Matmeca 1ère année - TP de Calcul Scientifique en Fortran
année 2018-2019

Approximation par différences finies de l'équation d'advection

Le but de ce TP est de programmer quelques schémas numériques vus en cours pour résoudre l'équation d'advection.

On rappelle qu'il est *indispensable* d'utiliser des options de debuggage pendant la phase de conception de vos programmes (au moins `-fcheck=all`).

Votre rapport devra contenir les réponses aux questions posées, des courbes significatives, et des commentaires pertinents.

1 Schéma décentré

Dans cette partie, tout devra être programmé dans le programme `advection`, en double précision.

On considère l'équation d'advection posée dans $[0,1]$, avec condition aux limites d'entrée :

$$\begin{aligned}
&\partial_t \rho + a\partial_x \rho = 0, \qquad x \in [0,1],\\
&\rho(0,x) = \rho_{init}(x) = \cos 2\pi x\\
&\rho(t,0) = \rho_G = 1,
\end{aligned}$$

où la vitesse d'advection a est supposée positive.

Question 1. Solution exacte. Dans cette question, le but est de calculer et stocker dans un fichier la solution exacte

$$\rho(t,x) = \begin{cases} \rho_{init}(x-at) & \text{si } x-at>0,\\ 1 \quad \text{sinon.} \end{cases}$$

Le programme doit calculer et tracer la solution en un certain nombre de points et à un temps donné. Pour cela, créez le fichier `donnees.dat` contenant la valeur de la vitesse `a` et le temps de calcul de la solution `tmax`.

Le programme commence par lire ces données, puis calcule la solution exacte au temps `tmax` en 200 points uniformément répartis dans $[0,1]$, et stocke le résultat dans le fichier `sol_exacte.dat`, afin de pouvoir afficher la solution avec gnuplot.

Dans le module `mod_solution`, programmez les fonctions `rho_init(x)` qui calcule la donnée initiale en un réel x donné, et `rho_ex(t,x)` qui calcule la solution exacte pour un temps t et une position x donnés.

La variable `a` sera déclarée dans le module `mod_param`, ainsi que toutes les constantes utiles au programme.

Calculez et tracez alors la solution exacte aux temps `tmax`= 0.1, 0.2, 04, 0.8. Vérifiez que les résultats correspondent à ce que vous attendez.

Question 2. **Schéma décentré**. Programmez ensuite le schéma décentré

$$\begin{aligned}
&\frac{\rho_i^{n+1}-\rho_i^n}{\Delta t}+a\frac{\rho_i^n-\rho_{i-1}^n}{\Delta x}=0, \qquad i=1:i_{max}+1,\\
&\rho_i^0=\rho_{init}(x_i),\\
&\rho_0^n=\rho_G.
\end{aligned}$$

Pour cela, créez le maillage à $i_{max}+2$ points $x_i=i\Delta x$, où $i=0$ à $i_{max}+1$, avec $\Delta x=1/(i_{max}+1)$. La valeur de i_{max} sera lue dans le fichier `donnees.dat`, à la suite des autres valeurs déjà stockées dans le fichier, de même que le nombre `cfl` qui permet de définir le pas de temps $\Delta t=cfl\Delta x/a$.

La solution numérique sera tracée au temps t_{max} en tous les points du maillage dans le fichier `decentre.dat`.

Calculez cette solution numérique aux mêmes temps que précédemment, comparez à la solution exacte, et commentez.

Question 3. **Instabilités**. Augmentez maintenant la CFL en prenant `cfl=1.5`, avec $t_{max}=0.1$, 0.2, 0.3, et 0.4. Qu'observez-vous ?

2 Autres schémas

Question 4. **Schéma de Lax-Wendroff.** Le schéma de Lax-Wendroff est

$$\begin{aligned}
&\rho_i^{n+1}=\rho_i^n-\frac{a\Delta t}{2\Delta x}(\rho_{i+1}^n-\rho_{i-1}^n)+\frac{1}{2}\left(\frac{a\Delta t}{\Delta x}\right)^2(\rho_{i+1}^n-2\rho_i^n+\rho_{i-1}^n), \qquad i=1:i_{max}+1,\\
&\rho_i^0=\rho_{init}(x_i),\\
&\rho_0^n=\rho_G.
\end{aligned}$$

Cependant, la structure centrée de ce schéma pose problème : au dernier point du maillage, le terme ρ_{i+1}^n nécessite la donnée de $\rho_{i_{max}+2}^n$, a priori inconnue. On doit donc définir une *condition aux limites artificielle* au bord droit. Une solution très simple est obtenue par extrapolation à l'ordre 0, qui donne

$$\rho_{i_{max}+2}^n=\rho_{i_{max}+1}^n,$$

qu'on appelle aussi *condition de sortie*.

Programmez alors ce schéma dans votre programme, à la suite du schéma précédent, et comparez vos résultats à la solution exacte et au schéma décentré pour `imax`=100, `cfl`=0.9, et `tmax`=0.1, 0.2, 0.4, 0.8.

Question 5. Schéma centré. Rajoutez à présent à votre programme le schéma centré

$$\rho_i^{n+1} = \rho_i^n - \frac{a\Delta t}{2\Delta x}(\rho_{i+1}^n - \rho_{i-1}^n), \qquad i = 1 : i_{max} + 1,$$
$$\rho_i^0 = \rho_{init}(x_i),$$
$$\rho_0^n = \rho_G,$$

en utilisant la même condition aux limites de sortie $\rho_{i_{max}+2}^n = \rho_{i_{max}+1}^n$ que pour le schéma de Lax-Wendroff.

Testez ce schéma avec `tmax`=0.1, 0.2, 0.4, et commentez ce que vous observez.

3 Artefacts numériques

Le but de cette partie [1] est d'observer les phénomènes de diffusion et dispersion numérique. Pour cela, nous allons étudier l'équation d'advection sur un temps plus long, dans un domaine plus étendu, avec des conditions aux limites et initiale différentes.

Faites une copie de votre programme dans le fichier `artefacts.f90`, et programmez tout ce qui suit dans ce nouveau programme.

Question 6. Comparaison sur une gaussienne. Adaptez le programme pour qu'il résolve maintenant l'équation d'advection sur le domaine en espace $[0, 25]$ avec $t_{max} = 17$, $a = 1$, $i_{max} = 500$, $cfl = 0.8$, et les données

$$\rho_{init}(x) = \exp(-(x-5)^2), \qquad \rho_G = 0.$$

Vous pouvez enlever le schéma centré de votre programme, il ne sera plus utilisé.

Observez alors les différences entre la solution exacte, le schéma décentré, et le schéma de Lax-Wendroff. Qu'observez-vous ? Décrivez les résultats de chaque schéma et nommez les résultats que vous observez.

Pour mieux voir ce qui se passe, vous pouvez zoomer sur la partie intéressante de la solution avec la commande gnuplot `set xrange[10:25]` suivie de `replot`. Pour annuler ce zoom, taper `set xrange[*:*]` suivie de `replot`.

Question 7. CI à plus fort gradient. Mêmes questions avec une CI à laquelle on rajoute une gaussienne plus étroite :

$$\rho_{init}(x) = \exp(-(x-5)^2) + \exp(-20(x-2)^2).$$

Question 8. Diminuez le pas de temps avec `cfl=0.4` puis `cfl=0.1`. Quelle conséquence cela a-t-il sur les résultats ?

Question 9. Même question en raffinant le maillage en espace : testez `imax`=1000, puis 2000, avec `cfl=0.9`.

1. issue du livre de R. J. Leveque "Finite Difference Methods for Ordinary and Partial Differential Equations"