Méthode d'Euler dans le plan.

Remarque importante : ce devoir maison fera l'objet d'une évaluation individuelle devant écran lors de vos TD la semaine du lundi 15 décembre. Vous disposerez de votre machine et de vos codes Python que vous me présenterez et que nous exécuterons sur place.

Introduction

Dans le devoir maison de M. BECK, vous vous êtes intéressés aux systèmes différentiels de la forme :

$$\begin{cases} x' = \dfrac{dx}{dt} = f(x, y) \\ y' = \dfrac{dy}{dt} = g(x, y) \end{cases}$$

Pour résoudre numériquement un tel système autour d'un point singulier $S(x_S, y_S)$, on peut utiliser le schéma d'Euler (à pas constant *h*) suivant :

$$\begin{cases} (x_0, y_0) \neq (x_S, y_S) \\ x_{i+1} = x_i + f(x_i, y_i) \times h \\ y_{i+1} = y_i + g(x_i, y_i) \times h \end{cases}$$

Le couple $(x_0, y_0) = (x(t_0), y(t_0))$ correspond aux conditions initiales.

Résolution d'un système linéaire

Préliminaires

Dans le devoir maison de M. BECK, les fonctions *f* et *g* ont été supposées linéaires en *x* et *y* :

$$\begin{cases} x' = \dfrac{dx}{dt} = f(x, y) = ax + by \\ y' = \dfrac{dy}{dt} = g(x, y) = cx + dy \end{cases}$$

Sur votre page du site Panamaths, vous trouverez un fichier Python (**EulerB2_base.py**) correspondant à un « programme de base » à modifier progressivement dans ce devoir (vous créerez un nouveau fichier **EulerB2_lin_*nom*.py** où ***nom*** correspondra à votre nom de famille. Votre code devra être clairement commenté.). Il met en œuvre le schéma d'Euler ci-dessus et permet d'obtenir trois courbes intégrales du système associées à trois conditions initiales différentes.

Les différentes situations (valeurs propres réelles, complexes) correspondant aux diverses configurations de la matrice A ont été « codées en dur » dans le programme. Une seule d'entre elles est bien sûr prise en compte lors d'une exécution, les autres apparaissant dans des commentaires. Pour changer de configuration, placer en commentaires la configuration courante et décommenter la configuration vous intéressant.

Dans un premier temps, vous testerez les diverses configurations proposées.

Travail demandé

1. Dans le programme de base, le pas est demandé à l'utilisateur. Ainsi, pour des conditions initiales données, on obtient seulement une « moitié » de la courbe intégrale correspondante.

   Comment doit-on modifier le programme de base pour obtenir « toute » la courbe ?

   Vous modifierez le programme de base en commentant clairement le(s) changement(s) apporté(s). Les branches correspondant à la valeur positive du pas seront affichées dans une certaine couleur et celles correspondant à la valeur négative du pas seront affichées dans une autre couleur.

2. Dans certains cas, on est naturellement conduit à faire apparaître un repère local d'origine le point $(0\,;0)$ et d'axes des droites de vecteurs directeurs des vecteurs propres $\vec{u}$ et $\vec{v}$ associés à la matrice A.

   Modifier votre programme de sorte que les axes (on attend de « simples » droites, sans graduations) de ce repère local soient affichés (dans une couleur spécifique, par exemple, pour les distinguer clairement des courbes intégrales).

3. Vous avez pu constater que pyplot adapte automatiquement les échelles de la fenêtre graphique de façon à tenir compte des valeurs minimales et maximales (calculées) des abscisses et des ordonnées des points des courbes intégrales.

   On peut cependant souhaiter imposer les dimensions de la fenêtre de visualisation :

   $$\left[x_{\min}\,;x_{\max}\right]\times\left[y_{\min}\,;y_{\max}\right]$$

   On peut alors choisir un pas et calculer, pour chaque branche d'une courbe intégrale, un nombre maximum $N_{\max}$ (fixé par l'utilisateur) de points $P(x,y)$ tels que $x\in\left[x_{\min}\,;x_{\max}\right]$

et $y \in \left[y_{\min} \,; y_{\max} \right]$. Il se peut cependant que, à l'issue des calculs, le nombre de points calculés soit insuffisant, disons inférieur à un seuil $N_{\min}$ (également fixé par l'utilisateur). On divisera alors le pas par 2 et on recommencera les calculs pour générer une nouvelle série de points.

Apporter de nouvelles modifications à votre programme de sorte que :

- Les dimensions de la fenêtre soient imposées (demandées à l'utilisateur ou directement écrites dans votre code).
- Chaque branche de chaque courbe intégrale comporte un nombre de points compris entre $N_{\min}$ et $N_{\max}$.

Etude d'un système non linéaire

Dans cette partie, on se propose d'étudier le système suivant :

$$\begin{cases} x' = \cos(\pi x) \times \ln(y) \\ y' = \ln(x) \times \cos(\pi y) \end{cases}$$

autour du point singulier $(1\,;1)$.

On demande de reprendre le programme de base initial et le modifier (vous le sauvegarderez cette fois dans le fichier **EulerB2_NL_*nom*.py**) pour qu'il affiche les courbes intégrales passant par les points $(1\,;1+0,05k)$ et $(1+0,05k\,;1)$ avec $k \in [\![1\,,8]\!]$ puis $k \in [\![1\,,18]\!]$ (on choisira, comme suggéré dans le code, un pas de 0,01).