

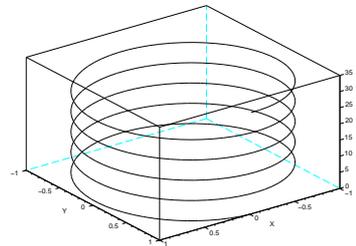
Graphiques 3D avec Scilab

1 Courbes 3D

Le tracé de courbes en 3D se fait à l'aide de la commande **param3d** qui possède une syntaxe semblable à **plot**.

Exemple pour tracer une ellipse :

```
--> t = 0:%pi/50:10*%pi;  
--> param3d(sin(t),cos(t),t);  
  
// Pour changer la vue  
--> aa=gca();  
--> aa.rotation_angles=[4.25,53.5];
```



2 Surfaces

Pour la représentation de la surface $z = f(x, y)$, on a besoin de connaître les triplets de coordonnées $(X_i, Y_i, Z_{i,j} = f(X_i, Y_j))$ pour un certain nombre de points (X_i, Y_j) $i = 1 \cdots N, j = 1 \cdots M$. On voit que Z a la structure d'une matrice. Matlab demande également à ce que X et Y soient des matrices. On construit ces dernières avec l'instruction **meshgrid**.

Si $x = (x_0, x_1, \dots, x_n)$ et $y = (y_0, y_1, \dots, y_m)$, l'instruction $[X, Y] = \text{meshgrid}(x, y)$; produit les matrices X et Y à m lignes, n colonnes suivantes :

$$X = \begin{pmatrix} x_0 & x_1 & \cdots & x_n \\ x_0 & x_1 & \cdots & x_n \\ \cdots & \vdots & \vdots & \vdots \\ x_0 & x_1 & \cdots & x_n \end{pmatrix}, \quad Y = \begin{pmatrix} y_0 & y_0 & \cdots & y_0 \\ y_1 & y_1 & \cdots & y_1 \\ \cdots & \vdots & \vdots & \vdots \\ y_m & y_m & \cdots & y_m \end{pmatrix}.$$

Grillage en perspective : mesh

```
// on commence par construire deux vecteurs qui d\efinissent le maillage  
// sur chacun des axes  
--> x=[-3:0.3:3];  
--> y=[-3:0.3:3];  
// meshgrid fournit les matrices X et Y a partir de x et y  
--> [X,Y] = meshgrid(x,y);  
--> Z = cos(X).*sin(Y);  
--> mesh(X,Y,Z) % trac\ 'e de la surface
```

L'instruction **mesh** donne un aspect "grillagé" à la surface. L'instruction **meshc** trace les lignes de niveau de la surface.

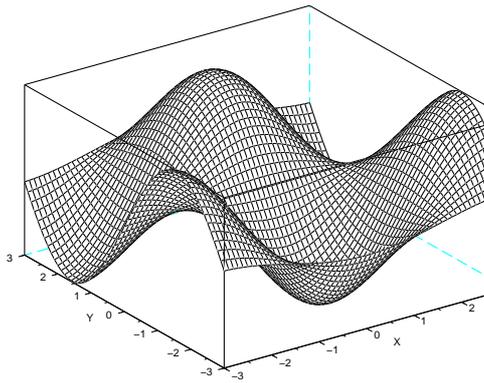


FIGURE 1 – mesh.

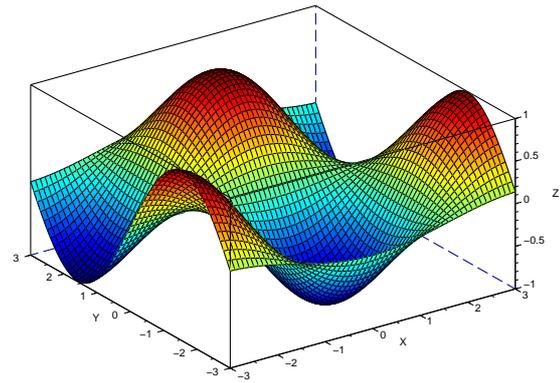


FIGURE 2 – surf.

Surfaces colorées : surf

```
// On commence par construire deux vecteurs qui d'efinissent le maillage
// sur chacun des axes
--> x=[-3:0.1:3];
--> y=[-3:0.1:3];
--> [X,Y]=meshgrid(x,y); //meshgrid fournit les matrices X et Y a partir de x et y
--> Z = cos(X).*sin(Y);
-->
--> f=gcf();
--> f.color_map = jetcolormap(128); //les couleurs s'etendent du bleu au rouge
-->
--> surf(X,Y,Z)           // trac'e de la surface
-->
-->
--> ee=gce()             // pour un courbe 'interpolee'
--> ee.color_flag=3
--> ee.thickness=0
-->
--> f.children(1).view="2d" // Vision 2d
```

La différence entre **surf** et **mesh** est que l'instruction **surf** colore chaque élément du grillage.

Courbes de niveau : contour

L'instruction suivante construit 20 lignes de niveau :

```
--> contour(x,y,Z,20);
```

