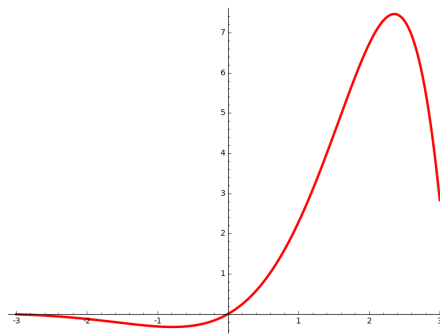


6. Courbes et surfaces

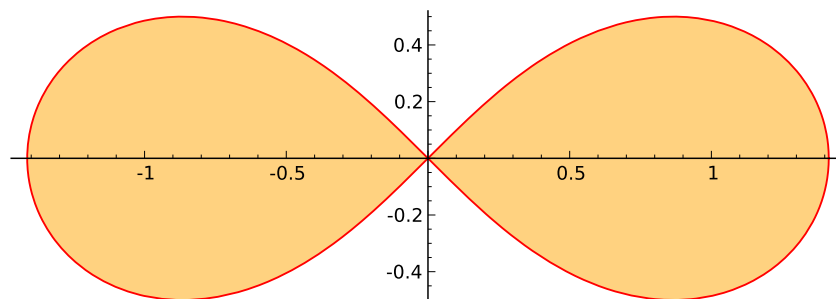
La visualisation est une étape importante dans l'élaboration des preuves en mathématiques. L'avènement des logiciels de calcul formel possédant une interface graphique évoluée a rendu cette phase attrayante. Nous allons explorer les possibilités graphiques offertes par Sage. Nous avons déjà vu comment tracer les graphes de fonctions avec la commande `plot` (voir « Premiers pas avec Sage »), par exemple `plot(sin(x)*exp(x), (x, -3, 3))` trace le graphe de la fonction $f(x) = \sin(x) \exp(x)$ sur l'intervalle $[-3, 3]$.



En plus des graphes de fonctions, Sage sait tracer des courbes et des surfaces par d'autres méthodes.

6.1. Courbes paramétrées

La commande `parametric_plot((f(t), g(t)), (t, a, b))` permet de tracer la courbe paramétrée plane donnée par les points de coordonnées $(f(t), g(t))$ lorsque le paramètre t varie dans l'intervalle $[a, b]$. Commençons par la lemniscate de Bernoulli :



Code 1 (*lemniscate-bernoulli.sage*).

```
var('t')
x = sqrt(2)*cos(t)/(1+sin(t)^2)
y = sqrt(2)*cos(t)*sin(t)/(1+sin(t)^2)
G = parametric_plot((x, y), (t, 0, 2*pi))
```

G.show()

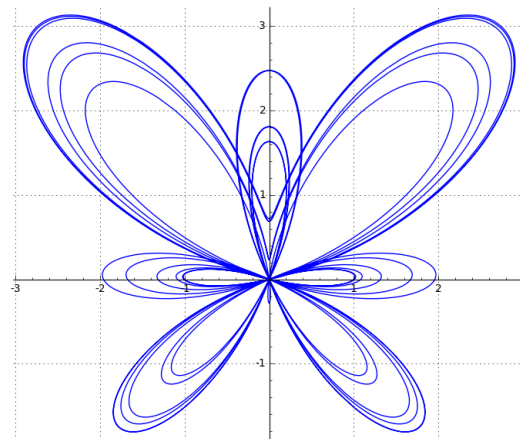
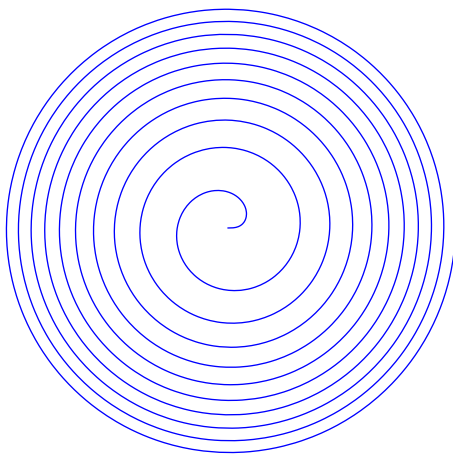
Travaux pratiques 1.

1. Tracer la spirale de Fermat d'équation

$$\begin{cases} x(t) = \sqrt{t} \cos t \\ y(t) = \sqrt{t} \sin t \end{cases} \quad t \in \mathbb{R}_+.$$

2. Tracer la courbe du papillon

$$\begin{cases} x(t) = \sin(t) \left(\exp(\cos(t)) - 2 \cos(4t) - \sin^5\left(\frac{t}{12}\right) \right) \\ y(t) = \cos(t) \left(\exp(\cos(t)) - 2 \cos(4t) - \sin^5\left(\frac{t}{12}\right) \right) \end{cases} \quad t \in \mathbb{R}.$$

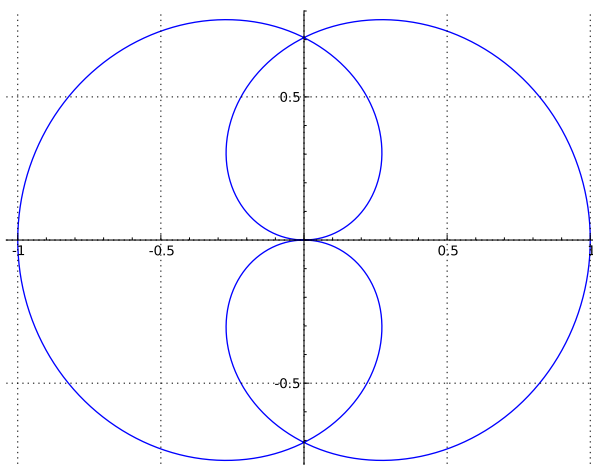


Les commandes de tracé possèdent de nombreuses options, qu'on pourra découvrir grâce à la commande `help(parametric_plot)`. Par exemple :

- Imposer un repère orthonormé : `aspect_ratio = 1`
- Nombre de points pour le tracé : `plot_points = 500`
- Ne pas afficher les axes : `axes = False`
- Afficher une grille : `gridlines = True`
- Couleur du trait : `color = 'red'`,
- Remplir l'intérieur : `fill = True`,
- Couleur du remplissage : `fillcolor = 'orange'`.

6.2. Courbes en coordonnées polaires

Les courbes en coordonnées polaires sont tracées grâce à la commande `polar_plot(r(t), (t, a, b))`, qui produira l'ensemble des points de coordonnées polaires $[r(t) : t]$ pour t variant dans l'intervalle $[a, b]$. Voici le folium de Dürer d'équation $r(t) = \sin \frac{t}{2}$, $t \in \mathbb{R}$.



Code 2 (*durer-folium.sage*).

```
var('t')
G = polar_plot(sin(t/2), (t, 0, 4*pi))
G.show()
```

Travaux pratiques 2.

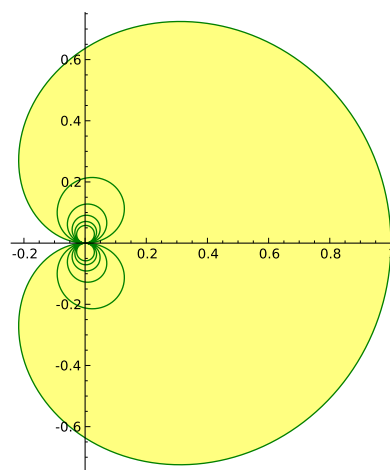
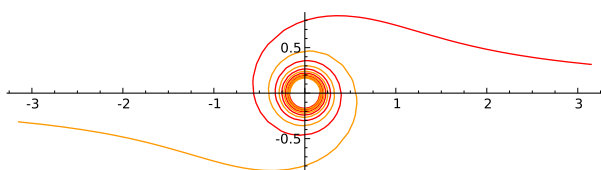
1. Tracer la courbe du Lituus d'équation polaire

$$r(t)^2 = \frac{1}{t} \quad t \in \mathbb{R}^*.$$

2. Tracer la cochléoïde d'équation polaire

$$r(t) = \frac{\sin t}{t} \quad t \in \mathbb{R}^*.$$

Indications : on peut définir deux graphes G1 et G2 pour chacun des intervalles de définition. On superpose les graphes avec $G = G1 + G2$, puis on les affiche avec $G.show()$.



6.3. Courbes définies par une équation

Une courbe peut avoir plusieurs types de représentations, comme par exemple un cercle d'équation paramétrique $(\cos t, \sin t)$ ou d'équation implicite $x^2 + y^2 = 1$. La commande

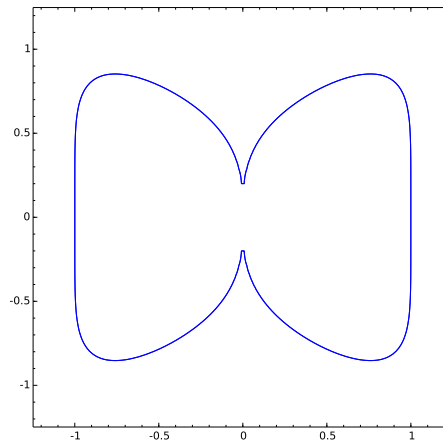
```
implicit_plot(f(x, y), (x, a, b), (y, c, d))
```

permet de tracer l'ensemble des couples (x, y) dans $[a, b] \times [c, d]$ qui sont solutions de l'équation $f(x, y) = 0$. Voici une autre courbe de papillon, cette fois algébrique.

Code 3 (*butterfly-curve-algebraic.sage*).

```
f(x,y) = x^6 + y^6 - x^2
G = implicit_plot(f, (x, -1.2, 1.2), (y, -1.2, 1.2))
G.show()
G.save('butterfly_curve_algebraic.png')
```

Noter comment il est possible de sauvegarder une image du tracé dans un fichier externe.



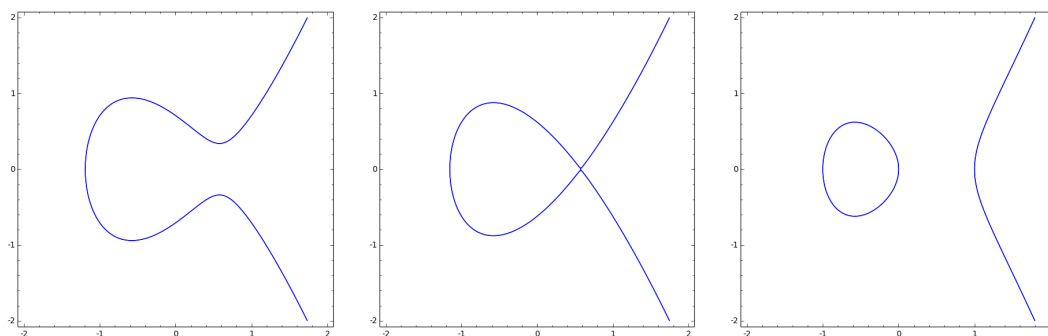
Travaux pratiques 3.

1. Construire une liste contenant les représentations graphiques de la courbe définie par

$$y^2 - x^3 + x + c = 0$$

lorsque le paramètre c parcourt l'intervalle $[-1, 1]$ par pas de 0.1.

2. À l'aide de la commande `animate`, réaliser une animation qui affiche l'évolution de la courbe pour $c \in [-1, 1]$.



6.4. Courbes de l'espace

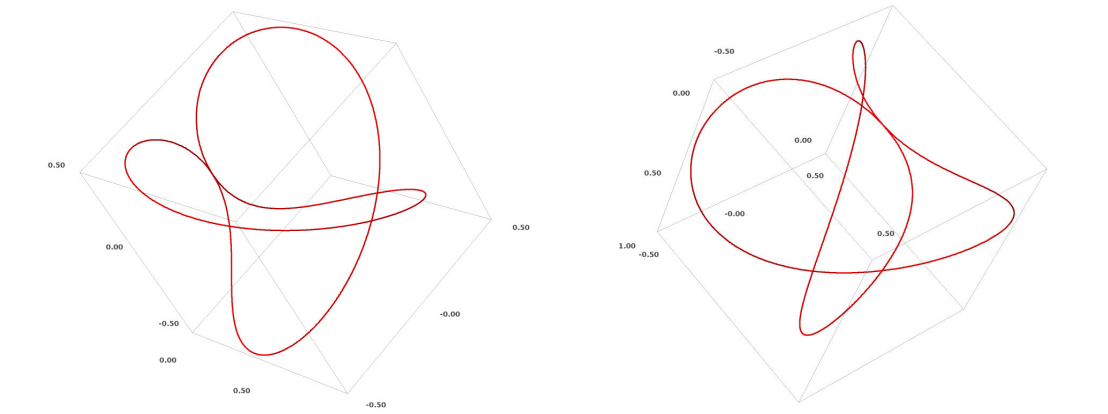
La commande `parametric_plot((x, y, z), (t, a, b))`, (ou bien `parametric_plot3d()`) analogue à celle de la dimension 2, trace la courbe paramétrée de l'espace donnée par les points de coordonnées $(x(t), y(t), z(t))$ lorsque le paramètre t varie dans l'intervalle $[a, b]$.

Travaux pratiques 4.

Tracer la courbe d'Archytas d'équation paramétrique :

$$\begin{cases} x(t) = \cos^2 t \\ y(t) = \cos t \sin t \\ z(t) = \pm \sqrt{\cos t (1 - \cos t)} \end{cases} \quad t \in \left[-\frac{\pi}{2}, +\frac{\pi}{2}\right].$$

Vous obtiendrez une figure qu'il est possible d'orienter dynamiquement avec la souris. Voici quelques-unes de ces vues.



6.5. Surfaces

Découvrez, à l'aide de l'énoncé suivant, les différentes méthodes pour tracer des surfaces avec Sage.

Travaux pratiques 5.

1. Tracer le graphe de la fonction $f(x, y) = x^2y^2 - (x^2 + y^2)^3$ définie sur $(x, y) \in [-1, 1] \times [-1, 1]$. Utiliser la fonction `plot3d`.

2. Tracer la surface d'Enneper définie par l'équation

$$\left(\frac{y^2 - x^2}{2z} + \frac{2}{9}z^2 + \frac{2}{3} \right)^3 - 6 \left(\frac{y^2 - x^2}{4z} - \frac{1}{4}(x^2 + y^2 + \frac{8}{9}z^2) + \frac{2}{9} \right)^2 = 0.$$

Utiliser la fonction `implicit_plot3d`.

3. Tracer la nappe paramétrée définie par :

$$\begin{cases} x(s, t) = t^2 \cos s \\ y(s, t) = s^2 \sin t \\ z(s, t) = \cos t + \sin t \end{cases} \quad s \in [0, 1], \quad t \in [-\pi, +\pi].$$

Utiliser la fonction `parametric_plot3d`.

4. Tracer la surface paramétrée définie en coordonnées cylindriques par :

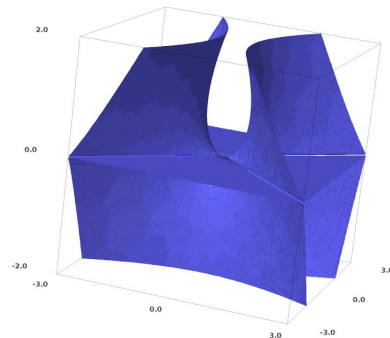
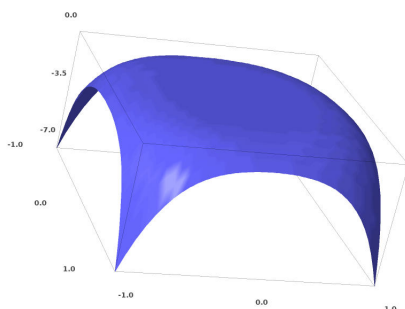
$$r(\theta, z) = z^3 \cos \theta \quad \theta \in [0, 2\pi], \quad z \in [0, 2].$$

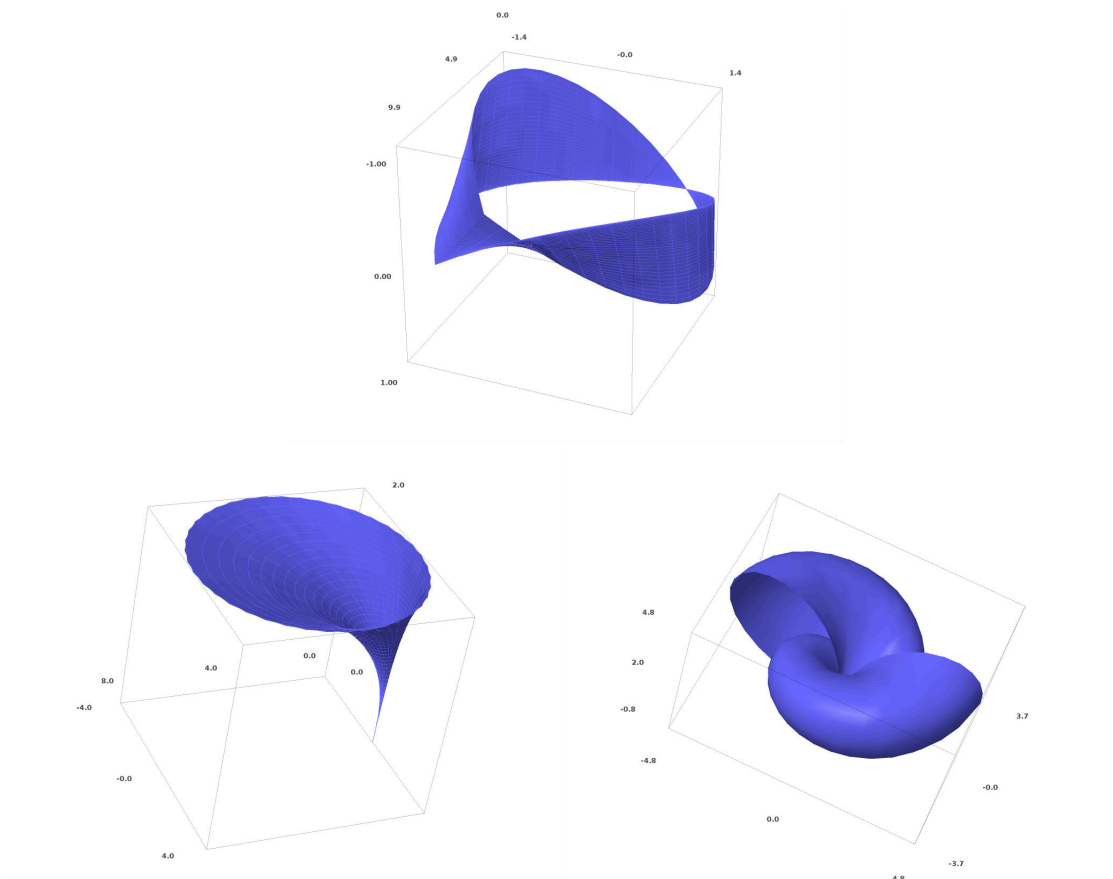
Utiliser la fonction `cylindrical_plot3d`.

5. Tracer la surface paramétrée définie en coordonnées sphériques par

$$r(\theta, \phi) = \theta \sin(2\phi) \quad \theta \in [-1, 2\pi], \quad \phi \in [0, \pi].$$

Utiliser la fonction `spherical_plot3d`.





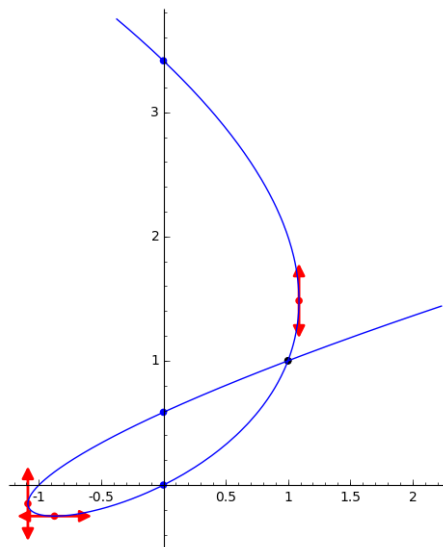
6.6. Étude d'une courbe paramétrée

Travaux pratiques 6.

Étudier en détail la courbe paramétrée du plan définie par

$$\begin{cases} x(t) = t^3 - 2t \\ y(t) = t^2 - t \end{cases} \quad t \in \mathbb{R}.$$

1. Tracer la courbe.
2. Trouver les points d'intersection de la courbe avec l'axe des ordonnées.
3. Trouver les points en lesquels la tangente est verticale, puis horizontale.
4. Trouver les coordonnées des points doubles.



1. La courbe a l'allure d'une boucle.
2. On définit $x = t^3 - 2t$ et $y = t^2 - t$. On obtient les valeurs de t correspondant aux points d'intersection de la courbe avec l'axe ($x = 0$) en résolvant l'équation $x(t) = 0$: `solve(x==0, t)`. On obtient trois solutions $-\sqrt{2}$, 0 et $+\sqrt{2}$ correspondant aux trois points $\{(0, 2 + \sqrt{2}), (0, 0), (0, 2 - \sqrt{2})\}$
3. On définit les fonctions dérivées $x'(t)$ et $y'(t)$ par `xx = diff(x, t)` et `yy = diff(y, t)`. Les valeurs de t pour lesquelles la tangente à la courbe est verticale s'obtiennent en résolvant l'équation $x'(t) = 0$, ce qui s'écrit `solve(xx==0, t)`.
4. Trouver les points doubles est souvent un calcul délicat (même pour un logiciel de calcul formel). Il s'agit ici de résoudre le système d'équations polynomiales :

$$\begin{cases} x(s) = x(t) \\ y(s) = y(t) \end{cases} \quad s, t \in \mathbb{R}.$$

Mais il faut exclure la solution évidente $t = s$. Algébriquement cela signifie que $(s - t)$ divise les polynômes de deux variables $x(s) - x(t)$ et $y(s) - y(t)$. Autrement dit, il s'agit de résoudre :

$$\begin{cases} \frac{x(s) - x(t)}{s - t} = 0 \\ \frac{y(s) - y(t)}{s - t} = 0 \end{cases} \quad s, t \in \mathbb{R}.$$

Le code suivant fournit la solution $(s, t) = \left(\frac{1-\sqrt{5}}{2}, \frac{1+\sqrt{5}}{2}\right)$ correspondant à un unique point double de coordonnées $(1, 1)$ (Sage donne deux solutions qui mathématiquement par symétrie n'en font qu'une).

Code 4 (*courbe.sage*).

```
var('s,t')
x = t^3-2*t
y = t^2-t
exprx = x.subs(t=s) - x                    # Expression x(s)-x(t)
exprx = (exprx/(s-t)).simplify_rational()   # (x(s)-x(t))/(s-t)
eqx = exprx == 0                           # Équation (x(s)-x(t))/(s-t) = 0
expy = y.subs(t=s) - y                    # Expression y(s)-y(t)
expy = (expy/(s-t)).simplify_rational()     # (y(s)-y(t))/(s-t)
eqy = expy == 0                           # Équation (y(s)-y(t))/(s-t) = 0
points_double = solve([eqx, eqy], s, t)    # Solutions
```

6.7. La projection stéréographique

Travaux pratiques 7.

Soit \mathcal{S} la sphère centrée à l'origine de \mathbb{R}^3 et de rayon 1. On note $N = (0, 0, 1)$ le pôle Nord. Soit \mathcal{P} le plan équatorial d'équation ($z = 0$). La **projection stéréographique** est l'application $\Phi : \mathcal{S} \setminus \{N\} \rightarrow \mathcal{P}$ qui à un point S de la sphère associe le point $P = (NS) \cap \mathcal{P}$ défini par l'intersection de la droite (NS) avec le plan \mathcal{P} .

1. En écrivant la relation $\overrightarrow{NP} = k\overrightarrow{NS}$ et sachant que $P \in \mathcal{P}$, trouver k et en déduire P .

Vérifier ainsi que l'application Φ est définie par :

$$\Phi : \mathcal{S} \setminus \{N\} \rightarrow \mathcal{P} \quad \Phi(x, y, z) = \left(\frac{x}{1-z}, \frac{y}{1-z} \right)$$

Définir la fonction correspondante avec Sage.

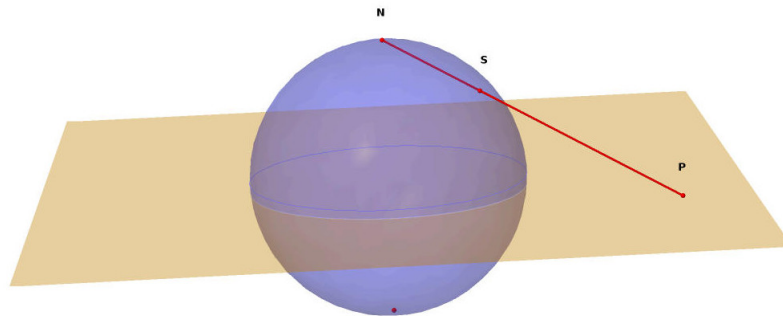
2. Définir et vérifier que l'application inverse est :

$$\Psi : \mathcal{P} \rightarrow \mathcal{S} \setminus \{N\} \quad \Psi(X, Y) = \left(\frac{2X}{\rho}, \frac{2Y}{\rho}, 1 - \frac{2}{\rho} \right) \quad \text{avec } \rho = 1 + X^2 + Y^2$$

3. Écrire une fonction qui dessine une courbe paramétrée $\mathcal{C}' : (x(t), y(t))$, $t \in [a, b]$ du plan \mathcal{P} , et qui dessine aussi l'image inverse de la courbe sur la sphère, $\mathcal{C} = \Psi(\mathcal{C}')$.
4. Vérifier graphiquement deux propriétés fondamentales de la projection stéréographique :
 - (a) « La projection stéréographique envoie les cercles de la sphère sur des cercles ou des droites du plan. »

- (b) « La projection stéréographique préserve les angles. » En particulier, deux courbes qui se coupent à angle droit, s'envoient sur deux courbes qui se coupent à angle droit.
5. Soit \mathcal{C}' la spirale logarithmique d'équation $(e^t \cos t, e^t \sin t)$, $t \in \mathbb{R}$. Tracer la loxodromie de la sphère qui est $\mathcal{C} = \Psi(\mathcal{C}')$.
6. Soit la courbe $\mathcal{C} \subset \mathcal{S}$ définie par $\frac{1}{13t^2-6t+2}(4t, -6t+2, 13t^2-6t)$. Montrer que son image $\mathcal{C}' = \Phi(\mathcal{C}) \subset \mathcal{P}$ est une droite, dont vous déterminerez une équation.

1.



Le vecteur \overrightarrow{NP} est colinéaire au vecteur \overrightarrow{NS} , il existe donc $k \in \mathbb{R}$ tel que $\overrightarrow{NP} = k\overrightarrow{NS}$. Mais on veut de plus que P appartienne au plan \mathcal{P} , c'est-à-dire $z_P = 0$. Ce qui permet de trouver k et d'en déduire $P = N + k\overrightarrow{NS}$. On laisse Sage faire les calculs :

Code 5 (*stereographic.sage (1)*).

```
var('x,y,z')          # Variables de l'espace
var('X,Y')             # Variables du plan
var('k')
N = vector([0, 0, 1])  # Les points N, S, P comme des vecteurs
S = vector([x, y, z])
P = vector([X, Y, 0])
V = (P-N) - k*(S-N)    # Le vecteur NP - k NS
eq = V[2] == 0         # On veut la troisième coordonnée nulle
sol = solve(eq, k)     # Equation en k pour cette condition
k = sol[0].rhs()       # Solution k
P = N + k*(S - N)      # Le point P
```

On obtient $k = \frac{1}{1-z}$. Ainsi, si $S = (x, y, z)$ alors $P = (\frac{x}{1-z}, \frac{y}{1-z}, 0)$. En identifiant les points du plan à \mathbb{R}^2 , on a ainsi prouvé $\Phi(x, y, z) = (\frac{x}{1-z}, \frac{y}{1-z})$. D'où la définition de la fonction Φ :

Code 6 (*stereographic.sage (2)*).

```
def stereo(x, y, z):
    X = x/(1 - z)
    Y = y/(1 - z)
    return X, Y
```

2. Voici la fonction Ψ :

Code 7 (*stereographic.sage (3)*).

```
def inverse_stereo(X, Y):
    r = 1 + X^2 + Y^2
    x = 2*X / r
    y = 2*Y / r
    z = 1 - 2/r
```



```
return x, y, z
```

On profite du fait que l'on nous donne le candidat, pour se contenter de vérifier que c'est bien la bijection réciproque, c'est-à-dire $\Phi(\Psi(X, Y)) = (X, Y)$ pour tout $(X, Y) \in \mathbb{R}^2$. La composition s'écrit

Code 8 (*stereographic.sage (4)*).

```
newx, newy, newz = inverse_stereo(X, Y)
newX, newY = stereo(newx, newy, newz)
```

et comme alors newX vaut X et que newY vaut Y, cela prouve le résultat.

Il est possible de composer les fonctions de plusieurs variables, mais il faut rajouter une « * » devant la fonction à composer :

```
stereo(*inverse_stereo(X, Y))
```

cela renvoie X, Y ce qui prouve bien $\Phi(\Psi(X, Y)) = (X, Y)$. L'opérateur « * » devant une liste permet de la décompacter, par exemple $*(2, 5, 7)$ s'interprète comme 2, 5, 7 qui peut alors être passé en argument à une fonction.

Pour prouver $\Psi(\Phi(x, y, z)) = (x, y, z)$, il faudrait se souvenir que $(x, y, z) \in \mathcal{S}$ et donc que $x^2 + y^2 + z^2 = 1$.

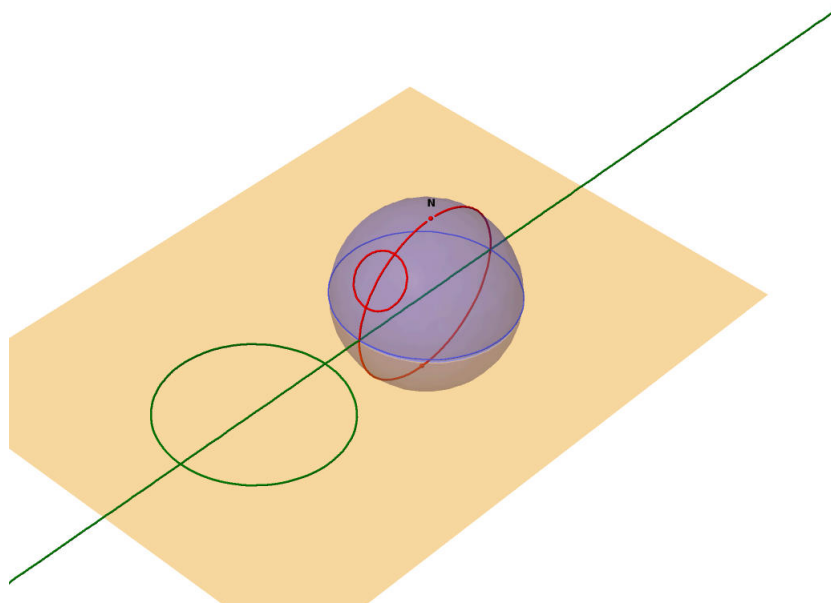
3. et 4. Voici comment tracer les courbes. La courbe du plan est tracée comme une ligne polygonale, avec un pas assez petit. Ensuite on calcule l'image par Ψ de chacun de ces points.

Code 9 (*stereographic.sage (5)*).

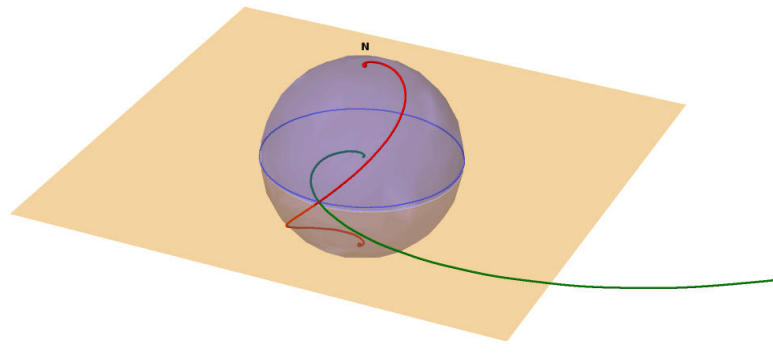
```
def courbes(X, Y, a, b):
    XYtab = [ [X.subs(t=myt), Y.subs(t=myt), 0] for myt in srange(a, b, 0.1) ]
    xyztab = [ inverse_stereo(coord[0], coord[1]) for coord in XYtab ]
    G = sphere((0, 0, 0), 1)
    G = G + line(XYtab)
    G = G + line(xyztab)
    return G
```

Par exemple $G = \text{courbes}(t^3, t^2, -2, 2)$ puis $G.\text{show}()$ trace la courbe du plan d'équation paramétrique (t^3, t^2) , $t \in [-2, 2]$, ainsi que son image par la projection stéréographique inverse.

Voici le cas d'un cercle et d'une droite du plan qui se coupent à angle droit. On visualise qu'il en est de même pour les cercles de la sphère.



5. Voici une loxodromie de la sphère, courbe utilisée par les navigateurs, car elle correspond à une navigation à cap constant.



6. Pour x, y, z définis par la paramétrisation, on calcule l'image par la projection stéréographique à l'aide de $X, Y = \text{stereo}(x, y, z)$, on n'oublie pas de simplifier à l'aide de `full_simplify()`. Cela donne une paramétrisation de l'image, $x(t) = 2t$, $y(t) = -3t + 1$, qui est bien l'équation d'une droite.