Cycloïde et spirographe

© Pierre Lantagne
Enseignant retraité du Collège de Maisonneuve

L'auteur de la programmation Maple est Mathieu Guay-Paquet. Mathieu a été l'un de mes étudiants de la filière Maple en 2004, dans les cours EED, NYB et NYC. Cette filière était optionnelle et a permis d'offrir aux élèves un enrichissement avec Maple. En 2004, Mathieu m'a accompagné à la soirée portes ouvertes au collège et c'est à l'occasion de cet événement qu'il a préparé ce document afin d'attirer les futures élèves du programme Sciences de la nature à choisir la filière Maple.

Bonne lecture à tous !

* Ce document Maple est exécutable avec la version 2019.1

Cycloïde et spirographe
Mathieu Guay-Paquet

Initialisation

```
> restart;
> with(plots,display,setoptions):
  setoptions(size=[200,200]);     # pour impression papier
  #setoptions( size=[300,300]);    # pour feuille à l'écran
> with(plottools,circle,line,point,rectangle):
> o:=-10^(-10):
```

Cycloïde

Les commandes suivantes définissent quelques objets géométriques utiles pour illustrer les différentes cycloïdes.

```
> Cercle:= t->circle([t,r1], r1):
> Segment:=t->line([t,r1], [t-r2*sin(t/r1),r1-r2*cos(t/r1)]):
> Point:=  t->point([t-r2*sin(t/r1),r1-r2*cos(t/r1)], color=red):
> Courbe:= t->plot([x-r2*sin(x/r1),r1-r2*cos(x/r1),x=o..t], color=navy):
> Complet:=t->display([Cercle,Segment,Point,Courbe](t), scaling=constrained):
```

Cycloïde

La cycloïde est la courbe tracée par un point fixe sur la circonférence d'un cercle qui roule sur une droite. La cycloïde formée par un cercle de rayon r roulant sur l'axe des x peut être décrite par une paire d'équations paramétriques:

$$x = t - r\sin\left(\frac{t}{r}\right)$$

$$y = r - r\cos\left(\frac{t}{r}\right)$$

$$t \in \mathrm{R}$$

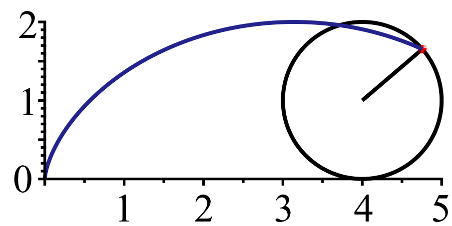
Comme une image vaut mille mots, les commandes suivantes permettent de visualiser le tracé de cette courbe pour un cercle de rayon 1.

```
> r1:=1;
  r2:=1;
```

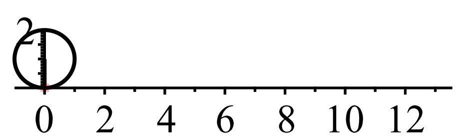
$$r1 := 1$$

$$r2 := 1 \qquad \textbf{(2.1.1)}$$

```
> Complet(4);
```




```
> display([seq(Complet(t*Pi/6),t=0..24)], scaling=constrained, insequence=
  true);
```



Cycloïde raccourcie

La cycloïde raccourcie ressemble à la cycloïde, mais au lieu de suivre le trajet d'un point sur la circonférence du cercle qui roule, elle suit le trajet d'un point à l'intérieur du cercle. La cycloïde raccourcie engendrée par un point à une distande d du centre d'un cercle de rayon r qui roule sur l'axe des x est donnée par la paire d'équations paramétriques suivante:

$$x = t - d\sin\left(\frac{t}{r}\right)$$

$$y = r - d\cos\left(\frac{t}{r}\right)$$

$$t \in \mathbb{R}$$

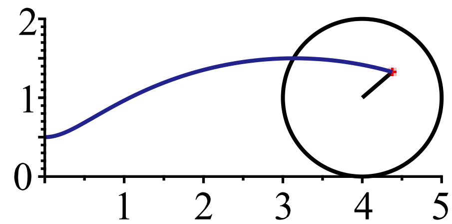
Les commandes suivantes illustrent la cycloïde raccourcie engendrée par un point à une distance $\frac{1}{2}$ du centre d'un cercle de rayon 1.

```
> r1:=1;
  r2:=0.5;
```

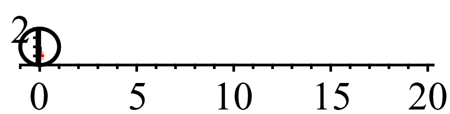
$$r1 := 1$$

$$r2 := 0.5 \tag{2.2.1}$$

```
> Complet(4);
```




```
> display([seq(Complet(t*Pi/6),t=0..36)], scaling=constrained,view=[-1..6*
  Pi+1.5,0..2], insequence=true);
```



Cycloïde allongée

La cycloïde allongée est comme la cycloïde raccourcie, sauf qu'elle suit le trajet d'un point situé à l'extérieur du cercle qui roule. Elle est décrite par les mêmes équations que la cycloïde raccourcie.
Les commandes suivantes illustrent la cycloïde allongée engendrée par un point à une distance 2 du centre d'un cercle de rayon 1.

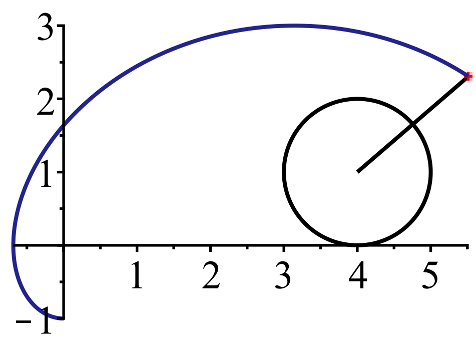
```
> r1:=1;
```

```
r2:=2;
```

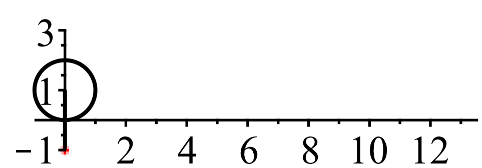
$$r1 := 1$$

$$r2 := 2 \tag{2.3.1}$$

```
> Complet(4);
```




```
> display([seq(Complet(t*Pi/6),t=0..24)], scaling=constrained, insequence=
  true);
```



Hypocycloïde, épicycloïde et spirographe

Les commandes suivantes définissent quelques objets géométriques utiles pour illustrer certaines variations de la cycloïde.

```
> u:=theta->[cos(theta),sin(theta)]:
> Cercle1:=t->circle([0,0], r1):
> Cercle2:=t->circle( (r1-r2)*u(t/r1) , r2):
> Segment:=t->line( (r1-r2)*u(t/r1) , (r1-r2)*u(t/r1)+r3*u(t/r1-t/r2) ):
> Point:=t->point( (r1-r2)*u(t/r1)+r3*u(t/r1-t/r2) ,color=red):
> Courbe:=t->plot([ ((r1-r2)*u(x/r1)+r3*u(x/r1-x/r2))[] ,x=o..t], color=navy):
> Complet:=t->display([Cercle1,Cercle2,Segment,Point,Courbe](t), scaling=
```

```
constrained):
```

Hypocycloïde

L'hypocycloïde est une courbe apparentée à la cycloïde, comme son nom l'indique. La différence entre ces deux courbes est que le cercle roule sur une droite dans le cas de la cycloïde, tandis qu'il roule à l'intérieur d'un autre cercle dans le cas de l'hypocycloïde.
Les commandes suivantes illustrent une hypocycloïde générée par un cercle de rayon 1 roulant dans un cercle de rayon 5.

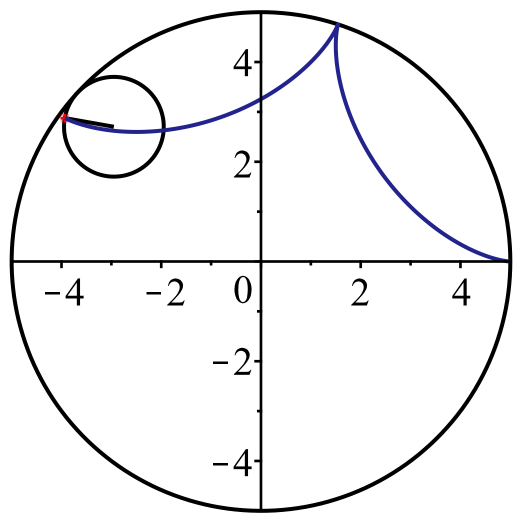
```
> r1:=5;
  r2:=1;
  r3:=1;
```

$$r1 := 5$$

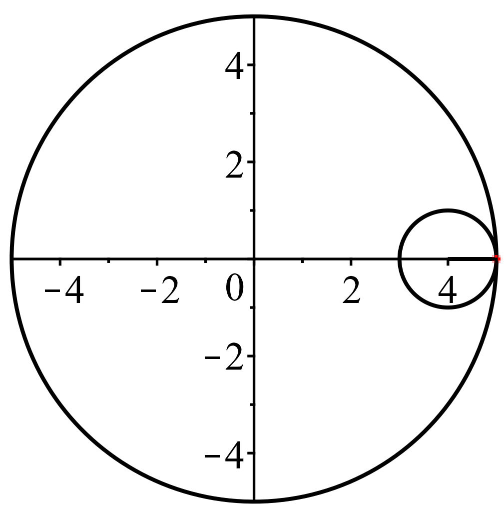
$$r2 := 1$$

$$r3 := 1 \tag{3.1.1}$$

```
> Complet(12);
```




```
> display([seq(Complet(t*Pi/6),t=0..60)], scaling=constrained, insequence=
  true);
```



Épicycloïde

L'épicycloïde est une autre variation de la cycloïde. Dans le cas de l'épicycloïde, le cercle roule à l'extérieur d'un autre cercle.
Les commandes suivantes illustrent une épiocycloïde générée par un cercle de rayon 1 roulant dans un cercle de rayon 5.

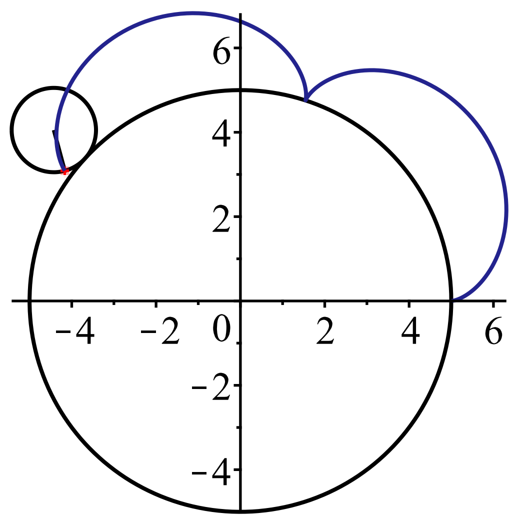
```
> r1:=5;
  r2:=-1;
  r3:=-1;
```

$$r1 := 5$$

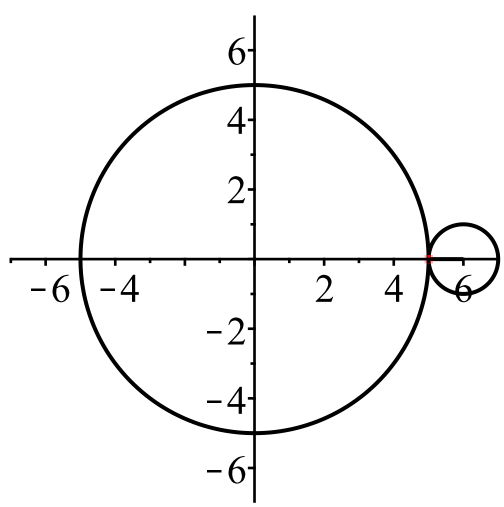
$$r2 := -1$$

$$r3 := -1 \tag{3.2.1}$$

```
> Complet(12);
```




```
> display([seq(Complet(t*Pi/6),t=0..60)], scaling=constrained, insequence=
  true);
```



En y pensant un peu, on se rend compte que les belles images tracées au spirographe sont en fait composées d'hypocycloïdes et d'épicycloïdes raccourcies. La fonction Spiro, définie ici, permet de tracer des spirographes avec Maple. Elle prend la forme suivante:

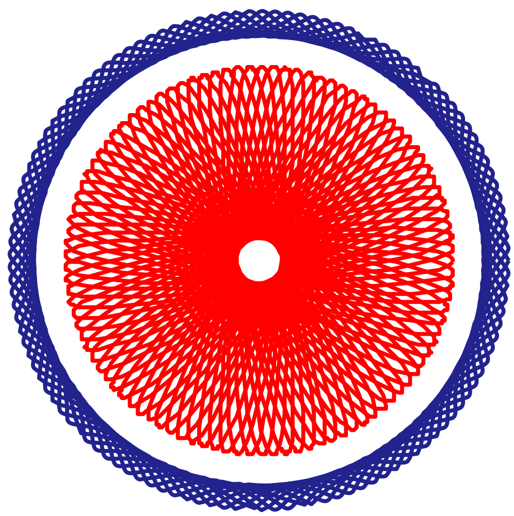
$$Spiro(r1, r2, r3, t1, t2, c) \text{, où:}$$

r1 est le rayon du cercle fixe
r2 est le rayon du cercle mobile
r3 est la distance entre le centre du cercle mobile et le point traceur
t1 est la distance parcourue par le cercle mobile sur la circonférence du cercle fixe avant de commencer le tracé
t2 est la distance parcourue par le cercle mobile sur la circonférence du cercle fixe à la fin du tracé
c est la couleur du tracé (par exemple, red, green, yellow, blue, navy, etc.)

```
> unassign('r1','r2','r3');
> Spiro:=(r1,r2,r3,t1,t2,c)->plot([ ((r1-r2)*u(x/r1)+r3*u(x/r1-x/r2))[] ,x=t1..t2], color=c, scaling=constrained, axes=none):
```

Spirographe 1

```
> Spiro1:=Spiro(121,10,6,0, 2*121*10*Pi ,navy):
> Spiro2:=Spiro(101,50,40,0, 2*101*50*Pi ,red):
> display(Spiro1,Spiro2);
```



Spirographe 2

```
> Spiro3:=Spiro(101,50,40, 0,3367*Pi ,red):
> Spiro4:=Spiro(101,50,40, 3367*Pi,6733*Pi ,blue):
> Spiro5:=Spiro(101,50,40, 6733*Pi,10100*Pi ,yellow):
> Fond:=rectangle([-101,-101],[101,101],color=black):
> display(Fond,Spiro3,Spiro4,Spiro5);
```