TP
L1 MI

Le but de ce TP est de comparer plusieurs méthodes d'intégration numérique.

Le calcul approché des intégrales est un problème important en analyse numérique. Soit f une fonction uniformément continue dans un intervalle [a,b]. On cherche à déterminer une valeur approchée de:

$$S = \int_a^b f(x)dx.$$

La valeur de S est l'aire de la surface délimitée par la courbe de la fonction f, de l'axe des x, et des verticales d'abscisses respectivement a et b. (voir fig 1).

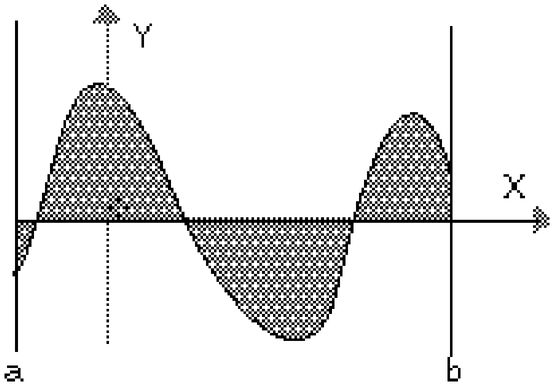


fig 1

Une technique qui permet d'avoir une approximation de S est de remplacer la fonction par son polynôme d'interpolation de degré n. Cette technique présente des inconvénients aussi bien théoriques (dans certain cas il y a non convergence quand n tend vers l'infini) que pratiques (instabilité des calculs). Dans les premières méthodes qu'on va voir, on décompose l'intervalle [a,b] en n sous intervalles de tailles égales. La taille de chaque intervalle est donnée par h=(b-a)/n. Les n intervalles sont donnés par $x_0=a$, x_1 , ..., $x_n=b$. Pour tout i entre 0 et n-1 nous avons donc $h = x_{i+1}-x_i$ ou encore $x_{i+1} = x_i + h$.(voir fig 2).

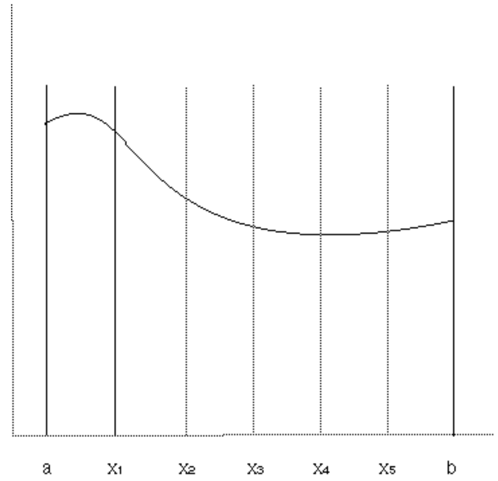


fig 2

Nous avons donc $S = \int_a^b f(x)dx = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} f(x)dx$. Il reste à déterminer une approximation de $S_i = \int_{x_i}^{x_{i+1}} f(x)dx$.

1.- La méthode des rectangles.

Dans cette méthode, on approche S_i par l'aire du rectangle de largeur h et de longueur $f(x_i)$ (voir fig 3).

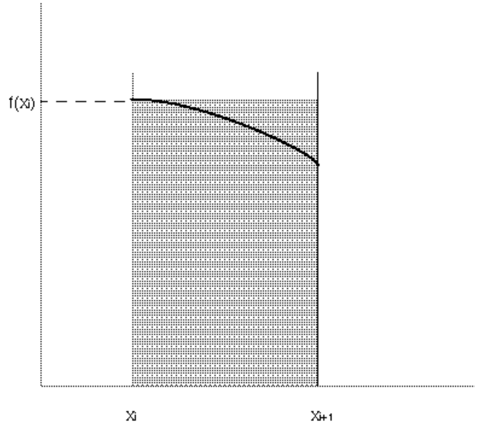


fig 3

$S_i = \int_{x_i}^{x_{i+1}} f(x)dx \approx h.f(x_i)$. Par conséquent $S \approx h.\sum_{i=0}^{n-1} f(x_i)$ avec $h = (b-a)/n$, $x_{i+1} = x_i + h$ et $x_0 = a$.

2.- La méthode des trapèzes.

Dans cette méthode, on interpole la courbe dans l'intervalle $[x_i,x_{i+1}]$ par un polynôme qui passe par 2 points $(x_i,f(x_i))$ et $(x_{i+1},f(x_{i+1}))$. Ce polynôme est donc une droite (voir fig 4).

S_i est alors approché par l'aire du trapèze comme indiqué dans la figure 4. Cette aire est donnée par la petite base plus la grande base multiplié par la hauteur divisé par 2 c'est à dire :

$S_i \approx (f(x_{i+1})+f(x_i)).h/2$. Ce qui nous donne : $S \approx \frac{h}{2}\sum_{i=0}^{n-1}(f(x_i)+f(x_{i+1}))=$

$$\frac{h}{2}\left(\sum_{i=0}^{n-1}f(x_i)+\sum_{i=0}^{n-1}f(x_{i+1})\right)=\frac{h}{2}\left(f(a)+2.\sum_{i=1}^{n-1}f(x_i)+f(b)\right)=h\left(\frac{f(a)+f(b)}{2}+\sum_{i=1}^{n-1}f(x_i)\right)$$

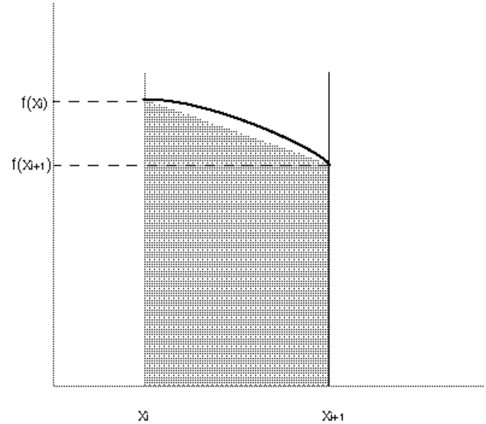


fig 4

3.- La méthode de la tangente.

Cette méthode consiste à prendre la tangente à la courbe en $(x_i+h/2,f(x_i+h/2))$. Cette tangente va couper la verticale d'abscisse x_i en y_i et la verticale d'abscisse x_{i+1} en y_{i+1} . Ceci nous donne également un trapèze (voir fig 5).

L'aire de ce trapèze est donnée par $A_i = (y_i +y_{i+1})h/2$.

L'équation de la tangente est $y = f'(x_i+h/2)x + b$.

Ceci nous donne: $y_i=f'(x_i+h/2)x_i+b$ de même $y_{i+1}=f'(x_i+h/2)x_{i+1}+b$ et donc: $A_i =(f'(x_i+h/2)(x_i + x_{i+1}) + 2b).h/2 = (f'(x_i+h/2)(2x_i + h)+ 2b).h/2$ car $x_{i+1}= x_i +h$. $A_i =(f'(x_i+h/2)(x_i + h/2) + b).h$. (1)

Or le point $(x_i+h/2, f(x_i+h/2))$ appartient également à la tangente et par conséquent: $f(x_i+h/2)=f'(x_i+h/2)(x_i+h/2)+b$. En remplaçant dans (1) on obtient: $A_i = h.f(x_i+h/2)$. Ce qui nous donne :

$$S \approx h.\sum_{i=0}^{n-1} f(x_i + \frac{h}{2})$$

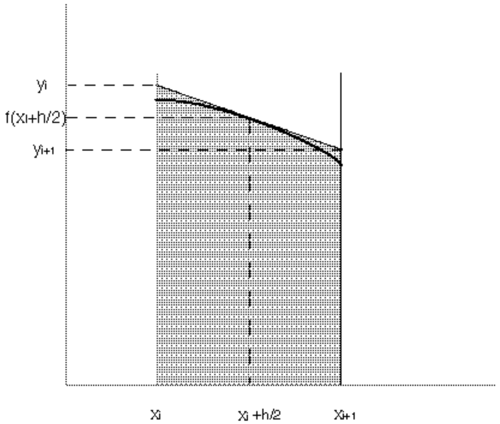


fig 5

4.- La méthode de Simpson

La méthode de Simpson consiste à décomposer l'intervalle $[a,b]$ en $2n$ intervalles $[a,x_1]$, $[x_1,x_2]$, ... ,$[x_{2n-1},b]$. Dans 2 intervalles successifs on interpole la courbe par un polynôme de degré 2. On obtient la formule globale suivante :

$$S \approx (\frac{b-a}{6n})(f(a)+f(b)+4\sum_{i=1}^{n} f(x_{2i-1})+2\sum_{i=1}^{n-1} f(x_{2i}))$$

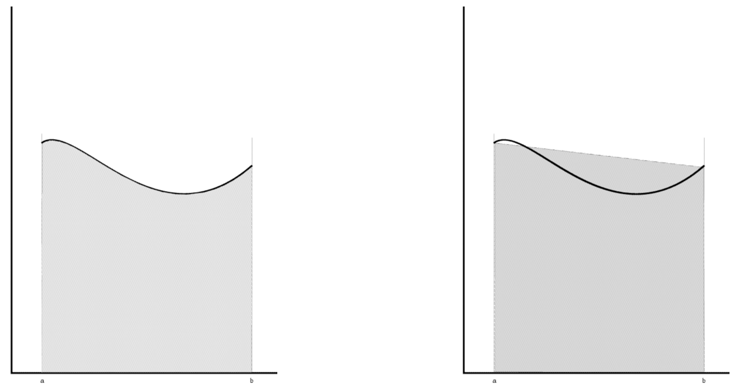
Les méthodes adaptatives.

Dans les méthodes adaptatives, on ne choisit pas d'emblée le nombre de sous-interrvalles. On part de l'intervalle de départ et on le décompose successivement en 2, 4, ... 2^n sous-intervalles. On construit ainsi une suite (I_n) pour $n=0,1,$ ...qui converge vers la valeur de l'intégrale. On définit ensuite l'expression de I_{n+1} en fonction I_n . Prenons $h_n= \frac{b-a}{2^n}$.

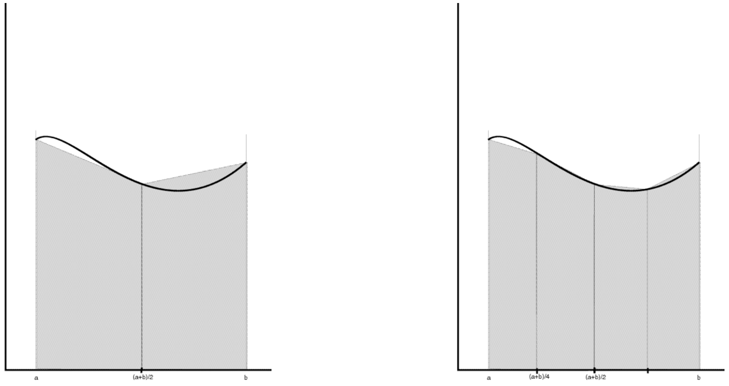
La méthode des trapèzes adaptative est donnée par la formule :

$$I_{n+1} = \frac{I_n}{2} + h_{n+1}\sum_{p=1}^{2^n} f(a+(2p-1)h_{n+1})$$

avec $I_0 = \frac{b-a}{2}[f(a)+ f(b)]$ (voir figure 6 ci-dessous)



I_∞ $\qquad$ I_0



I_1 I_2

fig 6

La méthode de Simpson adaptative est donnée par la formule :

$$I_{n+1} = \frac{I_n}{2} + \frac{2}{3} h_n \sum_{p=0}^{2^n - 1} f(a + (2p+1)h_{n+1}) - \frac{1}{3} h_n \sum_{p=0}^{2^{n-1} - 1} f(a + (2p+1)h_n)$$

avec $I_1 = \frac{b - a}{6}[f(a)+4f(\frac{a+b}{2})+f(b)]$

Programmation

Le TP consiste à écrire un premier programme qui :

- Déclare une fonction f (par exemple $f(x)= \frac{1}{x+1}$). On comparera le résultat de chaque méthode avec la valeur exacte. En prenant par exemple a=0 et b=1, la valeur exacte est $S = \int_a^b f(x)dx = Log(2)$.
- Définit pour chacune des 4 méthodes décrites ci-dessus, la fonction qui permet de donner une approximation de l'intégrale de la fonction dans un intervalle donné et pour une décomposition donnée :

```
double Irectangle(double a, double b, int n)
double Itrapeze(double a, double b, int n)
double Itangente(double a, double b, int n)
double Isimpson(double a, double b, int n)
```

Dans le programme principal, on affichera les valeurs pour chacune des méthodes ainsi que la valeur exacte.

Vous écrirez un deuxième programme qui compare les deux méthodes adaptatives, celle des trapèzes et celle de Simpson en choisissant, bien sûr la même fonction que ci-dessus, le même intervalle [0,1] et une même valeur epsilon. Vous écrirez une procédure pour chaque méthode. Dans chacun des cas, vous arrêterez le programme dès que $|I_{i+1}-I_i| < \varepsilon$ (avec ε donné). Vous mettrez les valeurs successives de I_n dans un tableau de taille MAX. Bien sûr, si la condition ci-dessus n'est pas satisfaite au bout de MAX itérations vous arrêterez le programme. Vous utiliserez les déclarations suivantes :

const int MAX = 1000;

typedef double tableau[MAX];

Les entêtes des deux procédures sont alors :

void Itrap_Adapt(double a, double b, double eps,int &n, tableau t)

void Isim_Adapt(double a, double b, double eps,int &m, tableau s)

n et m contiennent à la sortie le premier i tel que $|I_{i+1}-I_i| < \varepsilon$. i est inférieur à MAX.

Le programme principal saisit a, b et epsilon. Il appelle les deux procédures ***Itrap_Adap*** et ***Isim_Adap*** pour remplir les tableaux t et s. Ensuite, il affiche sur une première colonne le tableau t, sur une deuxième colonne l'erreur commise par rapport à la valeur exacte sur une troisième colonne le tableau s et sur une quatrième colonne l'erreur commise par rapport à la valeur exacte.