Projet n4

Systèmes non linéaires d'équations - Méthode Newton-Raphson

Groupe n1 - Equipe n2

Responsable : cel-ha910e
Secrétaire : cplages
Codeurs : aguichard002, jcalandra001

Résumé : Le but de ce projet est de programmer des algorithmes dédiés à la recherche des racines de systèmes d'équations non linéaires. La méthode utilisée ici est l'algorithme de Newton-Raphson, dont nous en évaluerons les avantages et inconvénients. Pour cela, nous testerons cette méthode dans différents cas : la détermination des points de Lagrange et la recherche de points d'équilibre électrostatique.

Méthode de Newton-Raphson

La méthode de Newton-Raphson représente la première partie de notre travail, permettant d'approximer les racines d'un système d'équation non linéaire, elle est requise dans les deux applications que nous étudierons par la suite.

Nous avons programmé la méthode de Newton-Raphson dans n'importe quelle dimension. L'idée de l'algorithme est de considérer qu'en une position U donnée, la meilleure direction pour s'approcher de la racine de la fonction f est donnée par la tangente de f en U.
Considérant U comme le vecteur position, nous obtenons le vecteur V suivant correspondant à l'équation $f(U+V) = 0$ où $f(U+V)$ est approximé par $f(U) + H(U) * V$ avec f la fonction précédente et H sa jacobienne. Notre but est donc de résoudre l'équation suivante : $H(U) * V = -F(U)$

Par la suite, nous remplaçons U par U + V jusqu'à ce que U converge ou que le critère d'arrêt (nombre d'itérations) soit atteint. C'est à dire, pour tester la convergence de U, nous prenons une fonction f et sa jacobienne J, récupérons le vecteur U de la fonction Newton-Raphson et vérifions que f(U) est égal à 0 à une certaine précision près.

```
fonction Newton_Raphson(f, J, U0, N, epsilon):
    U <- copie(U0)
    Pour tout i dans N faire
        fu <- f(U)
        na <- norme(fu)
        Si na < epsilon alors
            retourner U
        ju <- J(U)
        V <- solution_moindre_carre(ju, -fu)[0]
        U <- U + V
    Afficher "ERREUR : precision non atteinte"
retourner U
```

Listing 1 – Implémentation de la méthode `Newton_Raphson`

L'algorithme mis en place pour cette méthode est disponible dans le listing ci-dessus. Il prend en argument la fonction f et son Jacobien J ainsi que le vecteur position initial U0, le nombre maximal d'itérations N et la précision epsilon.

Enfin, nous avons amélioré notre programme en rajoutant un *backtracking* dans notre code, méthode permettant de revenir légèrement en arrière sur des décisions prises afin de sortir d'un blocage. Cette méthode nous est très utile dans le cas où notre vecteur position se met à diverger, cas tout à fait plausible étant donné que notre algorithme se base sur une position de départ donnée par l'utilisateur.

Détermination des points de Lagrange

Dans cette partie nous nous intéressons aux objets mobiles dans un plan et susceptibles d'être sujet aux forces suivantes :

La force élastique : $f_e : \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -kx \\ y \end{bmatrix}$

La force centrifuge : $f_c : \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} k(x - x_0) \\ k(y - y_0) \end{bmatrix}$

La force gravitationelle : $f_g : \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -k \times \frac{(x-x_0)}{((x-x_0)^2+(y-y_0)^2)^{3/2}} \\ -k \times \frac{(y-y_0)}{((x-x_0)^2+(y-y_0)^2)^{3/2}} \end{bmatrix}$

Nous avons utilisé la programmation fonctionnelle (fonction lambda dans python) pour représenter nos forces et les jacobiennes relative à cette somme de forces. Dans un premier temps, notre but consiste à déterminer les positions d'équilibre de cet objet. Pour cela, nous utilisons la méthode de Newton-Raphson dans les conditions suivantes : deux points A et B, de coordonnées $[0, 0]$ et $[1, 0]$, sont soumis à une force gravitationnelle de coefficient 1 et 0.01. De plus une force centrifuge, de coefficient 1, est appliquée au barycentre des deux masses (cf. Figure 1).

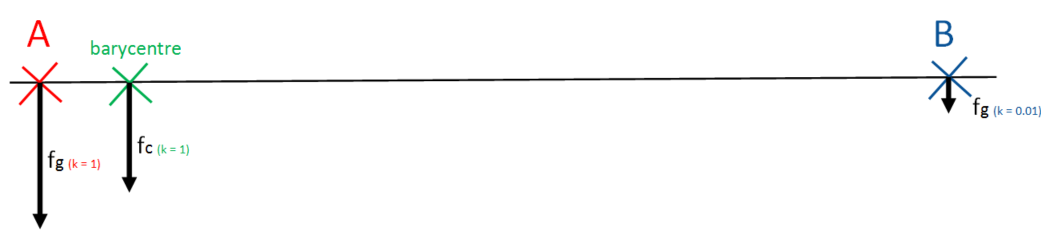

Figure 1 – Schéma simplifié des trois forces en jeu

En prenant le vecteur position initial $U = [1, 5, 0]$ nous obtenons bien les résultats attendus pour la force totale et sa jacobienne (l'égalité matricielle est conservée jusqu'à 8 décimales, au delà, la précision machine ne permet pas de garantir l'égalité).

Pour finir, nous nous intéressons au problème des points de Lagrange : « Un point de Lagrange est une position de l'espace dans un système à deux corps, où leurs champs de gravité se combinent de manière à fournir un point d'équilibre à un troisième corps de masse négligeable, tel que les positions relatives des trois corps soient fixes. Les points de Lagrange sont un cas particulier du problème à 3 corps, où l'un des 3 corps est de masse négligeable devant les 2 autres. ». *Explications tirées de Futura Sciences*

Dans l'application du système Terre-Soleil, les points de Lagrange sont ceux où un corps suit une orbite autour du Soleil (de coordonnée [0,0]) à la même vitesse angulaire que la Terre (de coordonnée [1,0]), sous l'action conjointe des attractions du Soleil et de la Terre. Les points L1 et L2 sont situés à proximité de la Terre, le point L3 est le symétrique de la Terre par rapport au Soleil, et les points L4 et L5 sont aux sommets de triangles équilatéraux ayant le segment Soleil-Terre pour base. Cette configuration est illustrée en Figure 2.

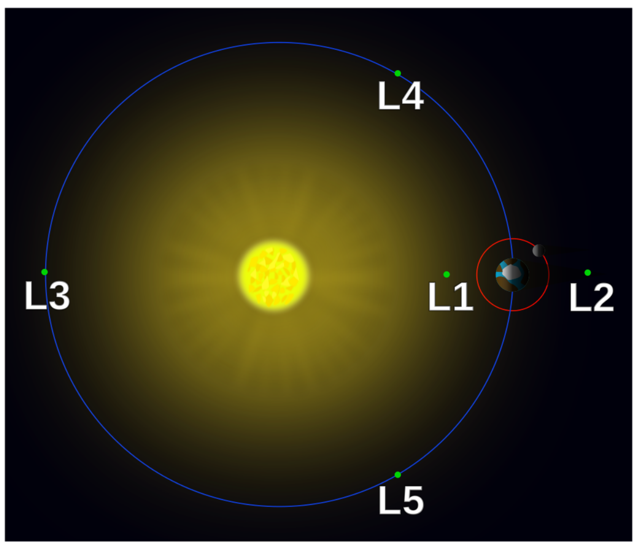

FIGURE 2 – Application théorique au système Terre-Lune des points de Lagrange

Nous avons appliqué la méthode de Newton-Raphson avec comme paramètres la somme des forces présentées précédemment, ainsi que la jacobienne relative à ces forces. Par soucis de précision, nous avons opté pour l'utilisation de la méthode de Newton avec Backtracking.

Afin de trouver les points de Lagrange, nous avons pris 5 points du plan de façon semi-aléatoire, c'est à dire, assez espacés les uns des autres sur le plan et s'approchant de la solution désirée (cf. Figure 2). La méthode de Newton nous donne un vecteur racine de la fonction, nous permettant de trouver le point pour lequel la somme des forces s'annule. Ici, chacun des points L_i choisis convergent bien vers l'une des racines, qui est également la coordonnée d'un point de Lagrange (cf. Figure 3).

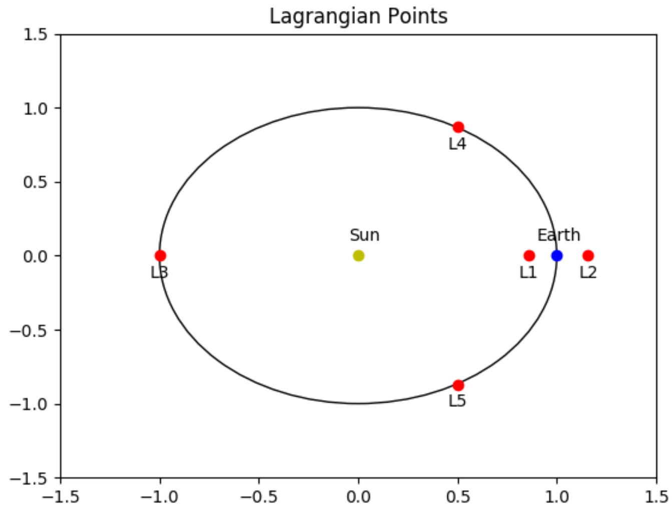

FIGURE 3 – Détermination des points de Lagrange

Recherche de points d'équilibre électrostatique

Le problème posé ici est celui de deux charges électrostatiques positionnées aux extrémités de l'axe $[-1, 1]$. Nous supposons qu'il existe N charges $x_1, x_2, ..., x_N$ en mouvement sur cet intervalle. L'énergie totale du système vaut :

$$E(x_1, x_2, ..., x_N) = \sum_{i=1}^{N} \log|x_i + 1| + \log|x_i - 1| + \frac{1}{2}\sum_{j=1, j\neq i}^{N} \log|x_i - x_j|$$

Afin de trouver les positions d'équilibre, nous devons résoudre le système d'équations non linéaire suivant :

$$\nabla E(x_1, x_2, ..., x_N) = [\frac{\partial E(x_1, ..., x_N)}{\partial x_i}] = 0$$

Nous avons utilisé la méthode de Newton-Raphson implémentée précédemment. Il nous a donc fallu dans un premier temps calculer la matrice jacobienne H de $\nabla E(x_1, x_2, ..., x_N)$ telle que :

$$H = [\frac{\partial^2 E(x_1, ..., x_N)}{\partial x_i \partial x_j}]_{1\leq i,j\leq N}$$

Après implémentation de la matrice jacobienne, nous avons utilisé notre fonction Newton-Raphson afin de résoudre le système, puis placer les points correspondants sur l'axe $[-1, 1]$. Nous obtenons ainsi un graphique montrant les points d'équilibre pour 4 charges électrostatiques (cf. Figure 4).

Une autre interrogation à présent est de comparer nos points aux racines des dérivées des polynômes de Legendre. Nous avons donc superposé celles-ci au graphique existant. Nous pouvons observer que les polynômes de Legendre s'annulent aux mêmes positions que les points d'équilibre.

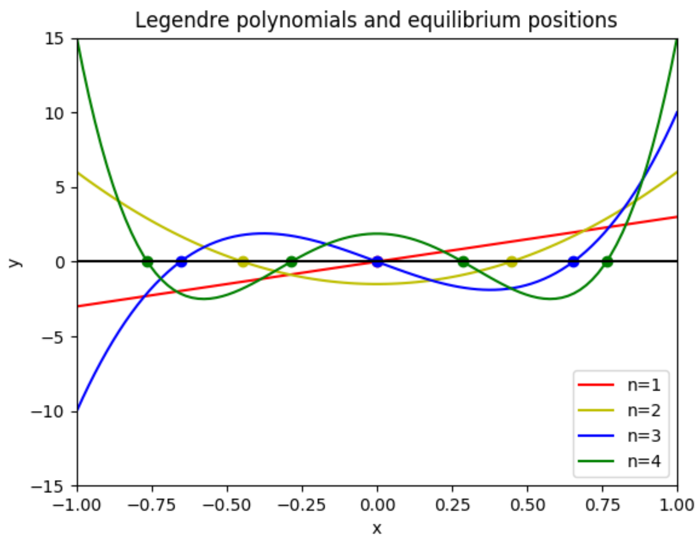

FIGURE 4 – Polynômes de Legendre et positions d'équilibre

Enfin, une dernière interrogation était de savoir si ces points correspondaient à des maximums ou des minimums d'énergie. Quand la dérivée de l'énergie est nulle (c'est à dire à une position d'équilibre), on a un équilibre soit instable ($\partial^2 E < 0$), soit stable ($\partial^2 E > 0$). Afin de déterminer si un point est un minimum ou un maximum d'énergie, il faut donc s'intéresser au signe de la dérivée seconde de l'énergie. Si c'est un équilibre stable, nous avons affaire à un minimum d'énergie, en revanche s'il s'agit d'un équilibre instable, nous avons affaire à un maximum d'énergie.

Conclusion

L'objectif de ce projet était d'implémenter une méthode résolvant un système d'équations non-linéaires : Newton-Raphson. Nous avons testé cette méthode sur deux applications et pouvons maintenant en tirer les avantages et inconvénients. Tout d'abord cette méthode possède une vitesse de convergence bien plus intéressante que la plupart des méthodes de recherche de racines de tels systèmes. En revanche, l'incertitude quant à la convergence vers la solution reste un point noir. Cependant le *backtracking* vient palier à ce problème en permettant de revenir en arrière en cas de mauvais vecteur initial.