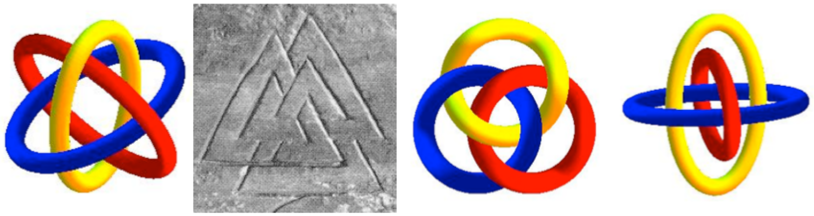
Examen LA207
Matlab: applications en mécanique
Jeudi 26 mai 2010

Tous les graphiques doivent être annotés: titres, labels des axes et légendes. Les scripts doivent être insérés dans le compte-rendu auprès des graphiques associés. Chaque bloc de commande des scripts doit être commenté. La notation prendra en compte la qualité de la présentation.

1 Sujet 1

1.1 Anneaux de Borromée

Compétences: *Tracer des courbes paramétrées en 3D avec la fonction* `plot3`. *Représentation selon différents angles de vue.*



Dans cet exercice, nous allons visualiser des entrelacs de courbes en trois dimensions avec la fonction `plot3`.

Les anneaux de Borromée tirent leur nom d'une célèbre famille de princes italiens de la Renaissance, les Borromée, qui les adoptèrent comme symbole héraldique. Ils sont gravés dans la pierre de leur château, sur l'une des îles Borromée du lac Majeur (isola bella), dans le nord de l'Italie. (www.mathcurve.com).

1. **Tracé des anneaux:** Voici les équations paramétriques des trois anneaux entrelacés, il s'agit de trois ellipses de grand et petit axes de longueur a et b:

$$\Gamma_1 \left\{ \begin{array}{l} x = a\cos(t), \\ y = b\sin(t), \\ z = 0 \end{array} \right. \qquad \Gamma_2 \left\{ \begin{array}{l} x = 0, \\ y = a\cos(t), \\ z = b\sin(t) \end{array} \right. \qquad \Gamma_3 \left\{ \begin{array}{l} x = b\sin(t), \\ y = 0, \\ z = a\cos(t) \end{array} \right.$$

   Pour $t \in [0, 2\pi]$, tracez les trois courbes $\Gamma_1, \Gamma_2, \Gamma_3$ avec trois couleurs différentes. Vous prendrez $a = 1$ et $b = 2$. Vous utiliserez `axis equal`, `grid on`, `box on` pour faciliter la visualisation. Dans quatre sous-graphiques affichez l'entrelac obtenu selon quatre points de vues différents, qui montrent bien comment les anneaux sont entrelacés.

1.2 Convergence de séries

Compétences: *Utilisation d'une boucle* `for`*, avec mémorisation de l'évolution de la somme partielle, et tracé de l'évolution des éléments de la série. Comparaison de l'évolution de la série avec un critère de convergence.*

Dans cet exercice nous étudions la convergence des séries

$$S_n = \sum_1^n u_n.$$

Ici S_n est la somme partielle, et u_n est le terme général.

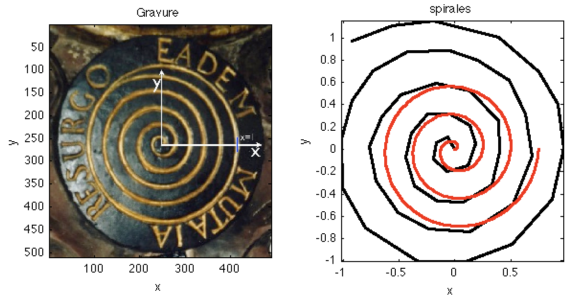
Nous avons un critère de convergence pour les séries à termes strictement positif: le critère de D'Alembert, qui dit que si $D_n = u_{n+1}/u_n$ tend vers une valeur finie $L < 1$, alors la série est convergente. Nous considérons la série géométrique

$$u_n = \frac{1}{3^n}$$

Tracez un graphique qui montre comment S_n, u_n, D_n dépendent de n. Les courbes seront en lignes continues de différentes couleurs avec des marqueurs `*`. Vous choisirez une valeur maximum de n de sorte à ce que l'on puisse bien voir les variations de ces variables. Sur ce graphique, vous tracerez une ligne horizontale noire pointillées qui permet de distinguer si le critère de D'Alembert donne une série convergente.

1.3 La spirale de Bernoulli

Compétences: *mesurer des points sur une image, transformer les coordonnées vers un référentiel donné. Tracé d'une courbe dont l'équation est donnée en coordonnées cylindriques.*



Jacques Bernoulli a demandé à faire graver sur sa tombe une spirale logarithmique

$$r = ab^{\theta}$$

en coordonnées cylindriques (r, θ), avec a et b deux constantes numériques. Il a demandé que cette phrase soit gravée: *eadem mutata resurgo*, "déplacée (mutata), je réapparais (resurgo) à l'identique". C'est en effet une propriété de la spirale logarithmique: une rotation est équivalente à une homotéthie. Malheureusement, le graveur qui n'était pas géomètre à réalisé une spirale Archimédienne

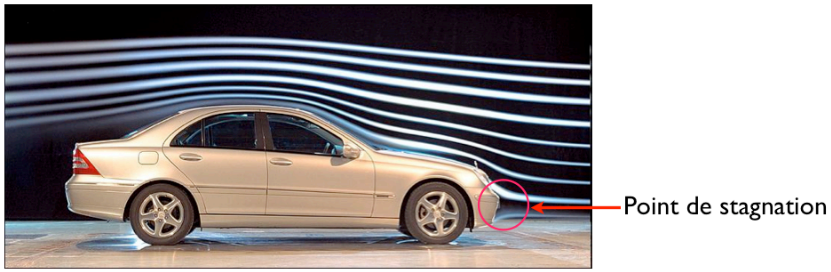
$$r = a\theta$$

qui ne vérifie pas la propriété de conservation par homotéthie. Dans cet exercice, nous allons vérifier graphiquement que la courbe est proche d'une spirale Archimédienne.

1. **Image:** L'image est stockée sur le disque sous le fichier `spirale.jpg`. Chargez l'image avec la fonction `imread` et affichez l'image avec la fonction `image`.
2. **Mesure:** Mesurer une vingtaine de points de la spirale grâce à l'outils d'étiquetage. Pour gagner du temps, vous ne considérerez que trois tours de spirale à partir du centre. Enregistrez ces points dans deux tableaux `x` et `y`.
3. **Changement de référentiel:** Transformez les coordonnées mesurées en pixels de sorte à ce que les coordonnées soient comme sur le référentiel tracé en blanc sur l'image: origine du repère au centre de la spirale, et rayon 1 pour le cercle autour de la spirale.
4. **Affichage:** Vous tracerez dans deux sous-graphiques: d'une part à gauche, l'image originale, et à droite la courbe mesurée, dans le bon référentiel.
5. **Spirale théorique:** Pour vérifier que la spirale gravée est une spirale Archimédienne, nous allons superposer une spirale Archimédienne théorique sur la courbe mesurée. Vous choisirez la paramètre a de sorte à ce qu'il y ai une bonne ressemblance. Indication: le rayon r est égal a à peu près 1 pour 5 tours de spirale.

1.4 Cinématique d'un fluide: point de stagnation

Compétences: *Représentation d'une fonction de deux variables: champ de vecteur, isovaleurs, surface. Utilisation de* `meshgrid` *pour construire une grille cartésienne. Opérations de tableaux élément par élément.*



Nous allons étudier le mouvement d'un fluide, décrit par le champ de vitesse

$$u = -Ux \quad v = Uy$$

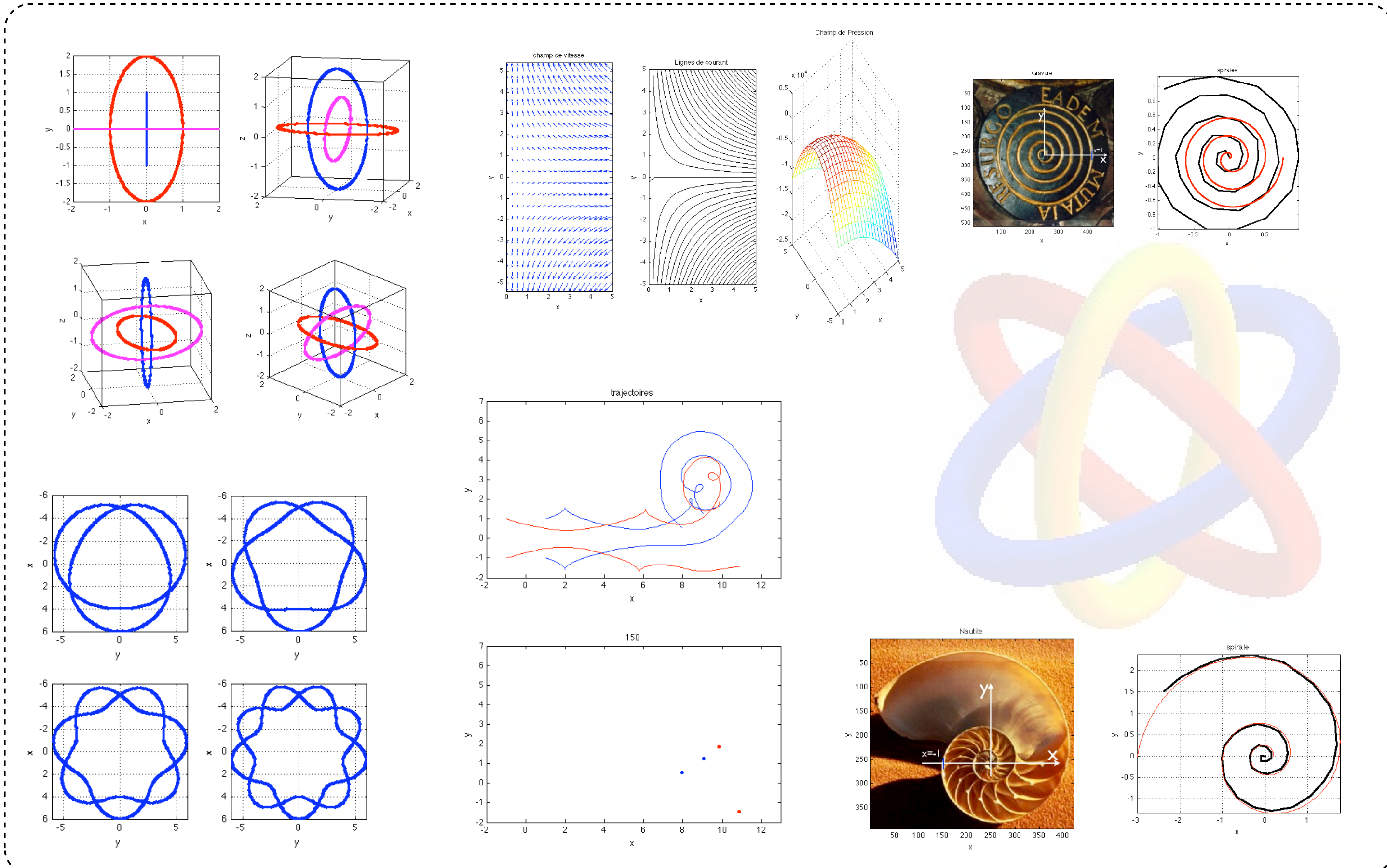
dans un repère cartésien (x, y). Il s'agit du champ de vitesse d'un fluide incompressible proche d'un point de stagnation. L'image ci-dessus représente l'écoulement de l'air autour d'une voiture dans une soufflerie, visualisé par des lignes d'émission de fumée.

1. **Champ de vitesse:** Tracer le champ de vitesse avec `quiver` et `meshgrid`. Vous prendrez $x \in [0, L], y \in [-L, L]$ avec $L = 5$. Au graphique vous ajouterez une ligne verticale noire pointillée avec `plot` pour $x = 0$ qui représente la paroi sur laquelle le fluide vient impacter. C'est l'équivalent pour l'image ci-dessus du pare-chocs de la voiture.
2. **Fonction de courant:** La fonction de courant qui correspond à ce champ de vitesse est
$$\phi(x, y) = Uxy.$$
Les lignes iso-valeurs de la fonction de courant correspondent aux lignes de courant: les lignes qui sont en tout point tangentes au vecteur vitesse. Tracez les lignes de courant avec la fonction `contour`. Vous choisirez un nombre de contours qui permette une bonne visualisation de la structure de notre écoulement (Pour une description des arguments d'entrée de `contour`, tapez `help contour`).
3. **Pression:** Nous allons maintenant considérer le champ de pression de cet écoulement. Cet écoulement est irrotationel, donc la pression est décrite en tout point par l'équation de Bernoulli
$$P + \frac{1}{2}\rho U^2 = p + \frac{1}{2}\rho(u^2 + v^2)$$
ou P et et U sont la pression et la vitesse à l'infini (loin de la plaque), et ρ la densité de notre fluide. Nous prendrons $P = 1$ et $\rho = 1000$. Nous avons ainsi
$$p(x, y) = \frac{1}{2}\rho[U^2 - u(x, y)^2 - v(x, y)^2].$$
Dans un troisième sous-graphique, tracez la pression en fonction de x et y avec la fonction `mesh`. Vous choisirez un point de vue tel que l'on observe bien que le point de stagnation est le point où la pression est la plus élevée (vitesse faible implique pression grande selon l'équation de Bernoulli).

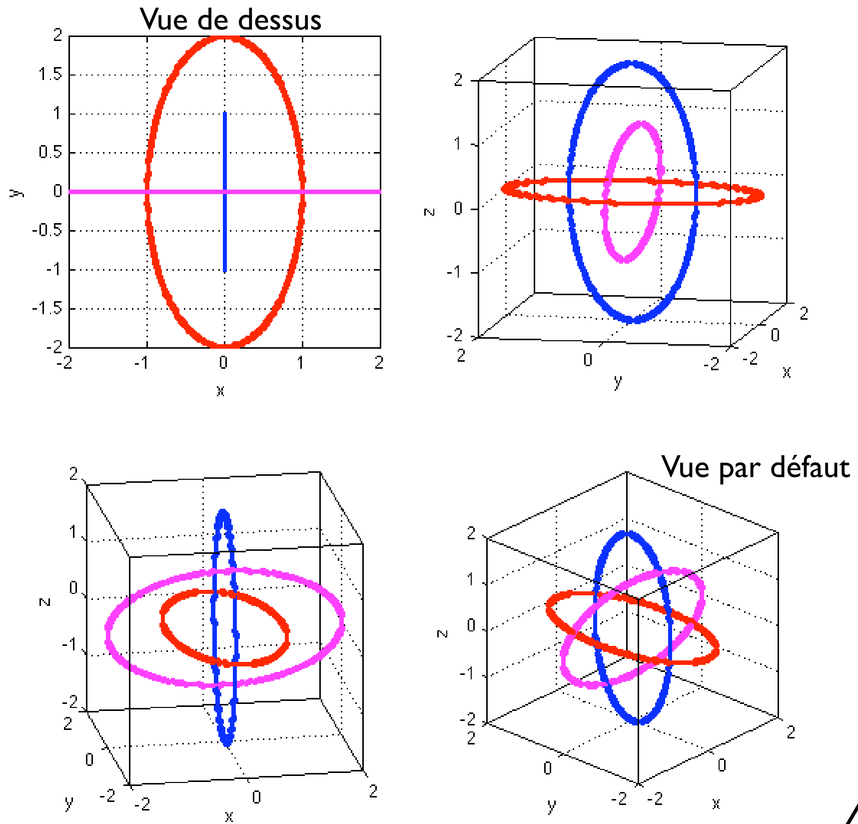
Matlab : applications en mécanique
LA207, 2010

http://www.lmm.jussieu.fr/~hoepffner/enseignement

Compte-rendu examen



1.1 Anneaux de Borromée



Quatre point de vues différents pour bien voir la structure entrelacée de ces trois anneaux elliptiques.

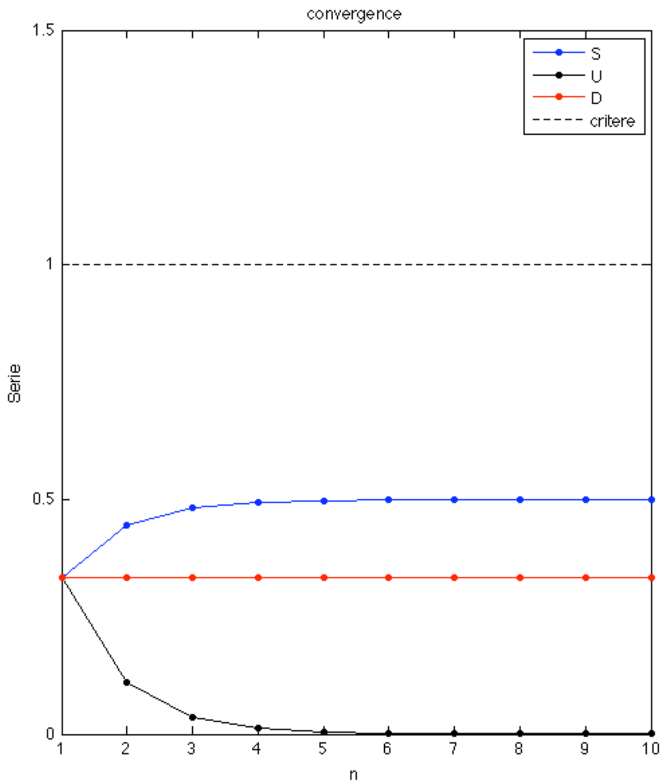
Dans le code, on trace quatre fois exactement la même figure, et on change le point de vue ensuite avec la souris et l'outils de rotation en 3D du menu de la fenêtre de graphique.

Script

```
% parametres
a=1; b=2

% boucle pour quatre sous-graphiques
for ind=1:4
subplot(2,2,ind)

% premier anneau
x=a*cos(t);
y=b*sin(t);
z=0*t;
plot3(x,y,z,'linewidth',3,'color','r');hold on

% second anneau
x=0*t;
y=a*cos(t);
z=b*sin(t);
plot3(x,y,z,'linewidth',3,'color','b');hold on

% troisieme anneau
x=b*sin(t);
y=0*t;
z=a*cos(t);
plot3(x,y,z,'linewidth',3,'color','m');hold on

% commandes de graphiques
grid on; box on; axis equal
rr=2;xlim([-1,1]*rr);ylim([-1,1]*rr);zlim([-1,1]*rr);
xlabel('x'); ylabel('y'); zlabel('z')
hold off

end
```

1.2 Convergence de série

Critère de D'Alembert.



Le terme général tend vers zéro, le critère de D'Alembert donne une valeur constante, plus petite que 1, et la somme partielle tend bien vers une constante.

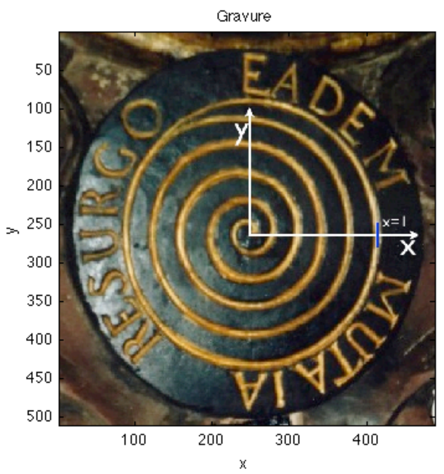
Il faut garder en mémoire la valeur du u précédent pour pouvoir calculer le critère de D'Alembert

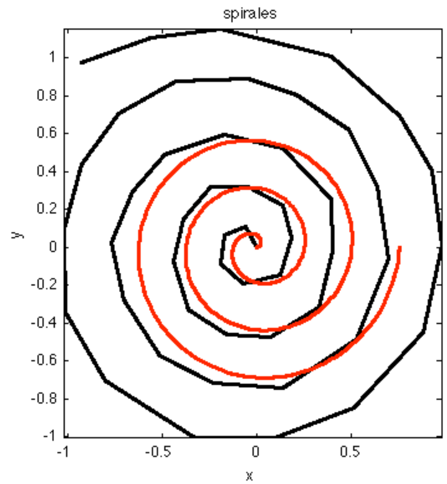
Une ligne horizontale avec plot, y=1 en ligne pointillée

Script

```
% Nombre d'itérations pour la série
nmax=10;

% initialisation des tableaux de memorisation
S=zeros(nmax,1);
U=zeros(nmax,1);
D=zeros(nmax,1);

% boucle de la serie
s=0;
u=1;
for n=1:nmax
    uu=u; % on garde le u précédent

    u=1/(3^n); % terme general
    s=s+u; % somme partielle

    % memorisation
    S(n)=s;
    U(n)=u;
    D(n)=u/uu;
end

% on trce les graphiques
nvec=1:nmax;
plot(nvec,S,'b.-',nvec,U,'k.-',nvec,D,'r.-');
hold on

% une ligne horizontale pour le critere de convergence
plot([1,nmax],[1,1],'k--')
legend('S','U','D','critere')
xlabel('n'); ylabel('Serie'); title('convergence')

ylim([0,1.5])
hold off
```

1.3 Spirale de Bernoulli





Noir: points de mesure,
rouge: spirale théorique

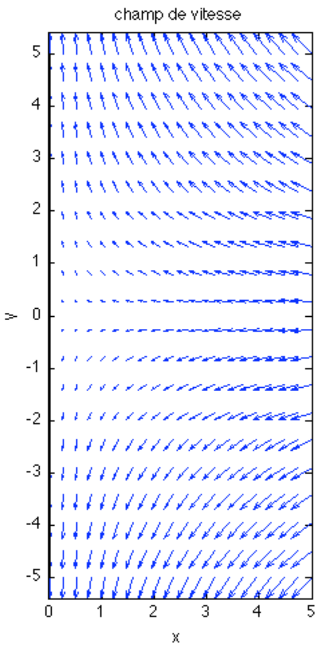
Les étapes de manipulation des points de mesure: recentrage: ici on soustrait à x et à y les coordonnées du premier point qui est au centre du référentiel. On pense aussi à inverser y car les pixels avec la fonction image sont ordonnés de haut en bas, et non de bas en haut comme pour plot. Ensuite on met à l’echelle une fois que la taille d’un pixel est mesurée grâce aux points de références sur l’image.

Script

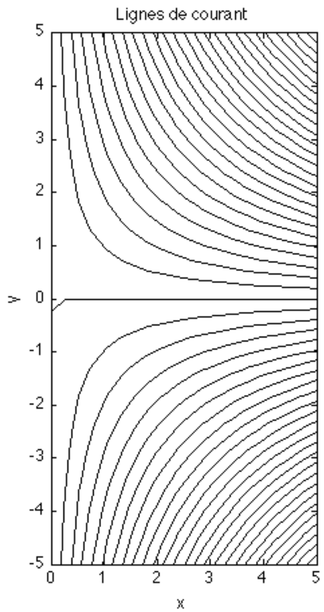
```
% spirales et mesures de points
a=imread('spirale.jpg');

subplot(1,2,1)
image(a)
axis equal; axis tight
xlabel('x'); ylabel('y');title('Gravure')

% points de mesure
d=[156.0044  160.7713
  151.3856  152.4575
  143.0718  156.1525
  141.2243  167.2375
  150.4619  175.5513
  165.2419  171.8563
  169.8607  157.0762
  166.1657  144.1437
  152.3094  136.7537
  137.5293  136.7537
  126.4443  149.6862
  122.7493  166.3138
  128.2918  185.7126
  143.9956  195.8739
  161.5469  196.7977
  180.9457  184.7889
  186.4883  162.6188
  185.5645  141.3724
  166.1657  121.0499
  143.0718  115.5073
  119.0543  123.8211
  106.1217  138.6012
   97.8079  158.9238
  102.4267  182.0176
  117.2067  204.1877
  138.4531  215.2727
  166.1657  217.1202
  195.7258  197.7214
  208.6584  165.3900
  204.0396  136.7537
  192.9545  113.6598
  171.7082   99.8035
  152.3094   93.3372
  123.6730   94.2610
  100.5792  107.1935
   85.7991  127.5161
   78.4091  153.3812
   79.3328  186.6364
   95.0367  214.3490
  131.0630  237.4428
  163.3944  237.4428
  194.8021  225.4340
  222.5147  194.9501
  229.9047  159.8475
  226.2097  130.2874
  213.2771  108.1173
  185.5645   84.0997
  141.2243   73.0147
  113.5117   76.7097
   85.7991   86.8710];

% manipulation des coordonn?es
x=d(:,1); y=-d(:,2);
x=x-x(1); y=y-y(1); % centrage du referenciel
pix=1/(230-154); % taille d'un pixel
x=x*pix; y=y*pix; % mise a l'echelle

subplot(1,2,2);
% trace les points de mesure
plot(x,y,'k','linewidth',2)
hold on

% spirale Archimedienne
t=linspace(0,3*2*pi,400);
a=1/(8*pi);
r=a*t;
xx=r.*cos(t); yy=r.*sin(t); % coordonnees cartesiennes
plot(xx,yy,'r','linewidth',2);
axis equal; axis tight
xlabel('x'); ylabel('y');title('spirales')
```

1.4 Cinématique d’un fluide:

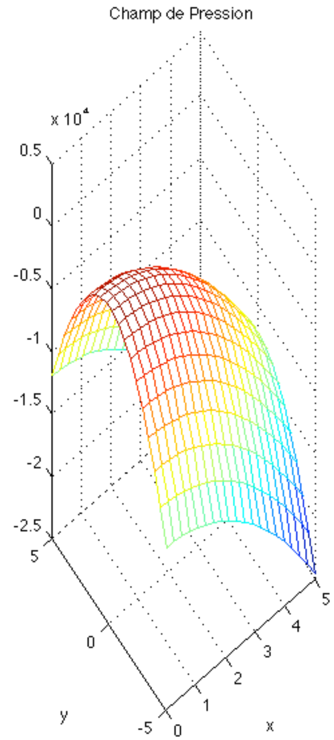
Point de stagnation



Voici le champ de vitesse. On voit bien que la vitesse s’anulle au point (0,0), c’est le point de stagnation.



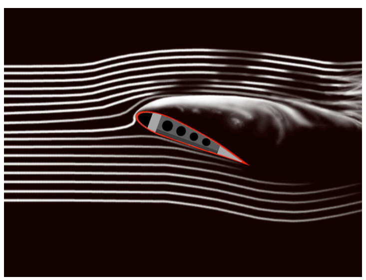
Les lignes iso-valeur de la fonction de courant, ce sont les lignes de courant. On voit bien que ces lignes correspondent au champ de vitesse représenté à gauche avec quiver.



Une représentation en 3D du champ de pression, que nous avons déduit du champ de vitesse avec la loi de bernoulli. Attention ici à bien utiliser des opérations élément par élément pour le calcul de la puissance au carré: X.^2+Y.^2. On voit ici que le point de stagnation a la pression la plus élevée.

```
L=5; % parametre de taille
x=linspace(0,L,20);
y=linspace(-L,L,20);
[X,Y]=meshgrid(x,y);
U=1;

% champ de vitesse
u=-U*X;
v=U*Y;

% trace le champ de vitesse avec quiver
subplot(1,3,1);
quiver(X,Y,u,v,1);
hold on

% on rajoute une ligne verticale
plot([0,0],[-L,L],'k','linewidth',2)
xlabel('x');ylabel('y');title('champ de vitesse')
axis equal; axis tight


% la fonction de courant
phi=U*X.*Y;
subplot(1,3,2);
contour(X,Y,phi,51,'k')
hold on

% on rajoute la ligne verticale
plot([0,0],[-L,L],'k','linewidth',2)
xlabel('x');ylabel('y');title('Lignes de courant')
axis equal; axis tight


% la pression avec bernoulli
P=0.5*1000*U^2*(1-X.^2-Y.^2);
subplot(1,3,3);
mesh(X,Y,P)
xlabel('x');ylabel('y');title('Champ de Pression')
```

Script



Point de stagnation sur le bord d’attaque dun profil d’aile

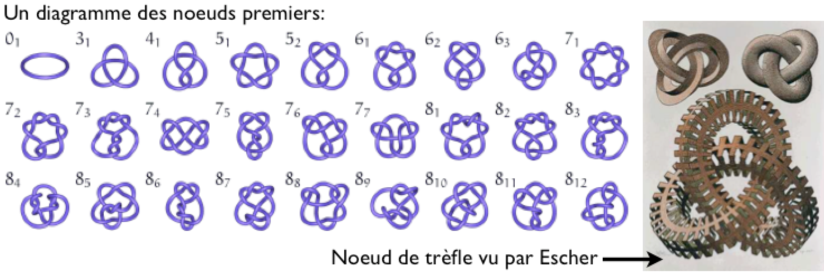
Examen LA207
Matlab: applications en mécanique
Jeudi 26 mai 2010

Tous les graphiques doivent être annotés: titres, labels des axes et légendes. Les scripts doivent être insérés dans le compte-rendu auprès des graphiques associés. Chaque bloc de commande des scripts doit être commenté. La notation prendra en compte la qualité de la présentation.

2 Sujet 2

2.1 Noeuds

Compétences: *Tracer des courbes paramétrées en 3D avec la fonction* `plot3`. *Représentation selon différents angles de vue.*



La théorie des nœuds a une longue histoire. On peut peut-être la faire commencer avec des travaux de Gauss liés à l'électromagnétisme. Les nœuds furent étudiés par Carl Friedrich Gauss qui a introduit une formule intégrale calculant le "nombre de liaison" entre deux nœuds. Son étudiant Johann Benedict Listing a poursuivi leur étude. La première étude poussée est survenue plus tard, lorsque William Thomson (Lord Kelvin) proposa une théorie des atomes vortex.

Nous allons considérer ici des noeuds qui sont construits comme une ligne s'enroulant régulièrement autour d'un tore. Voici une paramétrisation cartésienne d'une famille de telles courbes:

$$\begin{cases} x = [R + r\cos(nt)]\cos(t), \\ y = [R + r\cos(nt)]\sin(t), \\ x = r\sin(nt), \end{cases}$$

avec $t \in [0, 4\pi]$. Les paramètres sont R le grand rayon du tore et r son petit rayon. Nous prendrons $R = 3, r = 1$. Ici n est un paramètre que nous pouvons faire varier pour obtenir différents noeuds.

Tracé des noeuds: Dans quatre sous graphiques, tracez les courbes pour

les quatre valeurs du paramètre n:

$$\begin{cases} n = 3/2, \text{Noeud de trèfle}, \\ n = 5/2, \text{Pentagramme}, \\ n = 7/2, \text{Heptagramme}, \\ n = 9/2, \text{Nonagramme}, \end{cases}$$

Vous créerez deux graphiques: l'un ou les courbes sont vues du dessus (avec la commande `\view(90,90)`), et un second graphique pour lequel vous choisirez vous même l'angle de vue de sorte à bien montrer la structure tridimensionnelle des noeuds.

2.2 Convergence de séries

Compétences: *Utilisation d'une boucle* `for`, *avec mémorisation de l'évolution de la somme partielle, et tracé de l'évolution des éléments de la série. Comparaison de l'évolution de la série avec un critère de convergence.*

Dans cet exercice nous étudions la convergence des séries

$$S_n = \sum_1^n u_n.$$

Ici S_n est la somme partielle, et u_n est le terme général de la série.

Une série alternée est telle que son terme général change de signe à chaque itération: $u_n u_{n+1} \leq 0$. Le critère de Leibniz dit que si $\forall n, \|u_{n+1}\| \leq \|u_n\|$ (les termes décroissent en valeur absolue) et $u_n \to 0$ (le terme général tend vers zéro), alors la série est convergente. Pour la série de terme général

$$u_n = (-1)^n / \sqrt{n}$$

Tracez un graphique qui montre l'évolution de la somme partielle S_n, de la valeur absolue du terme général $\|u_n\|$, ainsi que du produit de deux termes consécutifs $u_n u_{n+1}$. Les courbes seront en lignes continues de différentes couleurs, avec des marqueurs `*`.

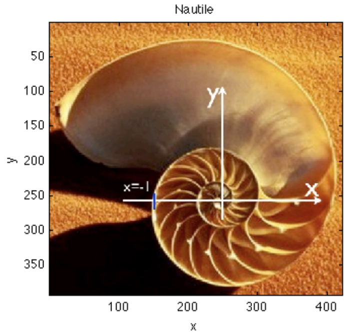
2.3 La coquille du Nautile

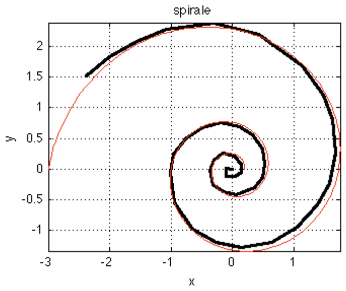
Compétences: *mesurer des points sur une image, transformer les coordonnées vers un référentiel donné. Tracé d'une courbe dont l'équation est donnée en cordonnées cylindriques.*

On trouve dans la nature des formes géométriques très simples. On observe dans la coquille du Nautile une forme de spirale, nous allons vérifier que c'est bien une spirale logarithmique d'équation

$$r = ab^\theta$$

en coordonnées cylindriques (r, θ), avec a et b deux paramètres numériques.





1. **Mesure:** Mesurer une vingtaine de points de la spirale grâce à l'outils d'étiquetage. Stockez ces points dans deux tableaux `x` et `y`.

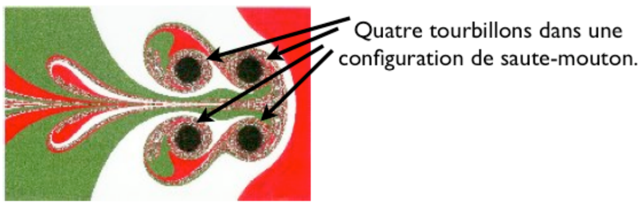
2. **Référentiel:** Vous manipulerez les coordonnées mesurées en pixels de sorte à ce que les coordonnées soient comme sur le référentiel tracé en blanc: origine du repère au centre de la spirale, et $x = -1$ pour le point indiqué sur l'image.

3. **Affichage:** Vous tracerez dans deux sous-graphiques: d'une part à gauche, l'image originale, et à droite la courbe mesurée, dans le bon référentiel.

4. **Spirale théorique:** Pour vérifier que la spirale gravée est une spirale logarithmique, nous allons superposer à la courbe tracée une spirale théorique. On observe que le rayon est multiplié par trois pour chaque tour de spirale, donc $b = \exp(\log(3)/(2\pi))$, et également que le rayon après deux tours est égal à 1, donc $a = 1/(b^{4\pi})$. Tracez cette spirale théorique sur le même sous-graphique que les points de mesure pour comparer les deux courbes.

2.4 Trajectoires de tourbillons

Compétences: *utilisation d'une fonction fournie avec description des arguments d'entrée et de sortie. Tracé de trajectoires dans le plan, animation du déplacement de points dans le plan.*



Nous avons vu dans un TP comment les tourbillons se déplacent sous l'influence les uns des autres. Nous allons dans cet exercice considérer le cas particulier de deux paires de tourbillons contrarotatifs. On peut créer une paire de tourbillons contrarotatifs en déplaçant une cuillère à la surface d'un bol de soupe, comme avec une rame. Pour créer deux paires, il suffit de mettre deux cuillères l'une derrière l'autre. Si l'on réalise cette expérience proprement, on peut observer que les deux paires de tourbillons jouent à saute-moutons: la paire de derrière, est aspirée entre les deux tourbillons de devant pour se retrouver elle-même devant et ainsi de suite. Nous allons étudier ce mouvement à l'aide de la fonction `tourbitraj`. Son usage: arguments d'entrée et de sortie est identique au TP sur les tourbillons.

1. **Trajectoires:** Calculer avec la fonction `tourbitraj`, les trajectoires de quatre tourbillons dont les positions et les intensités sont:

$$\begin{cases} T_1 : (x, y, \Gamma) = (1, 1, 1), \\ T_2 : (x, y, \Gamma) = (1, -1, -1), \\ T_3 : (x, y, \Gamma) = (-1, 1, 1), \\ T_4 : (x, y, \Gamma) = (-1, -1, -1), \end{cases}$$

   Tracez les trajectoires de ces quatre tourbillons dans un sous-graphique, avec une couleur différente pour chaque paire. Vous prendrez un temps final égal à 150 de sorte à observer plusieurs saute-moutons.

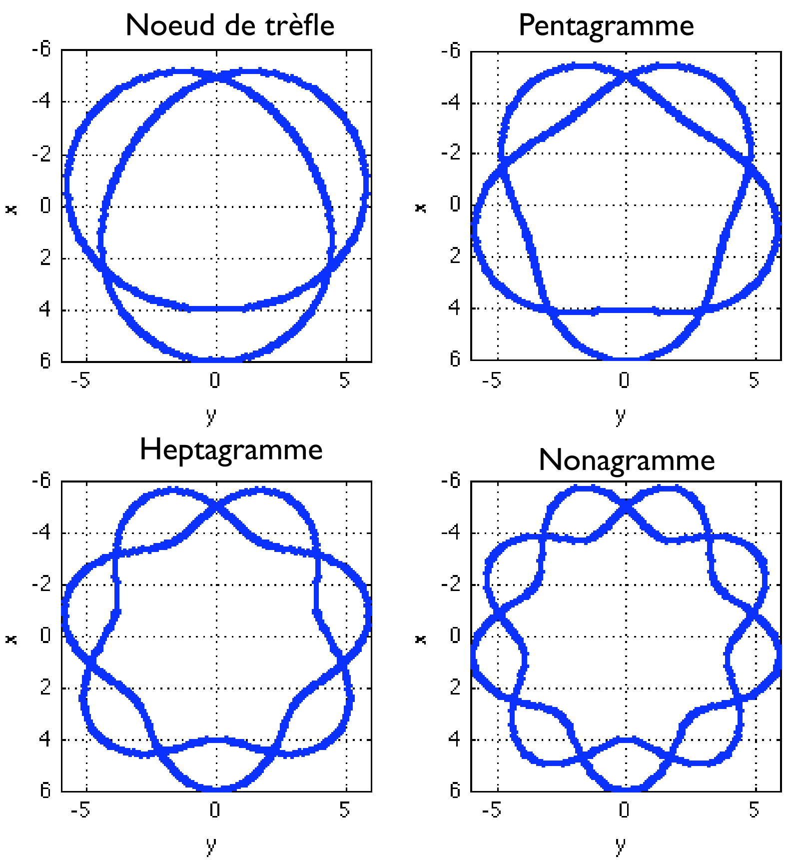
2. **Animation:** Ecrivez un code qui réalise l'animation du mouvement des tourbillons. Vous imposerez les limites des axes avec `xlim` et `ylim` de sorte à avoir sur le graphique un domaine fixe assez grand pour voir toute la trajectoire. Vous insérerez dans le compte-rendu la dernière image de l'animation dans un sous-graphique.

3. **Sensibilité:** Ce mouvement de saute-moutons est en fait difficile à réaliser avec des cuillères à soupe car le phénomène est très sensible à des petites imperfection de symétrie des paires de tourbillons. Réalisez les mêmes opérations et graphiques que pour les questions précédentes, mais avec une configuration légèrement dissymétrique: T_1, T_2, T_3 comme précédemment, mais pour le quatrième tourbillon:

$$T_4 : (x, y, \Gamma) = (-1, -1, -1.01),$$

   Vous adapterez les limites des axes de sorte à visualiser l'ensemble des trajectoires.

2.1 Noeuds



les quatre noeuds
vus de dessus

Script

```
% paramètres
R=5; r=1;

% les valeurs de n
nvec=[3/2,5/2,7/2,9/2];

% boucle sur les valeurs de n
for ind=1:length(nvec)
    n=nvec(ind);
    subplot(2,2,ind)

    % les courbes
    x=(R+r*cos(n*t)).*cos(t);
    y=(R+r*cos(n*t)).*sin(t);
    z=r*sin(n*t);

    % on trace
    plot3(x,y,z,'b','linewidth',3);
    xlabel('x'); ylabel('y'); zlabel('z');
    axis equal; box on; grid on
    xlim(6*[-1,1]);ylim(6*[-1,1]);
    view(90,90)
end
```

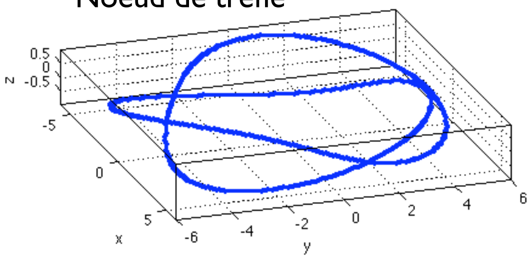
Ici pas besoin d'une boucle: on calcule les courbes en utilisant les manipulations de tableaux .*

Commande pour spécifier le point de vue

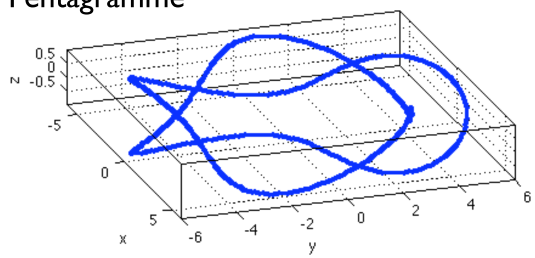
2.1 Noeuds

les quatre noeuds vus du même point de vue pour bien voir leur structure 3D

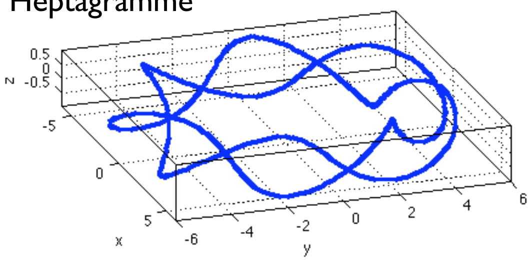
Noeud de trèfle



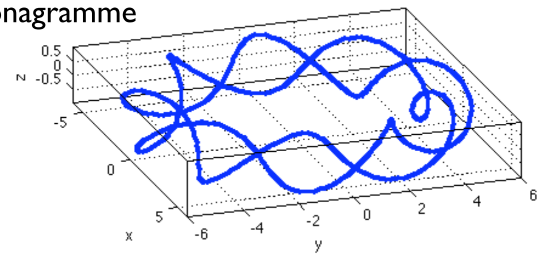
Pentagramme



Heptagramme



Nonagramme



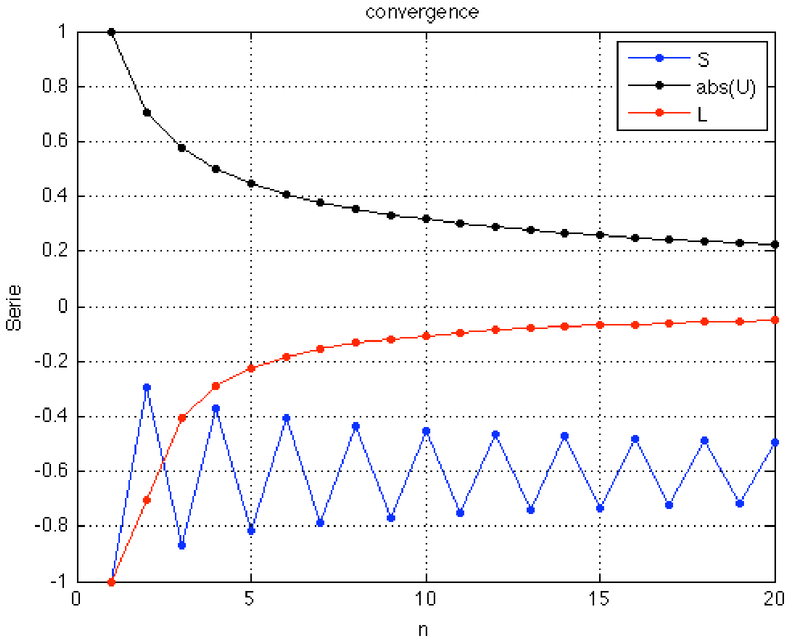
Script

```
% param?tres
R=5; r=1;

% les valeurs de n
nvec=[3/2,5/2,7/2,9/2];

% boucle sur les valeurs de n
for ind=1:length(nvec)
    n=nvec(ind);
    subplot(2,2,ind)

    % les courbes
    x=(R+r*cos(n*t)).*cos(t);
    y=(R+r*cos(n*t)).*sin(t);
    z=r*sin(n*t);

    % on trace
    plot3(x,y,z,'b','linewidth',3);
    xlabel('x'); ylabel('y'); zlabel('z');
    axis equal; box on; grid on
    xlim(6*[-1,1]);ylim(6*[-1,1]);
    view(71,20)
end
```

Commande pour spécifier le point de vue

2.2 Convergence de série

Série alternée.



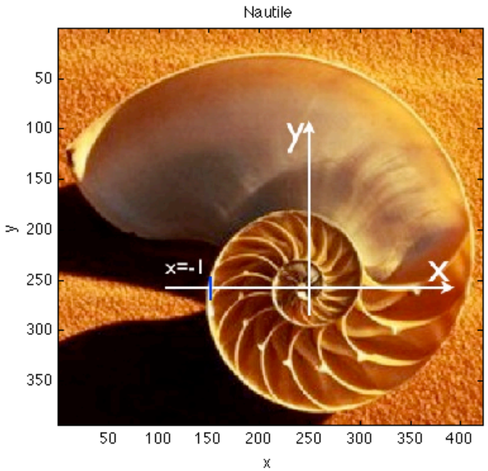
Le terme général tend bien vers zéro en valeur absolue, le produit de deux termes consécutifs est bien toujours négatif (série alternée), et on voit que la somme partielle S tend vers une constante.

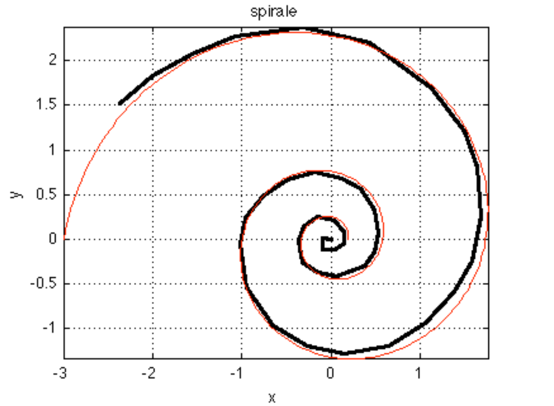
Il faut garder en mémoire la valeur du u précédent pour pouvoir calculer le produit de deux termes consécutifs.

Script

```
% nombre d'itérations pour la série
nmax=20;

% initialisation des tableaux de memorisation
S=zeros(nmax,1);
U=zeros(nmax,1);
L=zeros(nmax,1);

% boucle de la serie
s=0;
u=1;
for n=1:nmax
    uu=u; % on garde le u pr?c?dent

    u=(-1)^n/sqrt(n); % terme general
    s=s+u; % somme partielle

    % memorisation
    S(n)=s;
    U(n)=abs(u);
    L(n)=u*uu;
end

% on trce les graphiques
nvec=1:nmax;
plot(nvec,S,'b.-',nvec,U,'k.-',nvec,L,'r.-');
legend('S','abs(U)','L')
xlabel('n'); ylabel('Serie'); title('convergence')
grid on
```

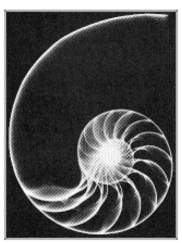
2.3 La coquille du Nautile

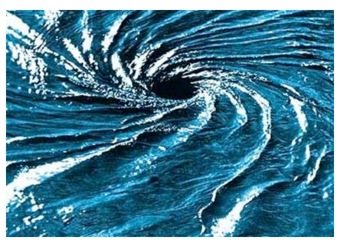




En rouge la spirale théorique et en noir les points de mesure.

Les étapes de manipulation des points de mesure: recentrage: ici on soustrait à x et à y les coordonnées du premier point qui est au centre du référentiel. On pense aussi à inverser y car les pixels avec la fonction image sont ordonnés de haut en bas, et non de bas en haut comme pour plot. Ensuite on met à l'echelle une fois que la taille d'un pixel est mesurée grâce aux points de références sur l'image.



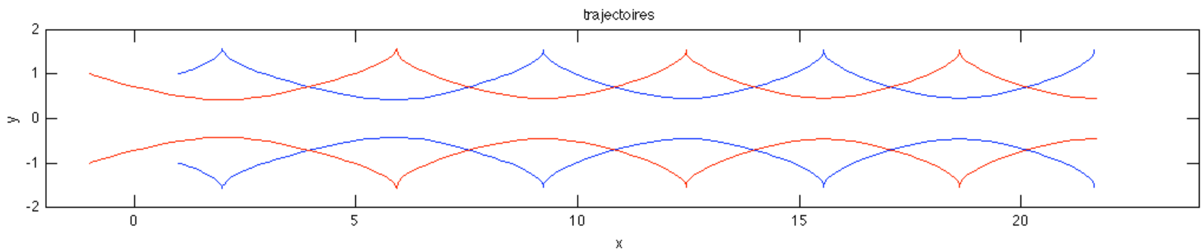


Script

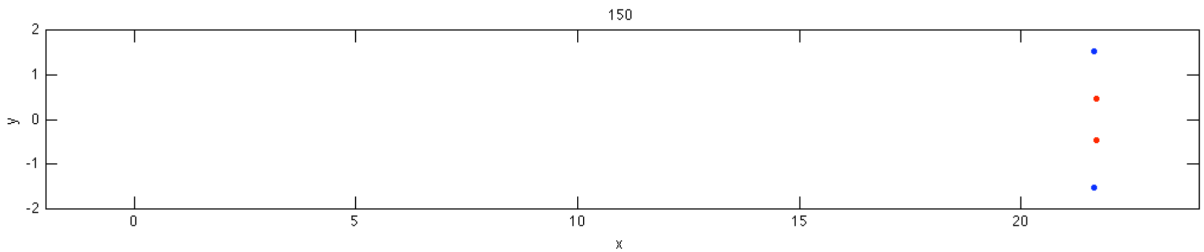
```
% lecture de l'image sur le disque
a=imread('coquillage.jpg');

% affichage de l'image
subplot(1,2,1)
image(a)
axis equal
axis tight
xlabel('x'); ylabel('y');title('Nautile')

% les points de mesure
d=[249.5605  256.6895
   240.0524  255.6331
   240.0524  268.3105
   253.7863  268.3105
   263.2944  260.9153
   263.2944  248.2379
   252.7298  235.5605
   234.7702  231.3347
   219.9798  242.9556
   213.6411  257.7460
   218.9234  280.9879
   234.7702  292.6089
   255.8992  296.8347
   285.4798  285.2137
   297.1008  270.4234
   301.3266  249.2944
   297.1008  224.9960
   281.2540  201.7540
   261.1815  190.1331
   231.6008  183.7944
   202.0202  191.1895
   175.6089  209.1492
   156.5927  233.4476
   151.3105  261.9718
   157.6492  309.5121
   186.1734  350.7137
   223.1492  372.8992
   264.3508  380.2944
   311.8911  371.8427
   352.0363  347.5444
   383.7298  312.6815
   402.7460  279.9315
   412.2540  231.3347
   408.0282  178.5121
   394.2944  139.4234
   359.4315   92.9395
   290.7621   43.2863
   219.9798   26.3831
   144.9718   35.8911
    98.4879   54.9073
    55.1734   79.2056
    19.2540  108.7863];

% Changement de referentiel
x=d(:,1); y=-d(:,2);
x=x-x(1); y=y-y(1); % centrage
pix=1/(248-151); % taille d'un pixel
x=x*pix; y=y*pix; % mise a l'echelle

% affichage des points de mesure
subplot(1,2,2);
plot(x,y,'k','linewidth',2)
hold on

% spirale Archimedienne
t=linspace(0,2.5*2*pi,400);
bb=exp(log(3)/(2*pi));
aa=1/(bb^(4*pi));
r=aa*bb.^(t+pi);

xx=r.*cos(t); yy=r.*sin(t); % coordonnees cartesiennes
plot(xx,yy,'r','linewidth',1);
axis equal; axis tight
grid on
xlabel('x'); ylabel('y');title('spirale')
```

2.4 Trajectoires de tourbillons

Questions 1 et 2.



Les trajectoires des quatre tourbillons: deux paires: la paire qui est initialement devant est en bleu et la paire qui est initialement derrière est en rouge. On observe bien le saute-mouton: ici pour t de 0 à 150, on a deux périodes et demi. Ceci est plus facile à voir sur l'animation



Ici l'animation de la position des deux paires de tourbillons.

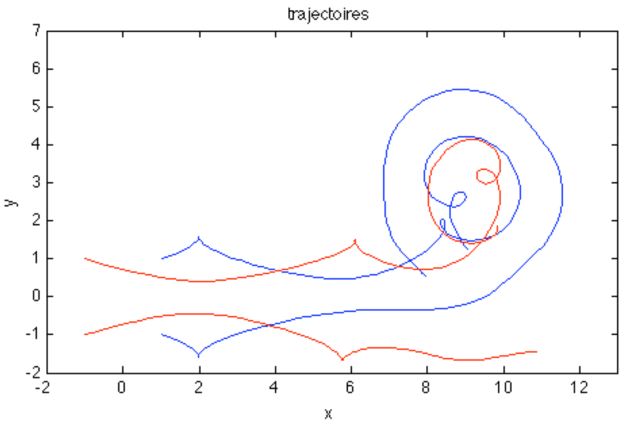
On utilise un tableau de caractères pour spécifier les couleurs: 'bbrr', bleu pour les deux premiers tourbillons et rouge pour les deux suivants.

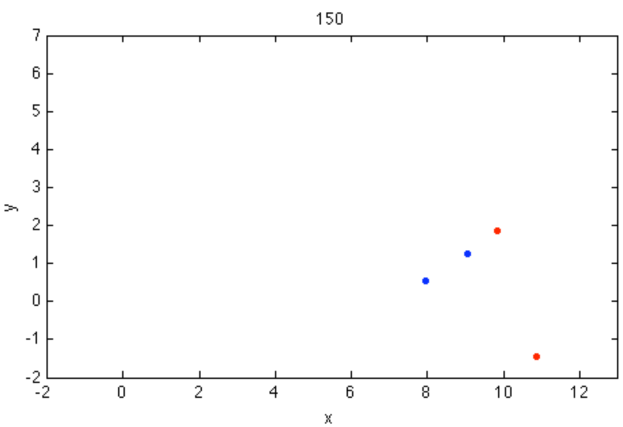
```
% configuration de tourbillons
  xpos=[1,1,-1,-1]; ypos=[1,-1,1,-1]; gamma=[1,-1,1,-1.0]; tmax=150;

  % calcul des trajectoires
  n=4; % nombre de tourbillons
  [xt,yt,t]=tourbitraj(xpos,ypos,gamma,tmax);

  % les trajectoires
  ccc='bbrr';
  subplot(2,1,1);
  for gre=1:n
      plot(xt(:,gre),yt(:,gre),ccc(gre)); hold on
  end
  axis equal; axis([-2,24,-2,2]);
  xlabel('x'); ylabel('y'); title('trajectoires');

  % animation du mouvement
  subplot(2,1,2);
  for ind=1:length(t)
      for gre=1:n
          plot(xt(ind,gre),yt(ind,gre),[ccc(gre) '.'],'markersize',10); hold on
      end
      hold off
      axis equal; axis([-2,24,-2,2]);
      xlabel('x'); ylabel('y'); title(t(ind));
      drawnow
  end
```

Script

2.4 Trajectoires de tourbillons

Question 3: sensibilité.





Pour l'animation: on affiche la dernière image. Pour le titre du graphique on prend la valeur du temps courant: t(ind)

On observe une demi-période de saute-mouton, puis les tourbillons s'enroulent vers la gauche. Et pourtant, la perturbation sur l'intensité du quatrième tourbillon était très faible: la configuration est très sensible aux perturbations.

Pour les deux sous-graphiques, on a adapté les limites des axes avec la commande axis([]), et avec axis equal pour que la rapport d'aspect soit réaliste.

Script

```
 % configuration de tourbillons
xpos=[1,1,-1,-1]; ypos=[1,-1,1,-1]; gamma=[1,-1,1,-1.01]; tmax=150;

% calcul des trajectoires
[xt,yt,t]=tourbitraj(xpos,ypos,gamma,tmax);

% les trajectoires
ccc='bbrr';
subplot(2,1,1);
for gre=1:4
    plot(xt(:,gre),yt(:,gre),ccc(gre)); hold on
end
axis equal; axis([-2,13,-2,7]);
xlabel('x'); ylabel('y'); title('trajectoires');

% animation du mouvement
subplot(2,1,2);
for ind=1:length(t)
    for gre=1:4
        plot(xt(ind,gre),yt(ind,gre),[ccc(gre) '.'],'markersize',10); hold on
    end
    hold off
    axis equal; axis([-2,13,-2,7]);
    xlabel('x'); ylabel('y'); title(t(ind));
    drawnow
end
```