7	64.	3.14155196349	3.14159265359	3.14159265359	3.14159265359	3.14159265359
8	128.	3.14158248106	3.14159265359	3.14159265359	3.14159265359	3.14159265359
9	256.	3.14159011046	3.14159265359	3.14159265359	3.14159265359	3.14159265359

Il faut cependant maintenant récupérer des valeurs approchées avec plus de décimales : c'est le moent de ressortir **divex2** de la bibliothèque **division_dec**.

Au préalable, mémorisons tous les résultats de la ligne 9 correspondant aux résultats fractionnaires dans des variables **res**, **res1**, ..., **res8** de droite à gauche.

Puis appliquons **divex2** en demandant l'affichage de 50 décimales :

<code>division_dec\divex2(res8,50)</code>	"3.14159011045828282884501179281464213560211181910808"
<code>division_dec\divex2(res7,50)</code>	"3.14159265358979309748194210990417347709998563904502"
<code>division_dec\divex2(res6,50)</code>	"3.14159265358979368960087041314970197287392886314454"
<code>division_dec\divex2(res5,50)</code>	"3.14159265358979323846291610144678046168535384809856"
<code>division_dec\divex2(res4,50)</code>	"3.14159265358979323846182206694862516079038709675044"
<code>division_dec\divex2(res3,50)</code>	"3.14159265358979323846264264995645113036799829563147"
<code>division_dec\divex2(res2,50)</code>	"3.14159265358979323846269594900021657377303207358131"
<code>division_dec\divex2(res1,50)</code>	"3.14159265358979323846182820332931421421281748655551"
<code>division_dec\divex2(res,50)</code>	"3.14159265358979323846154516251878467223786325510161"
<code>m</code>	"3,141592653589793238462643383279502884197169399375105"

et en les comparant à une bonne valeur approchée de π affichée sur la dernière ligne de calcul. On constate alors que **res3** donne la valeur la plus intéressante, avec 23 décimales exactes.

Concrètement, on aurait pu la trouver en calculant la différence entre deux valeurs successives de la dernière ligne jusqu'à ce que cette différence soit la plus petite possible :

C	D	E	F	G	H	I	J
◆							
1							
2	47/15						
3	8011/2550	6677/2125					
4	152916620159/4...	127430581361/...	4014052694591...				
5	25820192594299...	430336544596...	6777800528327...	28805652491867...			
6	19303557773257...	209121875887...	6587339090202...	55992382267779...	422420...		
7	37080768186778...	181768471503...	9733701649031...	31821716929525...	792503...	1959...	
8	78546270001715...	399276872508...	1257722148402...	53453191307105...	911376...	3291...	26958...
9	95577511562715...	796479263022...	2508909678521...	10662866133715...	265404...	6565...	53777... 58738...
10	37741643177419...	900709607461...	-181503084104...	-2875151213136...	452253...	5879...	-85833... -46591...
11	0.000002543132	5.92118928303...	-4.51137954312...	-1.09403449816...	8.20583...	5.32...	-8.677... -2.830...
12				8.2058300782597E-22			
13							
14							
15							
16							
17							
18							
G11	=approx($\varepsilon 10$)						

Évidemment, la méthode est empirique mais elle autorise un critère de choix, qui donne ici la bonne réponse. Dans notre exemple, cela correspond à la valeur située en G11 et donc à la variable **res3** de la cellule G9 de la feuille de calcul suivante :

Bref, on peut en déduire que : $\pi \approx 3,14159265358979323846264$, soit une valeur approchée avec 23 décimales.