

TP4 – Intégration numérique Éléments de correction

- Programmer la méthode composée de Gauss-Legendre pour approcher une intégrale de la forme $\int_a^b f(x)dx$, utilisant une subdivision régulière de l'intervalle $[a, b]$.

Pour mémoire : quadrature élémentaire de Gauss-Legendre :

$$\int_0^1 \varphi(x)dx \simeq \frac{1}{18} \left[5\varphi\left(\frac{1}{2} - \frac{\sqrt{15}}{10}\right) + 8\varphi\left(\frac{1}{2}\right) + 5\varphi\left(\frac{1}{2} + \frac{\sqrt{15}}{10}\right) \right].$$

Fichier methodes.sci - Fonction GaussLegendre

```

1 function I=GaussLegendre(f,a,b,N)
2     h = (b-a)/N;
3     T = a+(0:N-1)*h;
4     X1 = T + (1/2-sqrt(15)/10)*h;
5     X2 = T + h/2;
6     X3 = T + (1/2+sqrt(15)/10)*h;
7     F = 1/18*[5*f(X1)+8*f(X2)+5*f(X3)];
8     I = h*sum(F);
9 endfunction
    
```

- Programmer de même la méthode de Simpson.

Quadrature élémentaire de Simpson

$$\int_0^1 \varphi(x)dx \simeq \frac{1}{6} [\varphi(0) + 4\varphi\left(\frac{1}{2}\right) + \varphi(1)].$$

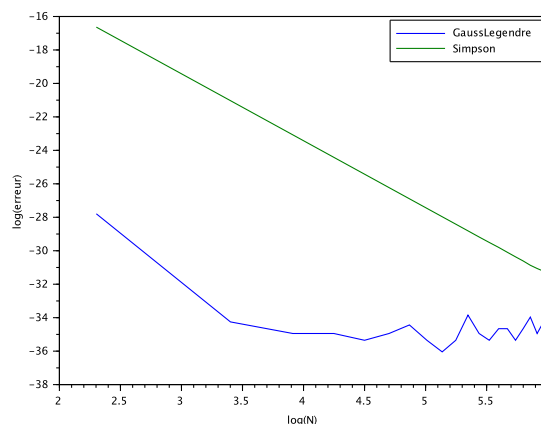
Fichier methodes.sci - Fonction Simpson

```

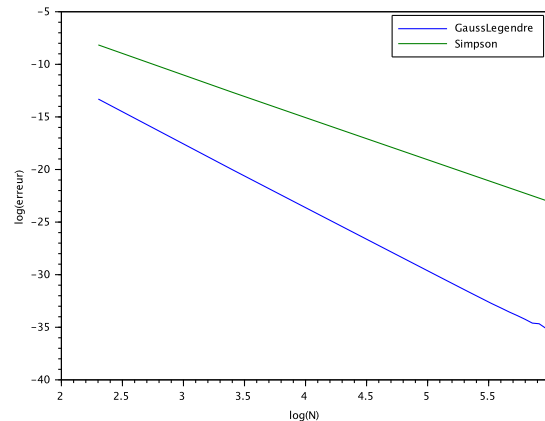
1 function I=Simpson(f,a,b,N)
2     h = (b-a)/N;
3     T = a+(0:N-1)*h;
4     X1 = T ;
5     X2 = T + h/2;
6     X3 = T + h;
7     F = 1/6*[f(X1)+4*f(X2)+f(X3)];
8     I = h*sum(F);
9 endfunction
    
```

1. Comparer les deux méthodes précédentes en termes de performance (ordre de convergence) et de coût de calcul (nombre d'évaluations de la fonction). On considérera une fonction $f \in \mathcal{C}^4([0, 1])$.

On teste les deux méthodes pour approcher $\int_0^1 e^x dx$. La figure représente le logarithme de l'erreur en fonction du logarithme du nombre de sous-intervalles de $[0, 1]$. Les courbes de convergence saturent à l'ordonnée de l'ordre du logarithme de l'erreur machine : -36.



On peut retarder l'influence de l'erreur machine en relevant les courbes de convergence. Il suffit pour cela de considérer une fonction plus difficile à intégrer, par exemple une fonction assez oscillante : $\int_0^1 \cos(20x) dx$.



Pour ce test, la regression linéaire sur ces calculs donne un ordre de convergence effectif de 5.9504269 dans le cas de la méthode de Gauss, et 4.0214371 dans celui de la méthode de Simpson. La théorie prévoit respectivement des ordres de 6 et de 4, sous réserve de régularité suffisante pour les fonctions intégrées. Concernant le coût de calcul, les programmes proposés ne sont pas optimisés et les deux méthodes requièrent chacune $3N$ évaluations de la fonction intégrée. Le coût de la méthode de Simpson peut être légèrement réduit en programmant la forme équivalente :

$$\int_a^b f(x) dx \simeq \frac{h}{6} \left[f(a) + 2 \sum_{k=1}^{N-1} f(a + kh) + 4 \sum_{k=0}^{N-1} f(a + (k + \frac{1}{2})h) + f(b) \right].$$

De cette manière il faut seulement $2N+1$ évaluations de la fonction intégrée. Néanmoins, cette réduction du coût ne suffit pas à compenser la différence sur l'ordre de la méthode. Dans l'idée, il existe deux constantes C_{GL} et C_S ne dépendant que de la méthode et de la fonction intégrée, telle que l'erreur pour N sous-intervalles soit de l'ordre de $C_{GL}N^{-6}$ pour Gauss-Legendre et $C_S N^{-4}$ pour Simpson. Supposons que chaque évaluation de f prenne 1 seconde et que le reste des calculs soit de durée négligeable, pour atteindre une précision de ϵ il faudra un temps de l'ordre de $3(C_{GL}/\epsilon)^{1/6}$ par Gauss-Legendre et de l'ordre de $2(C_S/\epsilon)^{1/4}$ par Simpson. À une constante multiplicative près qui ne dépend que des données (f , $[a, b]$ et des constantes universelles liées aux méthodes utilisées), le temps de calcul T_S pour Simpson augmente donc comme $T_{GL}^{3/2}$ lorsque la précision demandée ϵ tend vers 0.

Fichier question1.sce

```

1 clear;
2 exec methodes.sci;
3
4 function y=f(x)
5     y=exp(x);
6 endfunction
7 a=0; b=1;
8 Iex = exp(b)-exp(a);
9
10 ListeN = [10:20:400];
11 I_GaussLeg = []; I_Simpson = [];
12 for N=ListeN
13     I_GaussLeg=[I_GaussLeg, GaussLegendre(f,a,b,N)];
14     I_Simpson=[I_Simpson, Simpson(f,a,b,N)];
15 end
16 Erreur_GaussLeg = abs(I_GaussLeg-Iex)+%eps;
17 Erreur_Simpson = abs(I_Simpson-Iex)+%eps;
18
19 scf(1);clf();
20 plot(log(ListeN),log(Erreur_GaussLeg),log(ListeN),log(Erreur_Simpson));
21 legend('GaussLegendre','Simpson'); xlabel('log(N)'); ylabel('log(erreur)');
22
23

```

```

24 function y=g(x)
25     y=cos(20*x);
26 endfunction
27 a=0; b=1;
28 Iex = 1/20*(sin(20*b)-sin(20*a));
29
30 ListeN = [10:20:400];
31 I_GaussLeg = []; I_Simpson = [];
32 for N=ListeN
33     I_GaussLeg=[I_GaussLeg,GaussLegendre(g,a,b,N)];
34     I_Simpson=[I_Simpson,Simpson(g,a,b,N)];
35 end
36 Erreur_GaussLeg = abs(I_GaussLeg-Iex)+%eps;
37 Erreur_Simpson = abs(I_Simpson-Iex)+%eps;
38
39 scf(2);clf();
40 plot(log(ListeN),log(Erreur_GaussLeg),log(ListeN),log(Erreur_Simpson));
41 legend('GaussLegendre','Simpson'); xlabel('log(N)'); ylabel('log(erreur)');
42
43 [aGL,b] = reglin(log(ListeN),log(Erreur_GaussLeg));
44 [aSi,b] = reglin(log(ListeN),log(Erreur_Simpson));

```

□ Programmer la méthode du point milieu sur $[a, b]$ utilisant une subdivision régulière.

Fichier methodes.sci - Fonction PointMilieu

```

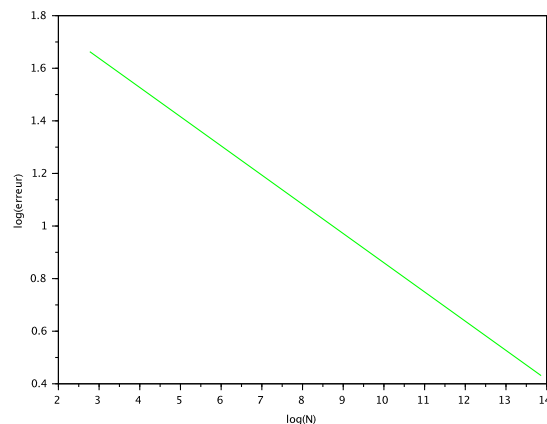
1 function I=PointMilieu(f,a,b,N)
2     h = (b-a)/N;
3     T = a+(0:N-1)*h;
4     X = T + h/2;
5     F = [f(X)];
6     I = h*sum(F);
7 endfunction

```

2. Par la méthode du point milieu, approcher l'intégrale suivante

$$\int_0^1 \frac{e^{\sin(x)}}{x^{8/9}} dx \simeq 10.09485330. \quad (1)$$

On pourra examiner la vitesse de convergence en fonction du nombre de points de la subdivision et analyser les observations.



L'ordre obtenu par regression linéaire est de $|0.1110968|$. Les valeurs obtenues pour l'intégrale approchée convergent très lentement vers la valeur exacte, pour les différentes subdivisions considérées, on obtient les valeurs approchées suivantes :

Sortie console

```

1 >>Iapp'
2 ans =
3     4.819315
4     5.2095967
5     5.5713888

```

```

6      5.9065463
7      6.21694
8      6.5043613
9      6.7704938
10     7.0169065
11     7.2450573
12     7.4562978
13     7.6518809
14     7.8329666
15     8.0006295
16     8.1558644
17     8.2995925
18     8.4326667
19     8.5558767

```

Fichier question2.sce

```

1 clear;
2 exec methodes.sci;
3
4 function y=f(x)
5     y = exp(sin(x))./(x.^(8/9));
6 endfunction
7 Iex = 10.09485330;
8
9 // methode du point milieu :
10 Liste_N = 2^(4:20);
11 Iapp = [];
12 for N = Liste_N
13     Iapp = [Iapp, PointMilieu(f,0,1,N)];
14 end
15 Erreur = abs(Iapp - Iex);
16 scf(3);clf;
17 plot(log(Liste_N), log(Erreur), 'g');
18 [a,b]=reglin(log(Liste_N), log(Erreur));
19 xlabel('log(N)'); ylabel('log(erreur)');

```

Quelques précisions sur ces observations : la formule d'erreur de la quadrature élémentaire du point milieu, dans le cas d'une fonction $f \in \mathcal{C}^2([a, b])$ prend la forme

$$\frac{(b-a)^3}{24} f''(\xi), \quad \xi \in]a, b[.$$

Ici la dérivée seconde de l'intégrande $\frac{e^{\sin(x)}}{x^{8/9}}$ prend la forme $\frac{\alpha(x)}{x^{26/9}}$ avec α une fonction régulière sur le fermé $[0, 1]$ non-nulle en 0. En considérant l'erreur globale pour la quadrature composée pour n sous-intervalles de taille $h = 1/n$, on obtient donc en sommant les erreurs locales sur les sous-intervalles constituant $[h, 1]$:

$$\sum_{i=1}^{n-1} \frac{h^3}{24} \frac{\alpha(\xi_i)}{\xi_i^{26/9}}, \quad \xi_i \in]ih, (i+1)h[.$$

On peut majorer la valeur absolue de cette somme par

$$\frac{h^3}{24h^{26/9}} \|\alpha\|_{\infty} \sum_{i=1}^{n-1} \frac{1}{i^{26/9}} \leq \frac{\|\alpha\|_{\infty}}{24} h^{1/9} \sum_{i=1}^{+\infty} \frac{1}{i^{26/9}}$$

où la série converge (on peut penser ici à faire le lien entre l'intégrabilité de f , les comparaisons série/intégrale, etc.) L'erreur commise sur le premier intervalle $[0, h]$, quant à elle, est liée au comportement de la quantité

$$\int_0^h \frac{1}{x^{8/9}} dx - \frac{h}{(h/2)^{8/9}},$$

Chacun des deux termes se comporte à des constantes multiplicatives près comme $h^{1/9}$. L'erreur globale pour la quadrature sur $]0, 1]$ est donc $O(h^{1/9})$, où $1/9 \simeq 0.111111$ est bien l'ordre observé.

Un peu d'analyse. On se donne $h \in]0, 1[$ fixé. Proposer une méthode gaussienne à un point, exacte pour tout $p \in \mathbb{R}_1[X]$ dans l'évaluation de

$$\int_0^h \frac{p(x)}{x^{8/9}} dx.$$

Il s'agit d'identifier une quadrature exacte sur $\mathbb{R}_1[X]$ destinée à l'évaluation d'intégrales de la forme précédente. On peut pour cela identifier les premiers polynômes orthogonaux pour le poids $x^{-8/9}$ sur $]0, 1]$, ou bien procéder par identification des paramètres de la quadrature à un point.

$$\int_0^h \frac{ax + b}{x^{8/9}} dx = a \frac{9}{10} h^{10/9} + 9bh^{1/9}.$$

La quadrature correspondante sera de la forme $Q(p) = \omega(ax_0 + b)$ et sera exacte sur $\mathbb{R}_1[X]$ si et seulement si $\omega x_0 = \frac{9}{10} h^{10/9}$ et $\omega = 9h^{1/9}$, c'est à dire si et seulement si $\omega = 9h^{1/9}$ et $x_0 = h/10$.

Ainsi la quadrature retenue est simplement :

$$\int_0^h \frac{f(x)}{x^{8/9}} dx \simeq 9h^{1/9} f\left(\frac{h}{10}\right).$$

3. Approcher l'intégrale (??) par une méthode hybride utilisant la quadrature gaussienne obtenue sur $]0, h]$ et la quadrature composée du point milieu à n points sur $[h, 1]$. On pourra ensuite choisir h en fonction de n de manière à optimiser la vitesse de convergence.

La formule considérée prend la forme suivante, dans laquelle on a posé $x_i = h + i \frac{1-h}{n}$ une subdivision régulière de $[h, 1]$, et $x_{i+1/2} = \frac{1}{2}(x_i + x_{i+1})$:

$$\int_0^1 \frac{f(x)}{x^{8/9}} dx = \int_0^h \frac{f(x)}{x^{8/9}} dx + \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} \frac{f(x)}{x^{8/9}} dx \simeq 9h^{1/9} f\left(\frac{h}{10}\right) + \frac{1-h}{n} \sum_{i=0}^{n-1} \frac{f(x_{i+1/2})}{x_{i+1/2}^{8/9}}.$$

Fichier question3.sce

```

1 clear;
2 stacksize('max');
3
4 function y=f(x)
5     y = exp(sin(x))./(x.^(8/9));
6 endfunction
7 I = 10.09485330
8
9 // methode mixte Gauss - point milieu :
10 scf(9);clf();
11 H=0.2*[0.8].^(1:15);
12 ERRH=[];
13 for i=1:length(H)
14     N = 2^(4:20);
15     Err2 = [];
16     for n = N
17         h = H(i);
18         X = linspace(h,1,n);
19         X = 0.5*(X(1:n-1)+X(2:n));
20         Y = f(X);
21         dx = X(2)-X(1);
22         Iapp = sum(dx*Y); // point milieu sur [h,1]
23         Iapp = Iapp + 9*h^(1/9)*exp(sin(h/10)); // gauss P_1-exacte sur [0,h]
24         Err2 = [Err2,abs(Iapp-I)];
25     end
26     subplot(2,1,1);
27     plot(log(N),log(Err2),'r');
28     ERRH=[ERRH,Err2($)];
29 end
30 title('Courbes de convergence en fonction de n pour differents h'); xlabel('log(n)');
    ylabel('log(terr)');
31 subplot(2,1,2);
32 plot(log(H),log(ERRH));
33 [a,b]=reglin(log(H($-3:$)),log(ERRH($-3:$)));

```

```

34 | title('erreur limite en fonction de h'); xlabel('log(h)'); ylabel('log(err_min)');
35 | xstring(-3, -12, string(a))

```

Précisions sur l'optimisation des paramètres

Adaptons l'analyse précédemment menée pour la méthode du point milieu, mais sur $[h, 1]$ seulement (le pas d'intégration est alors $(1-h)/n$). L'erreur de quadrature pour l'approximation de la somme des n termes s'écrit :

$$\left| \sum_{i=0}^{n-1} \left(\int_{x_i}^{x_{i+1}} \frac{f(x)}{x^{8/9}} dx - \frac{1-h}{n} \frac{f(x_{i+1/2})}{x_{i+1/2}^{8/9}} \right) \right| \leq \sum_{i=0}^{n-1} \frac{(1-h)^3}{24n^3} \|\alpha\|_{\infty} \frac{1}{\xi_i^{26/9}}, \quad \text{où } \xi_i \in [x_i, x_{i+1}],$$

$$\leq \frac{\|\alpha\|_{\infty}}{24n^3} \sum_{i=0}^{n-1} \frac{1}{x_i^{26/9}}.$$

Quelle que soit la valeur de $h \in]0, 1[$ et de $n \in \mathbb{N}^*$, la série partielle qui intervient prend aussi la forme

$$\sum_{i=0}^{n-1} \frac{1}{\left(h + \frac{i(1-h)}{n}\right)^{26/9}} = \frac{1}{h^{26/9}} \sum_{i=0}^{n-1} \frac{1}{1 + \frac{i(1-h)}{nh}} \leq \frac{n}{h^{26/9}}.$$

L'erreur pour les termes considérés est donc majorée par une quantité de la forme

$$\frac{C_1}{n^2 h^{26/9}},$$

où $C_1 > 0$ est une constante indépendante de h et de n .

Concernant l'intégration gaussienne sur $[0, h]$, l'erreur commise est liée aux premiers termes non-nuls du développement de Taylor du numérateur $f(x) = e^{\sin(x)}$ en $h/10$: $f(x) = P_1(x) + \frac{1}{2}(x - h/10)^2 \phi(x)$ avec $P_1(x) = f(h/10) + (x - h/10)f'(h/10)$ et ϕ régulière.

$$\int_0^h \frac{f(x)}{x^{8/9}} dx - 9h^{1/9} f\left(\frac{h}{10}\right) = \int_0^h \frac{P_1(x)}{x^{8/9}} dx - 9h^{1/9} P_1\left(\frac{h}{10}\right) + \int_0^h \frac{\frac{1}{2}(x - h/10)^2 \phi(x)}{x^{8/9}} dx$$

La formule étant exacte pour les polynômes de degré 1, la première différence du second membre est nulle. Le terme restant se majore (en valeur absolue) par :

$$\frac{\|\phi\|_{\infty}}{2} \int_0^1 \frac{(hy - h/10)^2}{(hy)^{8/9}} h dy = \frac{\|\phi\|_{\infty}}{2} h^{19/9} \int_0^1 \frac{(y - 1/10)^2}{(y)^{8/9}} dy.$$

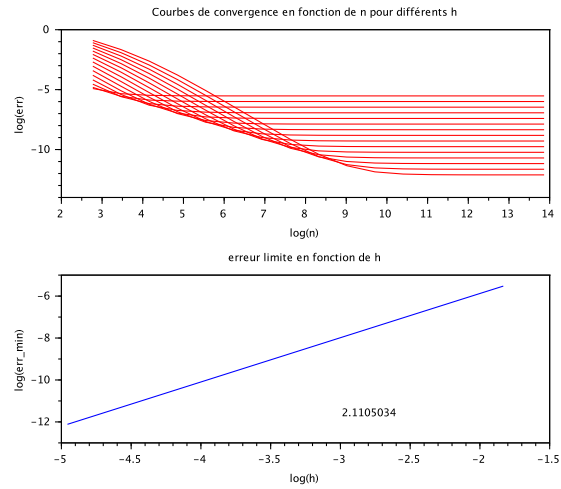
Il existe donc une constante C_2 indépendante de h et de n telle que l'erreur sur cette quadrature soit majorée par une quantité de la forme

$$C_2 h^{19/9}.$$

Les deux majorations précédemment obtenues permettent de dégager un régime optimal \ 'a la limite pour choisir h en fonction de n . Supposant que les majorations sont optimales (au sens où l'erreur est équivalente à son majorant, ce qui n'a pas été prouvé) l'erreur globale se comporte comme

$$\frac{C_1}{n^2 h^{26/9}} + C_2 h^{19/9}.$$

L'expérience numérique confirme ce comportement, on effectue l'analyse de convergence pour différentes valeurs de h , de sorte à s'eparer le comportement des deux termes. On observe la saturation de l'erreur sur une quantité qui n'est autre que l'erreur de la quadrature gaussienne. On peut ensuite tracer en fonction de h cette valeur et vérifier sa convergence vers 0 comme $h^{19/9}$.



Pour minimiser l'erreur globale on peut choisir h en fonction de n de sorte que les deux erreurs soient du même ordre :

$$\frac{C_1}{n^2 h^{26/9}} \sim C_2 h^{19/9}$$

En réalité, les constantes sont importantes car elles définissent une constante C telle que le choix optimal ait la forme

$$h = \frac{C}{n^{2/5}},$$

et l'erreur se comporte comme finalement

$$\frac{\tilde{C}}{n^{4/5}}.$$