Projet TP LM336

1 Particule dans un champ électromagnétique

Le but de cette section est de simuler numériquement un modèle d'évolution d'une particule chargée dans un champ électromagnétique.

1.1 Modèle

On considère une particule p que l'on repère par sa position $X = (x, y, z)$ dans l'espace euclidien R^3, de charge q_p et de masse m_p. En appliquant le principe fondamental de la dynamique, combiné à la force de Lorentz, et en ajoutant des conditions initiales, on arrive au problème de Cauchy suivant

$$\begin{cases} m_p\ddot{X}(t) = q_p(E(X(t), t) + \dot{X}(t) \wedge B(X(t), t)) \\ X(t_0) = X_0 \\ \dot{X}(t_0) = V_0 \end{cases} \tag{1}$$

avec E le champ électrique, B le champ magnétique. On rappelle que si $u = (u_1, u_2, u_3)$ et $v = (v_1, v_2, v_3)$, le produit vectoriel $u \wedge v$ est défini par

$$u \wedge v = \begin{pmatrix} u_2v_3 - u_3v_2 \\ -u_1v_3 + u_3v_1 \\ u_1v_2 - u_2v_1 \end{pmatrix}$$

On notera (e_1, e_2, e_3) la base canonique de l'espace R^3.

1.2 Analyse

Dans cette partie, nous supposons que E et B sont des champs de vecteur donnés de classe C^∞.

a) Écrire le système 1 sous la forme d'un système d'ordre 1.

b) Montrer l'existence et l'unicité de la solution du problème de Cauchy 1.

c) En supposant que $B = 0$ et que E dérive d'un potentiel V (i.e $E = -\nabla V$), montrer que le système admet pour hamiltonien

$$H(Q, P) = \frac{P^2}{2m} + \frac{q_p}{m_p}V(Q) \qquad \text{avec} \qquad P = X, Q = \dot{X}$$

1.3 Simulation numérique

a) Écrire une fonction Euler qui prend en paramètres un vecteur de temps t, une position initiale X_0, une vitesse initiale V_0, et qui renvoie une matrice composée de trois colonnes, chacune étant une coordonnée de la solution approchée avec la méthode d'Euler (par exemple, $M = [x, y, z]$, avec $x = (x_0, ..., x_n)$, $y = (y_0, ..., y_n)$, $z = (z_0, ..., z_n)$. Remarque: écrire des fonctions valables pour le cas général où B et E dépendent de l'espace et du temps.

b) Écrire une fonction AfficherSolution qui prend en paramètres le pas de temps Δt, le temps final T, et qui affiche la trajectoire approchée de la solution avec la méthode d'Euler.

c) Bonus: Écrire une fonction AnimationSolution qui affiche une animation, un film montrant le déplacement de la particule au cours du temps.

d) Écrire une fonction EulerRichardson, similaire à la fonction Euler, mais basée sur la méthode d'Euler-Richardson à pas adaptatif 1.

e) Bonus: En s'inspirant de l'algorithme d'Euler-Richardson, proposez une méthode d'ordre 4 à pas adaptatif.

Data: Donnée initiale $x(t_0)$, temps final T, pas de temps Δt, tolérance d'erreur tol
Result: Approximation de la fonction x solution du problème de Cauchy
$\dot{x}(t) = f(t, x(t)), x(t_0) = x_0$
$x_0 = x(t_0)$;
while $t < T$ **do**
  $k_1 = f(t_n, x_n)$;
  $k_2 = f(t_n + \frac{\Delta t_n}{2}, x_n + \frac{\Delta t_n}{2} k_1)$;
  $\epsilon = |k_2 - k_1|$;
  if $\epsilon > tol$ **then**
    $\Delta t = 0.9\sqrt{\frac{tol}{\epsilon}}\Delta t$
  else
    $x_{n+1} = x_n + k_2 \Delta t_n$;
    $\Delta t = 0.9\sqrt{\frac{tol}{\epsilon}}\Delta t$;
    $t_{n+1} = t_n + \Delta t$;
  end
end

Algorithm 1: Méthode d'Euler-Richardson

1.4 Application: accélérateur de particules

Dans ce cas, on supposera $E = E_0 e_3$ et $B = B_0 e_3$. Pour le problème de Cauchy, on posera $X(0) = (0, 0, 0)$ et $\dot{X}(0) = (1, 0, 1)$.

a) Solution exacte. Calculer la solution exacte du problème de Cauchy dans ce cas. Pour cela, on commencera par découpler le système (une EDO sur z, un système d'EDO couplées sur x et y), puis on introduira la notation $u = x + iy$, on résoudra une EDO sur u, puis on identifiera la partie réelle et la partie imaginaire de u pour trouver la solution.

b) Estimation d'énergie. On définit la puissance comme $P = f.v$. En appliquant le théorème de l'énergie cinétique $\frac{\partial E_c}{\partial t} = P$, montrer que le système est conservatif dans ce cas.

c) Graphe. Écrire une fonction TracerGraphe qui trace sur un graphique 3D la solution exacte et la solution approchée.

d) Courbe de convergence. Tracer sur un graphique la courbe $(\Delta t, Err_T \Delta t)$ pour $\Delta t = 10^{-1}, ..., 10^{-8}$ avec

$$err_T(\Delta t) = ||X(T) - X_{\Delta t}(T)||_2$$

où $X_{\Delta t}(T)$ désigne la solution approchée avec un pas de temps Δt au temps T.

2 Population structurée et principe de sélection

Le but est de modéliser une population structurée par un trait phénotypique par exemple une population de cellules tumorales structurée par leur niveau de résistance à une chimiothérapie. On s'intéresse à leur dynamique temporelle. La fonction $n(t,x)$ décrit la densité des individus avec le trait x au temps t. Comme modèle on propose une équation non-locale

$$\begin{cases} \frac{\partial}{\partial t} n(t,x) = n(t,x) \underbrace{[\overbrace{p(x)}^{\text{prolifération}} - \overbrace{d(x)\rho(t)}^{\text{competition}}]}_{R\left(x,\rho(t)\right)} & t \in [0,\infty), \quad x \in [0,1), \\ \rho(t) = \int n(t,x)\,dx, \\ n(t=0,x) = n_0(x) \geq 0. \end{cases} \tag{2}$$

Bien que le modèle fait intervenir des dérivées partielles, on va approcher l'équation (2) par un système d'EDO afin de développer et implémenter une approximation numérique. On commence avec R et n_0 constantes.

a) Pour $R \geq 0$, démontrer que Euler explicite préserve la positivité de la donnée initiale.

b) Pour $R \leq 0$, démontrer que Euler implicite préserve la positivité de la donnée initiale.

c) En déduire une méthode qui préserve la positivité de la donnée initiale indépendamment du signe de R.

d) Écrire une fonction AfficherSolution qui affiche la solution pour $R = -2, -1, 0, 1, 2$.

e) Étendre la méthode pour traiter le système

$$\frac{d}{dt} \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} = \begin{pmatrix} R_1 n_1 \\ R_2 n_2 \end{pmatrix}$$

et la coder pour $R_1 = -1, R_2 = 1$ dans une fonction AfficherSolution2 qui affiche la solution et dans une deuxième fenêtre affiche le logarithme de la solution.

f) À partir de maintenant on prend R et n comme des fonctions de x

$$\partial_t n(t,x) = R(x) n(t,x). \tag{3}$$

Pour $x \in [0,1)$, en utilisant l'approximation $n(t,x) \approx n_i(t)$ pour $x \in [i \cdot \Delta x, (i+1) \cdot \Delta x)$ ($\Delta x = 1/m, i \in \{0, 1, \ldots, m-1\}$), écrire un système des EDO comme l'approximation de (3).

g) Écrire une fonction AfficherSolutionSystem qui affiche la solution pour $R(x) = -4(x-0.3)^2 + 1/2$ avec $\Delta x = 1/200$; $1/1000$ et $T = 1$; 100 comme une surface en R^3 par exemple en utilisant plot3d. Comme donnée initiale, on pourrait prendre $n_0(x) = 2.5 \exp(-(x-0.7)^2/0.05)$.

h) À partir de maintenant on prend R comme une fonction de x et ρ. Utiliser l'approximation de $\rho(t) = \int n(t,x)\,dx$ par une somme de Riemann, pour écrire une fonction AfficherSolutionSystem-Rho qui affiche la solution de (2) avec $R(x, \rho(t)) = -4(x-0.3)^2 + 1/2 - \rho(t)$, $\Delta x = 1/1000$ et $T = 1$; 100.

i) Qu'est-ce qu'on observe numériquement pour $t \to \infty$ (par exemple $t = 100$)?

j) Bonus : Démontrer que ρ reste borné.

k) Bonus : Déterminer les états stationnaires.