TD machine 3 - Mose 1003

E. Ringeisen

Octobre 2015

Tracé d'une fonction de deux variables

Démarrez le programme Scilab et l'éditeur SciNotes comme il a été vu aux TD précédents. Sauvegardez un nouveau fichier de commandes dont le nom se termine par `.sce`, et en haut de ce fichier, mettez les instructions

```
clear;
xdel(winsid()); // détruit les fenêtres graphiques
```

On s'intéresse maintenant à la fonction de deux variables

$$f(x, y) = \frac{3\cos(x-y)}{1+x^2}$$

En scilab, on peut définir cette fonction de la façon suivante

```
function v = f(x, y)
    v = 3 * cos(x - y) ./ (1 + x.^2);
endfunction
```

On remarque l'utilisation des opérateurs terme à terme `./` et `.^` dans cette expression. Les utiliser permet d'appeler la fonction non seulement avec deux valeurs réelles x et y, mais aussi avec deux matrices de mêmes tailles de façon à évaluer la fonction en plusieurs points simultanément.
Avant de tracer la surface représentative de la fonction f, on commence par créer une fenêtre graphique

```
fenetre = figure("Figure_name", "Surface Représentative",..
                            "position", [100, 50, 1000, 600]);
set("current_figure", fenetre);
fenetre.color_map = autumncolormap(32);
fenetre.BackgroundColor = name2rgb('white')/256;
```

L'instruction (optionnelle) qui fixe la propriété `fenetre.color_map` permet de choisir la palette de couleurs qui sera utilisée pour tracer la surface. Une liste de palettes est disponible sur la page internet http://help.scilab.org/docs/5.5.2/fr_FR/index.html à la rubrique *Gestion des couleurs*.
Pour le tracé proprement dit, on commence par définir les intervalles des variables x et y sur lesquels on veut tracer, et on crée deux matrices discrétisant ces intervalles

```
x = linspace(-3,3,50);
y = linspace(-2,4,50);
```

Enfin, on invoque la fonction `fplot3d1()` qui réalise le tracé

```
fplot3d1(x, y, f);
```

De même que dans un graphique bidimensionnel, on peut décorer le tracé en utilisant le système de coordonnées. Commentez les instructions

```
-->coor = get("current_axes");
-->coor.x_label.text = "Premier axe (x)";
-->coor.x_label.font_size = 2;
-->coor.y_label.text = "Second axe (y)";
-->coor.y_label.font_size = 2;
-->coor.z_label.text = "$\frac{3 \cos(x-y)}{1+x^2}$";
-->coor.z_label.font_size = 2;
```

Sur la fenêtre graphique, en cliquant avec le bouton droit de la souris et en déplaçant la souris, on parvient à modifier la direction d'où elle est observée.

Création d'une image

On peut créer un fichier image de la façon suivante

```
xs2jpg(fenetre, "~/TPmose1003/image001.jpg");
```

Vérifiez que votre dossier `TPmose1003` contient bien l'image en question.

Un point selle (ou col)

Soit maintenant la fonction

$$g\left(x,y\right)=\frac{\left(2x-1\right)y}{x^{2}-x+1}+\cos\left(\pi x\right)$$

On vérifie assez facilement que $g\left(x,y\right)$ admet un unique point critique $\left(x_{c},y_{c}\right)=\left(1/2,3\pi/8\right)$.

Tracé au voisinage du point critique

On ouvre une nouvelle fenêtre graphique

```
fenetre2 = figure("Figure_name", "Quelques surfaces", "position", [150, 100, 1000, 600]);
set("current_figure", fenetre2);
fenetre2.color_map = jetcolormap(32);
fenetre2.BackgroundColor = name2rgb('white')/256;
```

On discrétise deux intervalles en x et y, centrés sur les coordonnées critiques

```
xc = 0.5;
yc = 3 * %pi / 8;
sx = 0.8;
sy = 1.0;
x = linspace(xc-sx, xc+sx, 30);
y = linspace(yc-sy, yc+sy, 30);
```

on définit la fonction g

```
function v = ggg(x, y)
    v = (2*x-1).*y./(x.^2-x+1) + cos(%pi * x);
endfunction
```

et on trace la fonction dans un premier subplot

```
subplot(2,2,1);
fplot3d1(x, y, ggg, theta=120, alpha=20);
```

On a utilisé ici deux arguments supplémentaire `theta` et `alpha` de la fonction `fplot3d1()`, qui sont deux angles permettant d'indiquer la direction où se trouve l'observateur dans l'espace (changer θ revient à tourner autour de l'axe Oz, changer α revient à regarder plus ou moins *par au dessus*).

Tracé de courbes de niveau

La fonction `coutour2d()` permet de tracer des courbes de niveau de la fonction de deux variables. Son quatrième argument est le nombre de courbes de niveau voulus (on peut également passer la liste des valeurs de la fonctions qu'on veut tracer)

```
subplot(2,2,2);
contour2d(x, y, ggg, 15);

fcoor = get("current_axes");
coor.title.text = "Quelques courbes de niveau";
coor.title.font_size = 3;
```

Equation différentielle

On considère maintenant le problème de Cauchy pour l'équation différentielle $y'(x) = g(x, y(x))$, c'est à dire

$$\begin{cases} y'(x) = \dfrac{(2x-1)}{x^2 - x + 1} y(x) + \cos(\pi x) \\ y(a) = u \end{cases}$$

La fonction `ode()` de scilab permet de calculer des solutions de cette équation. On commence par se donner un intervalle en x sur lequel on veut résoudre, et on le discrétise

```
intervalle = linspace(-1, 3, 100);
a = intervalle(1);
```

Le point a est le début de l'intervalle. On va résoudre simultanément pour plusieurs valeurs de $u = y(a)$, qu'on met dans une matrice ligne

```
cauchy = [-3 -1.5 0 1.5 3 4.5];
```

La matrice résultat aura autant de colonnes que de valeurs de u, et chaque colonne sera une solution $y(x)$ sur l'intervalle

```
resultat = zeros(length(intervalle), length(cauchy));
```

La boucle suivante remplit la matrice résultat, commentez chaque ligne de ce morceau de code

```
i = 1;
for u = cauchy
    y = ode(u, a, intervalle, ggg);
    resultat(:,i) = y';
    i = i + 1;
end
```

On peut maintenant tracer les solutions dans un troisième subplot

```
subplot(2,2,3);
plot2d(intervalle', resultat);
```

A vous de jouer

Soit l'équation différentielle

$$\begin{cases} y'(x) = \dfrac{1}{x} y(x) + x \log(x) \\ y(1) = u \end{cases}$$

Résolvez numériquement cette équation différentielle sur l'intervalle $I_x = [1, 4]$ en choisissant une dizaine de valeurs de u régulièrement espacées entre -2 et 2. Tracez les courbes obtenues dans le quatrième subplot.

Surface

On peut effacer le premier subplot de la façon suivante

```
subplot(2, 2, 1);
coor = get("current_axes");
delete(coor);
subplot(2,2,1);
coor = get("current_axes");
```

Dans le subplot ainsi libéré, tracez la surface représentative de la fonction $(x, y) \mapsto \dfrac{y}{x} + x \log(x)$ sur le domaine $[0.1, 0.6] \times [-1, 1]$. On pourra choisir l'angle de vue $\theta = -100, \alpha = 10$.
De même, après avoir effacé le second subplot, y tracer un ensemble de lignes de niveau de la même fonction.