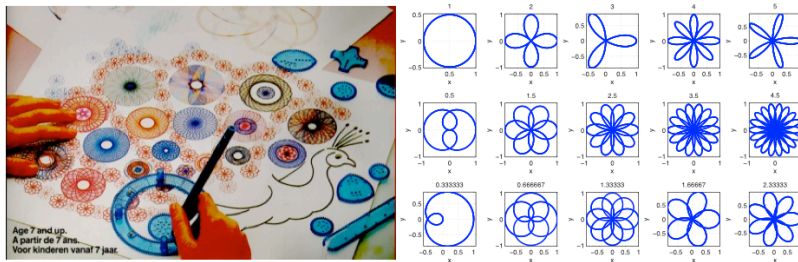


# Matlab: applications en mécanique

LA207, Université Pierre et Marie Curie.

## 2.3 TP2: Prise en main graphique

Dans ce TP, nous allons pratiquer les outils graphiques de Matlab pour tracer des listes de points et des formules mathématiques: sinus, cosinus et spirales. Il nous faudra également utiliser des boucles pour voir comment des courbes paramétrées changent de forme en fonction de leurs paramètres. Image: Rosaces tracées grâce à un spirographe à disque interne:



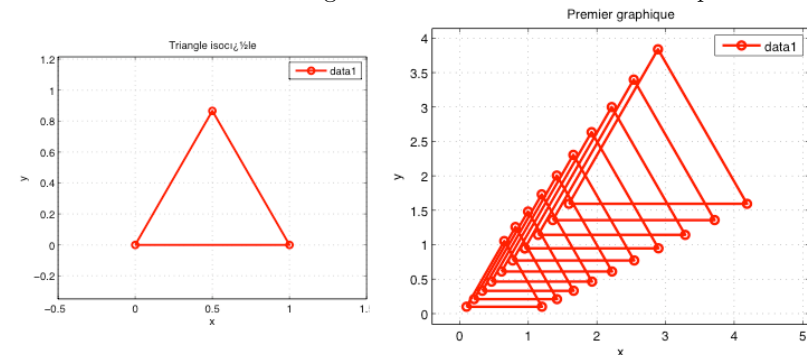
Compétences techniques:

- Tracer les points et segments d'une liste de points dont on connaît l'abscisse et l'ordonnée.
- Construire les points d'abscisse et d'ordonnée pour représenter une formule mathématique.
- De même pour une courbe paramétrée. Utilisation de sous-graphiques avec la fonction `subplot`. Utilisation des boucles `for` dans des scripts graphiques.

### 2.3.1 Manipulations

1. Tracer avec la fonction `plot` trois points d'abscisse  $x = (0, 1, 5)$ ,  $y = (1, -4, 10)$ . Les points sont tracés par des astérisques\*, et reliés par une ligne de couleur noire d'épaisseur 2. Donner un titre à votre graphique, et des noms aux axes avec `xlabel` `ylabel`, et donner la légende "data1", avec la fonction `legend`.

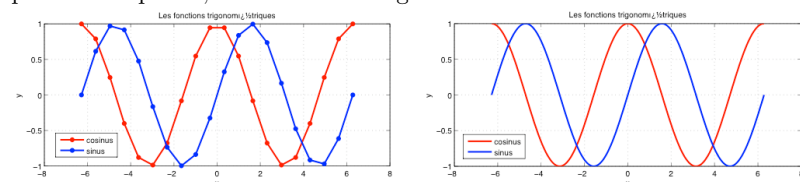
2. Tracez dans une fenêtre graphique avec la fonction `plot` un triangle isocèle de côté de longueur 1. Donner un titre, labels, légende à votre figure.
3. Avec une boucle `for`, tracer 20 triangles isocèles de tailles différentes et de centres différents. La figure suivante vous donne un exemple:



4. Nous allons maintenant tracer la représentation graphique d'une formule mathématique: les fonction trigonométriques sinus et cosinus. Pour cela il nous faut construire une liste de point d'abscisse et d'ordonnée pour les points de la courbe. Pour les abscisses on utilise la commande `x=linspace(-2*pi,pi,20)` avec une liste de 20 points équirépartis entre  $-2\pi$  et  $2\pi$ .

Dans une fenêtre graphique, tracez un sinus et un cosinus. Donner titre, labels, légende à votre graphique. Donnez des couleurs différentes et des marqueurs différents pour les deux courbes.

5. Dans la question précédente nous avons volontairement utilisé peu de points pour clairement voir que la figure est en fait composée d'une série de segments. Maintenant utilisez un grand nombre de points pour donner l'illusion d'une courbe continue. Pour cela, on n'utilise pas de marqueurs, seulement des segments.



6. Jusqu'à maintenant nous avons calculé la valeur de l'ordonnée  $y$  en fonction de l'abscisse  $x$ . Nous pouvons atteindre plus de richesse de courbes en utilisant des courbes paramétrées, c'est à dire pour lesquelles  $x$  et  $y$  sont obtenues à partir d'un paramètre qui varie, comme par exemple l'angle  $\theta$  pour un cercle  $x = r\cos(\theta)$ ,  $y = r\sin(\theta)$ .

Nous allons tracer des spirales. On peut définir des spirale avec  $x = r\cos(\theta)$ ,  $y = r\sin(\theta)$ , mais ici le rayon  $r$  dépend de l'angle  $\theta$ . Pour la spirale logarithmique, on a la loi  $r = a^\theta$ , et pour la spirale Archimédienne, on a  $r = a\theta$ , ou  $a$  est un paramètre fixe, par exemple 1.3.

Utilisez deux sous-graphiques (utilisez la fonction `subplot`, voir dans les notes de cours) pour scinder votre fenêtre graphique en deux. Tracez dans le sous-graphique de gauche une spirale logarithmique et dans le sous-graphique de droite une spirale Archimédienne comme pour l'exemple:



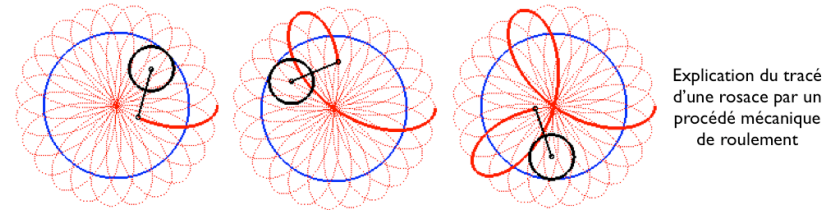
Donner à votre graphique titres, labels, légendes.

### 2.3.2 Pour aller plus loin

Les hypotrochoïdes sont les courbes décrites par un point lié à un disque roulant sans glisser intérieurement à un cercle de base. Ce sont donc les courbes que l'on obtient avec un spirographe avec disque interne. (voir le site [www.mathcurve.com](http://www.mathcurve.com), et la première image de ce TP). Ce qui nous intéresse ici, c'est que bien que la formule mathématique soit toujours la même, la forme de ces courbes peut changer beaucoup lorsque l'on fait varier les paramètres qui la définissent. Nous allons nous intéresser aux rosaces, qui sont un cas particulier d'hypotrochoïde.

La représentation cartésienne des rosaces est  $x = r\cos(\theta)$ ,  $y = r\sin(\theta)$ , avec la définition du rapport entre le rayon et l'angle:  $r = \cos(n\theta)$ ; ou  $n$  est le seul paramètre. Saurez-vous reproduire la figure suivante, qui représente

une série de rosaces pour  $n = 1, 2, 3, 4, 5$ ,  $n = 1/2, 3/2, 5/2, 7/2, 9/2$ ,  $n = 1/3, 2/3, 4/3, 5/3, 7/3$ ?



Explication du tracé d'une rosace par un procédé mécanique de roulement

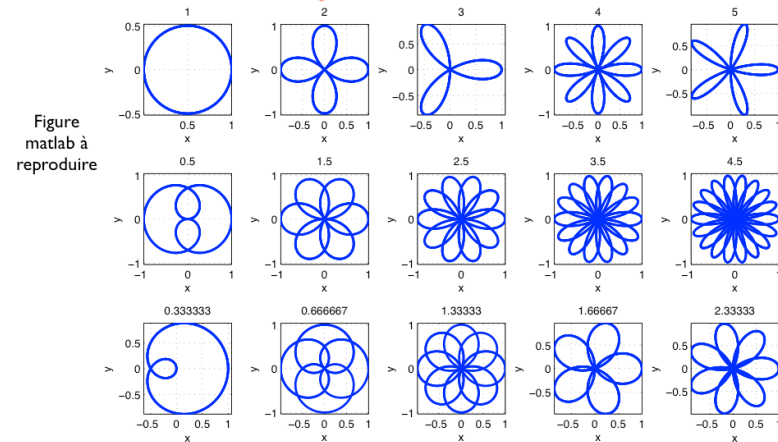
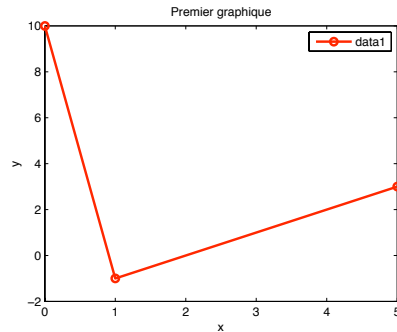


Figure matlab à reproduire

# Compte-rendu modèle pour le TP2 (2011)

## 2.3 TP2: Prise en main graphique

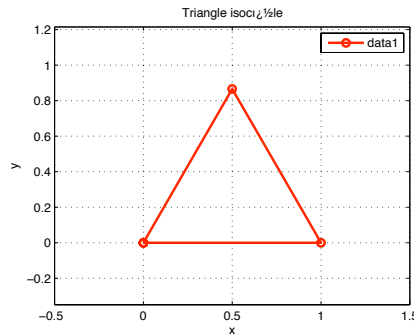
Dans ce TP, nous allons pratiquer les outils graphiques de Matlab pour tracer des listes de points et des formules mathématiques: sinus, cosinus et spirales. Il nous faudra également utiliser des boucles pour voir comment des courbes paramétrées changent de forme en fonction de leurs paramètres. Image: Rosaces tracées grâce à un spirotape à disque interne:



```
% definition des abscisses et des ordonnees
x=[0,1,5];
y=[10,-1,3];

% on trace
plot(x,y,'r-o','linewidth',2);
xlabel('x'); ylabel('y'); title('Premier graphique');
legend('data1')
```

Script

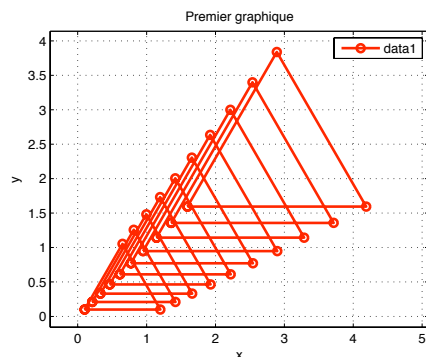


```
% definition des abscisses et des ordonnées des points
x=[0 1 0.5 0];
y=[0 0 sin(pi/3) 0];

% on trace
plot(x,y,'r-o','linewidth',2);
xlim([-0.5,1.5]); ylim([-0.5,1]);
xlabel('x'); ylabel('y'); title('Triangle isocèle'); legend('data1')
axis equal
grid on
```

Script

Il faut bien penser à remettre le premier point à la fin du tableau de sorte à "boucler la boucle"



```
% plein de triangles isocèles
hold off

% definition des points du triangle
x=[0 1 0.5 0];
y=[0 0 sin(pi/3) 0];

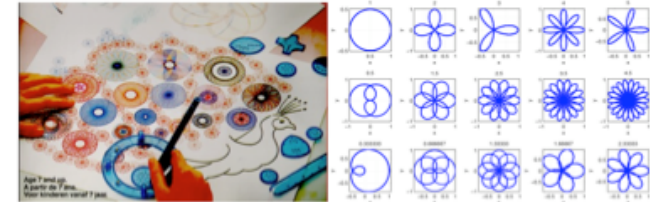
% on fait une boucle
for ind=1:10

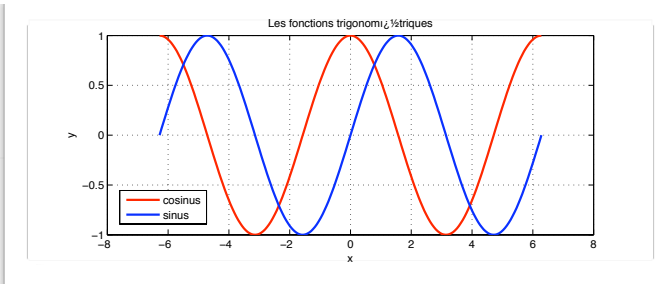
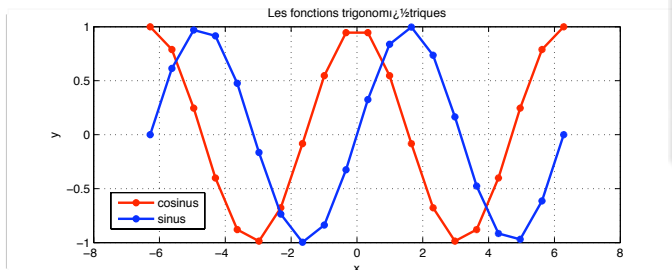
    % on translate le triangle, et on l'agrandit
    x=1.1*x+0.1;
    y=1.1*y+0.1;

    % on trace le triangle
    plot(x,y,'r-o','linewidth',2);
    xlabel('x'); ylabel('y'); title('Premier graphique'); legend('data1')
    axis equal
    hold on
end
grid on
```

Script

A chaque itération, on déplace le triangle en ajoutant à x et à y une valeur constante, et on l'agrandit en multipliant les tableaux des x et des y par une constante.





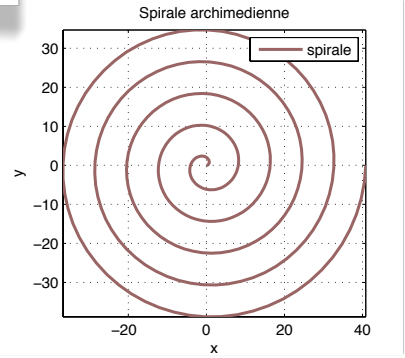
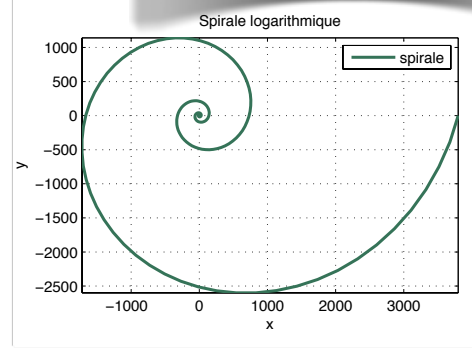
```

% definition des abscisses et des ordonnees
x=linspace(-2*pi,2*pi,20);
y1=cos(x);
y2=sin(x);

% on trace
plot(x,y1,'r+-',x,y2,'b*-');
legend('cosinus','sinus');
xlabel('x'); ylabel('y'); title('Les fonctions trigonométriques');
grid on

```

Script



```

##### la spirale logarithmique
subplot(1,2,1)
th=linspace(0,2*pi*5,300);

% les abscisses et ordonnees
r=1.3.^th;
x1=r.*cos(th);
y1=r.*sin(th);

% on trace
plot(x1,y1,'k-');
axis equal tight
legend('spirale');
xlabel('x'); ylabel('y'); title('Spirale logarithmique'); grid on

##### la spirale Archimédienne
subplot(1,2,2)
th=linspace(0,2*pi*5,300);

% les abscisses et ordonnees
r=1.3*th;
x2=r.*cos(th);
y2=r.*sin(th);

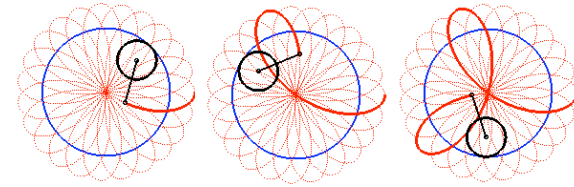
% on trace
plot(x2,y2,'k-');
axis equal tight
legend('spirale');
xlabel('x'); ylabel('y'); title('Spirale archimédienne'); grid on

```

Script

Ici il faut utiliser le symbole `*` pour la multiplication de tableau, puisque `r` est un tableau et `cos(th)` aussi: on multiplie chaque case du tableau `r` par la case correspondante du tableau `cos(th)`

## Pour aller plus loin



Explication du tracé d'une rosace par un procédé mécanique de roulement

Ici, on fait une boucle en ayant tout d'abord construit un vecteur des valeurs à affecter à  $n$ . On change de subplot à chaque itération, en utilisant l'indice "ind" de notre boucle.

### Script

```
% pour aller plus loin
figure(6)

% definition des valeurs a donner a n
nvec=[1,2,3,4,5,1/2,3/2,5/2,7/2,9/2,1/3,2/3,4/3,5/3,7/3];

% boucle sur le parametre n
for ind=1:length(nvec)
    subplot(3,5,ind)

    % on affecte a n sa valeur pour cette iteration
    n=nvec(ind);

    % on construit les abscisses et ordonnees
    th=linspace(0,10*pi,1000);
    r=cos(n*th);
    x=r.*cos(th);
    y=r.*sin(th);

    % on trace les figures
    plot(x,y,'b-', 'linewidth',2);
    axis tight equal
    xlabel('x'); ylabel('y'); title(n); grid on
end
```

