Université de Lille 1
Master 1 de Mathématiques 2004 / 2005
Mathématiques Assistées par Ordinateur (MAO)

Projet sur l'équation de Schrödinger cubique

Systèmes différentiels linéaires

On considère le système différentiel

$$\frac{du(t)}{dt} + Au(t) = f(t), \tag{1}$$

$$u(0) = u_0 \in \mathbb{R}^N, \tag{2}$$

où A est une matrice $N \times N$ donnée et $f : t \mapsto f(t) \in \mathbb{R}^N$.

1. Montrer que

$$u(t) = e^{-tA}u_0 + \int_0^t e^{-(t-s)A} f(s)ds$$

ce qui revient à

$$u(t + \Delta t) = e^{-\Delta tA}u(t) + \int_0^{\Delta t} e^{-(\Delta t-s)A} f(t + s)ds \tag{3}$$

2. On s'intéresse à l'approximation numérique des solutions du système. On considère 2 niveaux d'approximation

i. Approximation de l'intégrale

$$\begin{aligned}\int_0^{\Delta t} e^{-(\Delta t-s)A} f(t+s)ds \ &\approx \int_0^{\Delta t} e^{-(\Delta t-s)A} f(t)ds = A^{-1}(Id - e^{-\Delta tA})f(t) \text{ (rectangles à gauche)} \\ &\approx A^{-1}(Id - e^{-\Delta tA})f(t + \Delta t) \text{ (rectangles à droite)}\end{aligned}$$

ii. Approximation de l'exponentielle

- Schéma exact : $e^{-\Delta tA}$
- Approximation explicite

$$e^{-\Delta tA} \approx Id - \Delta tA$$

- Approximation implicite

$$e^{-\Delta tA} \approx (Id + \Delta tA)^{-1}$$

On décompose l'intervalle [0,T] en sous-intervalles de longueur Δt et on pose $t_k = k\Delta t$. On pose $f_k = f(t_k)$ et on note u^k l'approximation de $u(t_k)$ obtenue par un des trois schémas suivants :

- Schéma exact pour les termes linéaires

$$\text{pour } k = 0, \cdots \ u^{k+1} = e^{-dtA}u^k + A^{-1}(Id - e^{-\Delta tA})f_k$$

- Schéma d'Euler explicite

$$\text{pour } k = 0, \cdots \quad \text{on définit } (u^k)_{k\geq 0} \text{ par :} u^{k+1} = (Id - \Delta t A)u^k + \Delta t f_k$$

- Schéma d'Euler implicite

$$\text{pour } k = 0, \cdots \quad \text{on définit } (u^k)_{k\geq 0} \text{ par :} u^{k+1} = (Id + \Delta t A)^{-1}(u^k + \Delta t f_k)$$

(a) Explorer la commande `expm` de Matlab.

(b) Ecrire ces schémas lorsque la matrice est diagonale. En déduire un algorithme de résolution lorsque la matrice est diagonalisable.

(c) Application. Soit A la matrice $N \times N$ suivante :

$$A = \frac{1}{h^2} \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & & 0 \\ 0 & \ddots & \ddots & \ddots & \\ \vdots & & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{pmatrix}$$

avec $h = \dfrac{1}{N+1}$ et $f(t) = \cos(t) * exp(\sin(t))S + \pi^2 exp(\sin(t))C$ où

$$S = (\sin(\pi h), \sin(\pi 2h), \cdots, \sin(\pi N h))^T \text{ et } C = (\cos(\pi h), \cos(\pi 2h), \cdots, \cos(\pi N h))^T$$

de sorte que $u(t) = Sexp(\sin(t))$.
Programmer les 3 schémas et effectuer des simulations pour différentes valeurs de N et de Δt, on calculera l'évolution de l'erreur en fonction du temps. Conclusion ?
Indication : on construira la matrice A en Matlab à l'aide de la commande `A=gallery('poisson',N)`

(d) Proposer une méthode de calcul utilisant des transformées de Fourier discrètes.

Schrödinger cubique

On considère l'équation de Schrödinger cubique

$$i\frac{du}{dt} + u_{xx} + |u|^2 u = 0, \; x \in [0, L], \; t > 0 \tag{4}$$

$$u(x, 0) = u_0(x) \tag{5}$$

où $u : t \mapsto \mathbb{C}$ est une fonction L-périodique, $i = \sqrt{-1}$ et $|.|$ désigne le module d'un nombre complexe. On admettra que cette équation possède une unique solution u vérifiant

$$\int_0^L u^2(x,t)dx = \int_0^L u_0(x)^2 dx \text{ et } \frac{1}{2}\int_0^L (u_x(x,t))^2 dx - \frac{1}{4}\int_0^L u(x,t)^4 dx = cste.$$

On admettra que l'on peut décomposer u en série de Fourier

$$u(x,t) = \sum_{k=-\infty}^{+\infty} c_k(t) exp(-i\frac{2\pi k x}{T}).$$

En injectant cette expression dans l'équation, il vient

$$i\frac{dc_k}{dt} - (\frac{2\pi k}{L})^2 c_k + v_k = 0$$

où v_k est le $k$ième coefficient de Fourier de $|u|^2 u$. Le but de ce problème est de mettre en œuvre et de comparer différents schémas de discrétisation en temps lorsque la fonction u est approchée en espace par une série de Fourier tronquée.

On décide d'approcher $|u(t_{n+1})|^2 u(t_{n+1})$ par $|u(t_n)|^2 u(t_{n+1})$ sur l'intervalle $[t_n, t_{n+1}]$. Ainsi

$$i\frac{du}{dt} + u_{xx} + |u^k|^2 u = 0,\ x \in [0, L], t \in [t_n, t_{n+1}],$$

Cette équation est linéaire. En considérant la série de Fourier de chaque terme, on obtient l'équation différentielle ordinaire

$$i\frac{dc_k}{dt} - (\frac{2\pi k}{L})^2 c_k + |u^k|^2 c_k = 0,\ x \in [0, L], t \in [t_n, t_{n+1}], \tag{6}$$

$$c_k(t_n) \text{ donné .} \tag{7}$$

1. Résoudre cette EDO et en déduire une schéma d'approximation des solutions lorsqu'on remplace les coefficients de Fourier par ceux fournis par la FFT.

2. Programmer ce schéma en Matlab et le faire tourner pour les données suivantes :

```
N=32,(ou 64)
L=20; dx=L/N; for j=1:N,
x(j)=(j-1)*dx;
end
U=2.*exp(i*(xi-L/2)./2).*sech(sqrt(2).*(x-L/2));
```