TP noté du 16 décembre 2020

- *Durée : 2 heures.*
- *La réponse aux questions théoriques doit être rédigée sur une feuille de copie.*
- *La réponse aux questions nécessitant une programmation python sera rédigée dans le notebook qui doit être téléchargé sur Moodle.*
- *A la fin des 2 heures, le notebook doit être déposé sur Moodle.*
- *Le barème est indicatif.*

Exercice 1 (9 points).

Soit $\alpha \in]0,1[$ et $\omega = (\omega_0, \omega_1, \omega_2) \in \mathbb{R}^3$. On considère la formule d'intégration numérique élémentaire sur $[-1,1]$:

$$J^{Q,\omega,\alpha}_{[-1,1]}(f) = \omega_0 f(-\alpha) + \omega_1 f(0) + \omega_2 f(\alpha)$$

pour approcher $I_{[-1,1]}(f) = \int_{-1}^{1} f(s)ds$.

Partie à rédiger sur feuille. deg ≤ p.

1. Déterminer $\omega(\alpha) = (\omega_0(\alpha), \omega_1(\alpha), \omega_2(\alpha))$ pour que la formule de quadrature $J^{Q,\omega(\alpha),\alpha}_{[-1,1]}$ soit d'ordre au moins 2.

2. Montrer que, quel que soit $\alpha \in]0,1[$, la formule de quadrature
$$J^{Q,\alpha}_{[-1,1]} = \frac{1}{3\alpha^2}\big(f(-\alpha) + f(\alpha)\big) + 2\left(1 - \frac{1}{3\alpha^2}\right) f(0)$$
est au moins d'ordre 3.

3. Montrer qu'il existe $\alpha^\star \in]0,1[$ pour lequel la méthode $J^{Q,\alpha^\star}_{[-1,1]}$ est d'ordre 5.

4. Écrire la formule de quadrature élémentaire associée à $J^{Q,\alpha}_{[-1,1]}(f)$ sur un intervalle quelconque $[c,d]$: $J^{Q,\alpha}_{[c,d]}(f)$.

5. En déduire la formule de quadrature composée qui approche $\int_a^b f(x)dx$ sur une subdivision régulière de pas h : $(x_i = a + ih)_{0 \le i \le N}$, avec $h = \dfrac{b-a}{N}$: $J^{Q,\alpha}_{[a,b],h}(f)$.

6. Pour $a = 0$, $b = \frac{\pi}{2}$ et $f(x) = \sin(x)$, donner la valeur de $\int_a^b f(x)dx$.

Partie programmation en python, dans le notebook.

1. Écrire une fonction `Jelementaire(f,c,d,alpha)` qui calcule $J^{Q,\alpha}_{[c,d]}(f)$.

2. En déduire une fonction `Jcomposee(f,a,b,alpha,N)` qui calcule $J^{Q,\alpha}_{[a,b],h}(f)$.

3. Tester votre fonction `Jcomposee` pour $a = 0$, $b = \pi/2$, $f(x) = \sin(x)$ et une ou deux valeurs de α de votre choix.

4. Écrire une fonction `calculerreur(f,a,b,Iex,alpha,k)` pour calculer les erreurs commises par la formule de quadrature pour des valeurs de $N = (5 \cdot 2^j)_{0 \leq j \leq k}$.

5. Mettre en évidence graphiquement l'ordre de convergence de la méthode $\alpha = 0.5$ et $\alpha = \sqrt{\frac{3}{5}}$.

Exercice 2 (6 points).

Soit $F : \mathbb{R} \to \mathbb{R}$ et $y_0 \in \mathbb{R}$. On s'intéresse à la résolution approchée du problème de Cauchy :

$$\begin{cases} y' = F(y), \\ y(0) = y_0. \end{cases}$$

Pour calculer une solution approchée sur un intervalle $[0, T]$, on se donne une subdivision :

$$t_n = n\Delta t \quad \forall 0 \leq n \leq N, \text{ avec } \Delta t = T/N.$$

On propose d'étudier la méthode numérique suivante, que l'on appellera "méthode d'Euler modifiée" :

$$\begin{cases} Y_{n+1} = Y_n + \Delta t\, F\left(Y_n + \dfrac{\Delta t}{2} F(Y_n)\right), \\ Y_0 = y_0. \end{cases}$$

Partie à rédiger sur feuille.

1. À quelle(s) méthode(s) d'intégration numérique est associée la méthode proposée ?

2. Dans le cas du modèle malthusien $F(x) = ax$, exprimer Y_{n+1} en fonction de Y_n. Dans ce cas particulier, avec quelle méthode vue en TP, la méthode d'Euler modifiée coïncide-t-elle ?

3. Quel est par conséquent l'ordre de convergence attendu ?

Partie programmation en python, dans le notebook.

1. Écrire une fonction `EulerMod` qui calcule la suite $(Y_n)_{0 \leq n \leq N}$ donnée par la méthode d'Euler modifiée. Les paramètres d'entrée sont la fonction F, la donné initiale y_0, l'instant final T et N.

$T = \cancel{56} \leftarrow 20.$

2. Tester la fonction `EulerMod` sur le modèle de croissance logistique vu en cours et en TP (on rappelle que $F(x) = x(a - bx)$). On tracera sur un même graphique la solution exacte sur l'intervalle $[0, 20]$ ainsi que la solution approchée calculée pour $N = 5$ et $N = 10$ pour $a = 0.2$, $b = 0.001$ et $y_0 = 0.1 \times a/b$.

3. À l'aide d'une fonction `calculerreurs`, illustrer graphiquement l'ordre de convergence de la méthode d'Euler modifiée.

Exercice 3 (5 points).
Soient $f \in \mathcal{C}^2(\mathbb{R})$ et $\bar{x} \in \mathbb{R}$ tels que $f(\bar{x}) = 0$ et $f'(\bar{x}) \neq 0$.

Partie à rédiger sur feuille

1. Rappeler la méthode de Newton dédiée au calcul de la valeur approchée de $\bar{x}$.

2. On introduit une variante de la méthode de Newton :

$$\begin{cases} x_0 \text{ donné}, \\ y_n = x_n - \dfrac{f(x_n)}{f'(x_n)}, \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)}, \quad n \geq 0. \end{cases}$$

Définir une fonction g telle que cette méthode itérative se réécrive:

$$\begin{cases} x_0 \text{ donné}, \\ x_{n+1} = g(x_n). \end{cases}$$

On pourra introduire h telle que $h(x) = x - \dfrac{f(x)}{f'(x)}$

3. Calculer $g(\bar{x})$ et $g'(\bar{x})$. Que peut-on en déduire sur la convergence et l'ordre de la méthode proposée ?

Partie programmation en python, dans le notebook.

1. Écrire une fonction `NewtonMod` qui a pour paramètres d'entrée f, f', x_0, une tolérance τ et un nombre d'itérations $nmax$ (il faudra choisir un critère d'arrêt). La fonction renvoie l'approximation calculée ainsi que le nombre d'itérations N.

2. Soit $f(x) = \sin(x)$. Utiliser la fonction `NewtonMod` avec $x_0 = 4$ pour déterminer une valeur approchée du zéro π. Faire varier la tolérance ($\tau = 10^{-k}$ pour $1 \leq k \leq 14$). Représenter sur un premier graphique le nombre d'itérations effectuées en fonction de τ et sur un second graphique l'erreur $|x_N - \pi|$ en fonction de τ.

y en f x