Analyse Numérique

Corrigés des Travaux Pratiques 2024 – 2025
Séance 9

Les méthodes des trapèzes et de Simpson utilisées ici sont mises au point dans les fonctions `int_trp` et `int_smp` dont le code est en annexe. Les deux fonctions ont la forme

```
int = itg_xxx(f,a,b,ni);
```

où `f` est la fonction à intégrer, `a`, `b` définissent l'intervalle et `ni` donne le nombre de sous-intervalles. Il va de soi qu'il ne s'agit que d'une implémentation possible, et vous êtes libres d'en choisir une autre.

1. Une possible vérification est comme suit :

```
clear; close;
for j = 0:3
        % ici on intègre la fonction x^j, j=1,2,3
        % sur [0,1]; le résultat doit être 1/(j+1)
    f = @(x)(x^j);
    int = (f(0)+4*f(1/2)+f(1))/6;
    err =  abs( int - 1/(j+1) )
end
```

A priori, une vérification sur un seul intervalle d'intégration est suffisante (pourquoi ?). De plus la longueur et la position de cet intervalle sont peu importantes, car la formule de Simpson est invariante par translation et linéaire en f .

Notons qu'une vérification similaire est utile pour tester des programmes d'intégration plus généraux (y compris ceux que vous avez codés pour cette séance). A titre d'exemple on peut tester la fonction `itg_smp` en annexe qui correspond à la méthode de Simpson (avec `ni=5` sous-intervalles) :

```
clear; close;
ni = 5;     % nombre d'intervalles
for j = 0:3
        % ici on intègre la fonction x^j, j=1,2,3
    f = @(x)(x^j);
        % sur [0,1] le résultat doit être 1/(j+1)
    int = itg_smp(f,0,1,ni);
    err =  abs( int - 1/(j+1) )
end
```

2. Pour les intégrales de la forme

$$\int_a^b f(x)dx$$

l'erreur absolue d'intégration pour la méthode des trapèzes est donnée (avec $c \in]a,b[$) par

$$E_{\text{glob}}(h) \ = \ -\frac{1}{12}(b-a)h^2 f''(c)\,,$$

	trapèzes	Simpson	$\vert$trapèzes − Simpson$\vert$
a)	0.40600608	0.40600585	2.3e-07
b)	0.99532225	0.99532227	1.8e-08
c)	1.85193674	1.85193705	3.1e-07

TABLE 1 – Les intégrales a)-c) évaluées par les méthodes des trapèzes et de Simpson, ainsi que leur différence.

et donc on peut garantir que $|E_{\text{glob}}(h)| \leq 10^{-6}$ si l'on choisit

$$h^2 \approx \left|\frac{12 \cdot 10^{-6}}{(b-a)f''(c)}\right| = \frac{12 \cdot 10^{-6}}{|b-a| \cdot |f''(c)|}\,. \tag{1}$$

Dans cette expression seule $|f''(c)|$ n'est pas connue directement et un majorant doit être obtenu sur l'intervalle d'intégration considéré. Dans ce qui suit nous présentons une des manières possibles d'estimer un tel majorant. Notons que, dès que ce majorant est obtenu, on peut estimer la taille h des sous-intervalles via (1), les intégrales étant alors évaluées (par exemple) via

```
int = itg_trp(f,a,b,ceil((b-a)/h));
```

et indiqués dans la première colonne de la Table 1.

On notera néanmoins qu'il n'y a aucune garantie que la méthode de Simpson est plus précise que la méthode de trapéses.

Une des majorations de $|f''(c)|$ possibles :
a) pour $x \in [-2, 1]$ on a

$$|f''(x)| = e^x(x+2) \leq 3e\,,$$

et donc (1) devient $h \approx 7.0\,10^{-4}$.

b) on a

$$|f''(x)| = |\frac{2}{\sqrt{\pi}}2e^{-x^2}(2x^2-1)|\,,$$

dont le maximum sur $x \in [0, 2]$ est atteint en $x = 0$; dès lors

$$|f''(x)| \leq \frac{4}{\sqrt{\pi}} \approx 2.2568\,,$$

et donc (1) devient $h \approx 0.0016$.

c) on a

$$|f''(x)| = \left|\frac{(2-x^2)\sin(x) - 2x\cos(x)}{x^3}\right|\,,$$

dont le supremum sur $[0, \pi]$ correspond à la limite de f'' pour x tendant vers 0 ; dès lors

$$|f''(x)| \leq \frac{1}{3} \approx 0.3333\,,$$

et donc (1) devient $h \approx 0.0034$.

NOTE : dans le cas c) la fonction à intégrer présente une fausse singularité en 0 . Pour éviter la division zéro par zéro (qui génère un `NaN`), on peut remplacer cette fonction par

$$f(x) = \begin{cases} 1 & \text{si x} = 0 \\ \frac{\sin(x)}{x} & \text{si non} \end{cases}$$

C'est en particulier la solution utilisée pour génération de la première colonne de la Table 1 ; le code correspondant est :

```
clear; close
inta = itg_trp(@(x)(x*e^x), -2, 1, ceil(3/7.0e-04))
intb = itg_trp(@(x)((2/sqrt(pi))*e.^(-x.*x)), 0, 2, ...
                ceil(2/0.0016))
intc = itg_trp(@f1, 0, pi, ceil(pi/0.0034))

function r = f1(x)
    if(x==0)
        r = 1.0;
    else
        r = sin(x)/x;
    end
```

Une autre option est de légèrement modifier les bornes d'intégration en utilisant $[\delta, \pi]$ au lieu de $[0, \pi]$, avec un δ choisi de manière à contrôler l'erreur ainsi commise. Dans ce cas, l'erreur liée à la modification de borne est majorée par $\delta \cdot \sup_{x\in[0,\delta]} |f(x)|$ (valeur maximale de l'aire non prise en compte sur l'intervalle $[0, \delta]$) , et l'erreur globale commise satisfait

$$|E_{\text{glob}}(h)| \;\leq\; \left|\frac{1}{12}(b-a)h^2 f''(c)\right| + \delta \cdot \sup_{x\in[0,\delta]} |f(x)|$$

avec ici $a = \delta$, $b = \pi$ et $c \in]\delta, \pi[$. De plus, pour le cas considéré on peut montrer que $\sup_{x\in[0,\delta]} |f(x)| = 1$. Dès lors, en choisissant par exemple $\delta = 10^{-7}$, on a toujours $|E_{\text{glob}}(h)| \;\leq\; 10^{-6}$ si

$$\left|\frac{1}{12}(\pi - 10^{-7})h^2 f''(c)\right| \leq 10^{-6} - 10^{-7} = 9\,10^{-7} \,.$$

En utilisant l'inégalité $|f''(c)| \leq 1/3$ vue plus tôt, le pas d'intégration doit satisfaire

$$h^2 \;\leq\; \frac{12 \cdot 9\,10^{-7}}{(\pi - 10^{-7}) \cdot 1/3} \;<\; (0.0032)^2 \,.$$

3. Il suffit de reprendre l'exercice 2 avec la méthode de Simpson au lieu de celle des trapèzes :

```
int = itg_smp(f,a,b,ceil((b-a)/h));
```

Les résultats sont indiqués dans la deuxième colonne de la Table 1. La troisième colonne indique la différence entre les deux méthodes ; notez que toutes les différences sont en effet inférieures à 10^{-6}, ce qui confirme indirectement que la borne imposée sur l'erreur globale est satisfaite.

4. Notons pour commencer que l'équivalence peut être démontrée sur chaque sous-intervalle séparément. L'application de la méthode des trapèzes avec les sous-intervalles de taille h sur l'intervalle $[x_i, x_i + 2h]$ donne

$$I_1(h) \;=\; \frac{h}{2}(f_i + 2f_{i+1} + f_{i+2}) \,.$$

Par ailleurs, la méthodes des trapèzes avec les intervalles de taille $2h$ fournit

$$I_1(2h) \;=\; \frac{2h}{2}(f_i + f_{i+2})\,.$$

Le premier pas de la méthode de Romberg mène donc pour l'intervalle $[x_i, x_i + 2h]$ à

$$I_2(h) \;=\; \frac{4I_1(h) - I_1(2h)}{3} \;=\; \frac{2h}{6}(f_i + 4f_{i+1} + f_{i+2})\,,$$

ce qui n'est rien d'autre que la méthode de Simpson appliquée à l'intervalle $[x_i, x_i + 2h]$.

Annexe

Le code des fonctions `itg_trp` et `itg_smp` :

— méthode des trapèzes :

```
function int = itg_trp(f,a,b,ni)
% INT = itg_trp(F,A,B,NI)
%    retourne l'intégrale numérique INT sur l'intervalle
%    [A,B] de la fonction F d'une variable scalaire.
%    La fonction utilise la formule de trapèzes
%    avec NI sous-intervalles de même taille;
%
% arguments:
%    F - la fonction d'une variable à intégrer
%    A, B - les extrémités de l'intervalle d'intégration
%    NI - nombre de sous-intervalles; doit être > 0
%
% sortie:
%    INT - approximation de l'intégrale de F sur [A,B]
    if(ni <= 0)
        error('NI doit être > 0');
    else
        h = (b-a)/ni;
        int = 0;
        for i = 1:ni
            int = int + (h/2) * ( f(a+(i-1)*h) + f(a+i*h) );
        end
    end
```

— méthode de Simpson :

```
function int = itg_smp(f,a,b,ni)
% INT = itg_smp(F,A,B,NI)
%    retourne l'intégrale numérique INT sur l'intervalle
%    [A,B] de la fonction F d'une variable scalaire.
%    La fonction utilise la formule de Simpson
%    avec NI sous-intervalles de même taille;
%
% arguments:
%    F - la fonction d'une variable à intégrer
%    A, B - les extrémités de l'intervalle d'intégration
```

```
%    NI - nombre de sous-intervalles; doit être > 0
%
% sortie:
%    INT - approximation de l'intégrale de F sur [A,B]
    if(ni <= 0)
        error('NI doit être > 0');
    else
        h = (b-a)/ni;
        int = 0;
        for i = 1:ni
            int = int + (h/6)*( f(a+(i-1)*h) ...
                            + 4*f(a+(i-1/2)*h) + f(a+i*h) );
        end
    end
```