Representing Curves and Surfaces

2D and 3D curves and 2D surfaces are useful in many computer applications:

- Computer-Aided Design (CAD): eg, designing the body of a car.

- Illustration: eg, a book figure done as a line drawing.

- Modeling reality: eg, the path along which something moves, or the shape of an object.

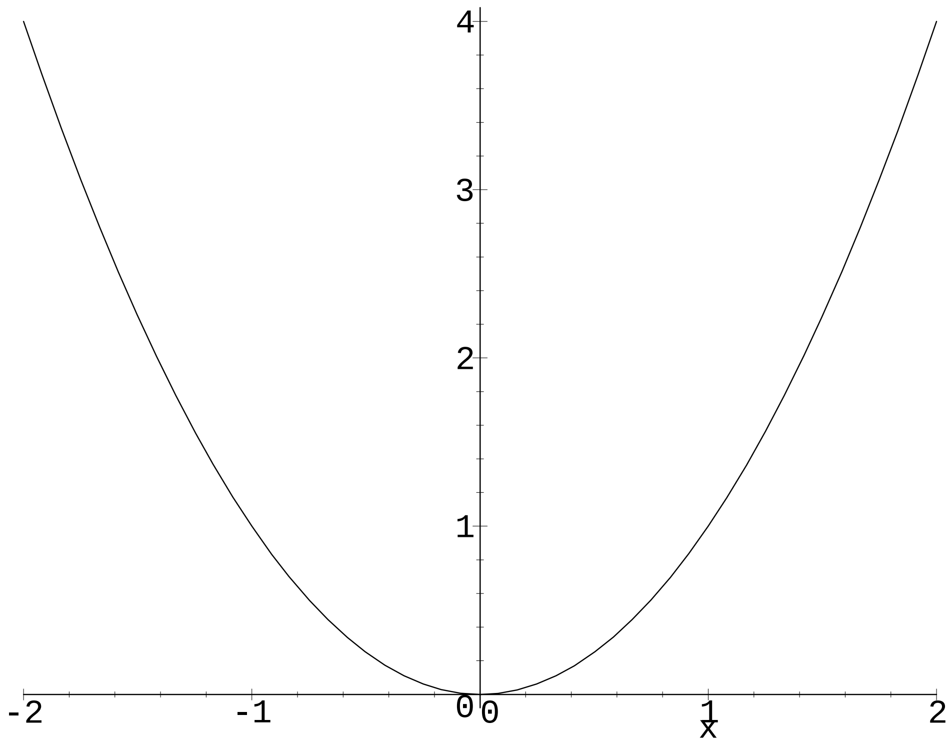
How can we represent curves and surfaces conveniently?

Explicit Representations

Some curves and surfaces can be represented by an explicit function.

For example: A parabola, given by $y = x^2$

```
> plot(x^2,x=-2..2);
```



But not all curves (or sufaces) can be seen as functions. Some curve around so that there is more than one y value for each x value.

Implicit Representations

Any curve or surface can be represented by an equation that is satisfied for points on the curve or surface.

For example:

$$x^2 + y^2 = r^2$$

defines a circle of radius r with centre at the origin.

We can get the surface of a sphere of radius r with:

$$x^2 + y^2 + z^2 = r^2$$

These representations may be hard to work with, however. Equations are often not easy to solve!

Parametric Representations of Curves

We can represent a curve (2D or 3D) by imagining a point travelling along the curve, tracing it out. We specify the coordinates of this point as function of time.

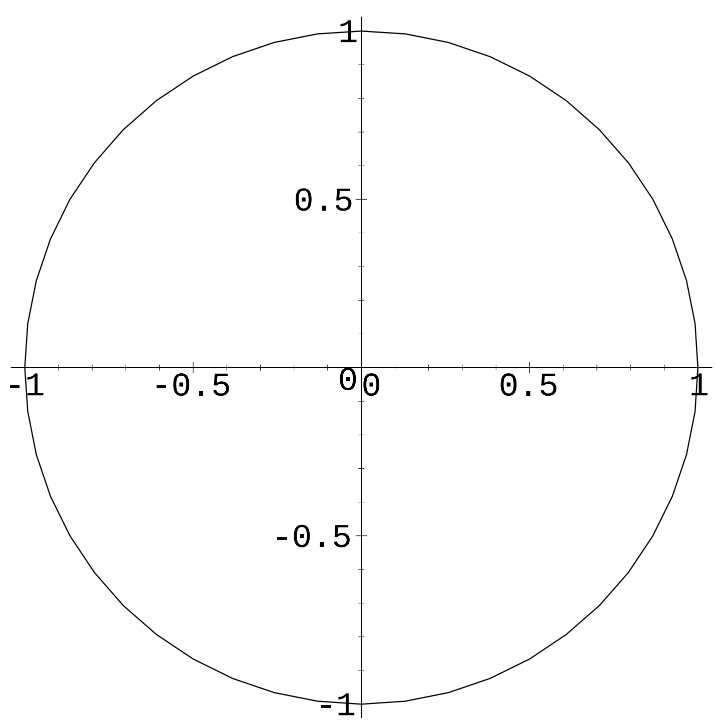
For instance, we get a circle of radius r using

$$x = \cos(t), \quad y = \sin(t)$$

We let the time t go from 0 to 2π.

In Maple, we can plot this directly:

```
> plot([sin(t),cos(t),t=0..2*Pi],scaling=constrained);
```



Parametric Representations Aren't Unique

There are always many ways of representing a curve parametrically.

A circle can also be represented as

$$x = \cos(-t^2), \quad y = \sin(-t^2)$$

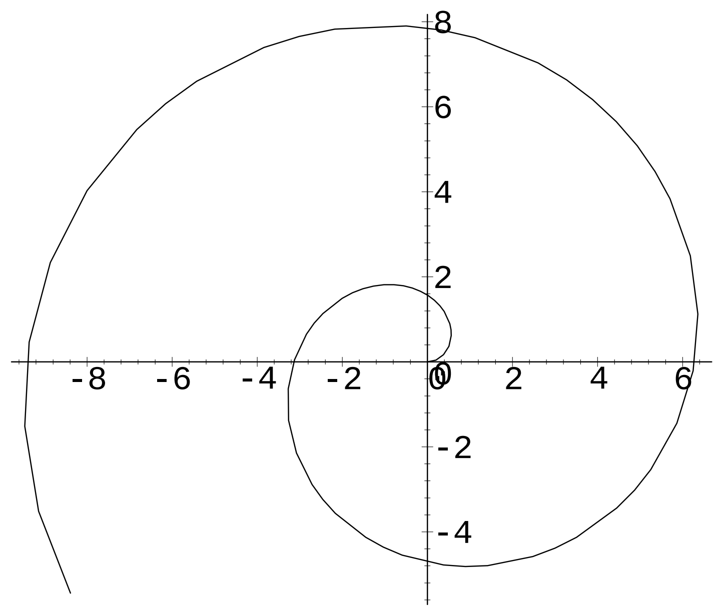
for t in the range 0 to $\sqrt{2\pi}$.

How does this representation differ in the way the circle is trace out?

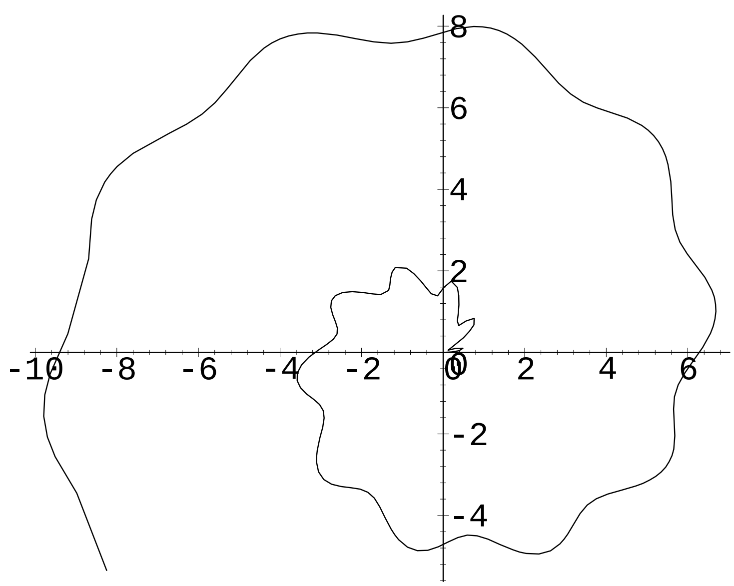
Two Parametric Spirals

With parametric representations, we can make all sorts of complicated curves:

```
> plot ([t*cos(t),t*sin(t),t=0..10], scaling=constrained);
```




```
> plot ([(t+sin(10*t)/3)*cos(t),(t+sin(10*t)/3)*sin(t),t=0..10],
        scaling=constrained);
```



Parametric Surfaces

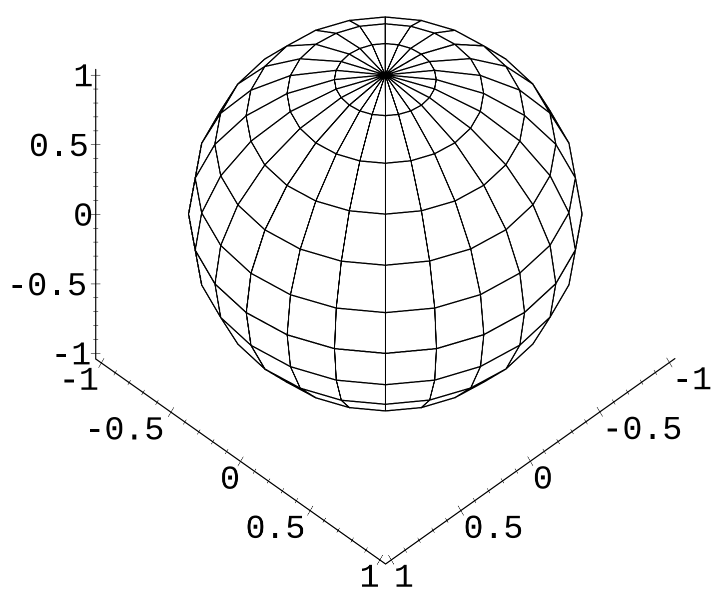
A parametric representation of a surface gives the coordinates x, y, and z as functions of two parameters, u and v.

We can get a sphere by

$$x = \cos(u)\cos(v), \;\; y = \cos(u)\sin(v), \;\; z = \sin(u)$$

In Maple:

```
> plot3d ([cos(u)*cos(v),cos(u)*sin(v),sin(u)],
>          u=-Pi..Pi, v=-Pi..Pi,
>          scaling=constrained, axes=frame, colour=black);
```



Representing a Line Segment

The simplest curve is a straight line segment.

We can represent most such line segments by explicitly giving y as a function of x:

$$y = ax + b$$

for some range, $x_0 \leq x \leq x_1$.

What are a and b? Why doesn't this always work?

We can also use a parametric representation, such as

$$x = (1-t)\,x_0 \; + \; t\,x_1, \quad y = (1-t)\,y_0 \; + \; t\,y_1$$

where t has the range $0 \leq t \leq 1$.

Here, $P_0 = (x_0, y_0)$ and $P_1 = (x_1, y_1)$ are the endpoints of the line segment. We can write this parameterization in vector notation as

$$L(t) = (1-t)\,P_0 \; + \; t\,P_1$$

Changing Parameterization

We can change from one parameterization:

$$x = F_x(t), \quad y = F_y(t)$$

to another by letting $t = g(u)$. Our new parameterization is

$$x = F_x(g(u)), \quad y = F_y(g(u))$$

We must find a range for u that corresponds to the old range for t. Will this always be possible?

For example, the line segment

$$x = 1 + 2t, \quad y = 3 - t$$

for $t \in [-8, 8]$ can be reparameterised by $t = u^3$:

$$x = 1 + 2u^3, \quad y = 3 - u^3$$

for $u \in [-2, 2]$.

Changing Parameterization in Maple

We can use Maple's symbolic computaton facilities to change a parameterisation:

```
> x := 1+2*t;
                                  x := 1 + 2 t

> y := 3-t;
                                   y := 3 - t

> plot([x,y,t=-8..8]);

> x:=subs(t=u^3,x);
                                              3
                                  x := 1 + 2 u

> y:=subs(t=u^3,y);
                                             3
                                   y := 3 - u

> solve(u^3=-8,u);
                                      1/2          1/2
                          -2, 1 + I 3   , 1 - I 3

> solve(u^3=8,u);
                                      1/2            1/2
                         2, - 1 + I 3   , - 1 - I 3

> plot([x,y,u=-2..2]);
```

Transforming Curves

Once we have a parametric representation of a curve, we can consider producing a new curve by a transformation — eg, a rotation.

Rotating (x, y) counterclockwise by an angle of a radians gives the following point, (x', y'):

$$\begin{aligned} x' &= \cos(a)x - \sin(a)y \\ y' &= \sin(a)x + \cos(a)y \end{aligned}$$

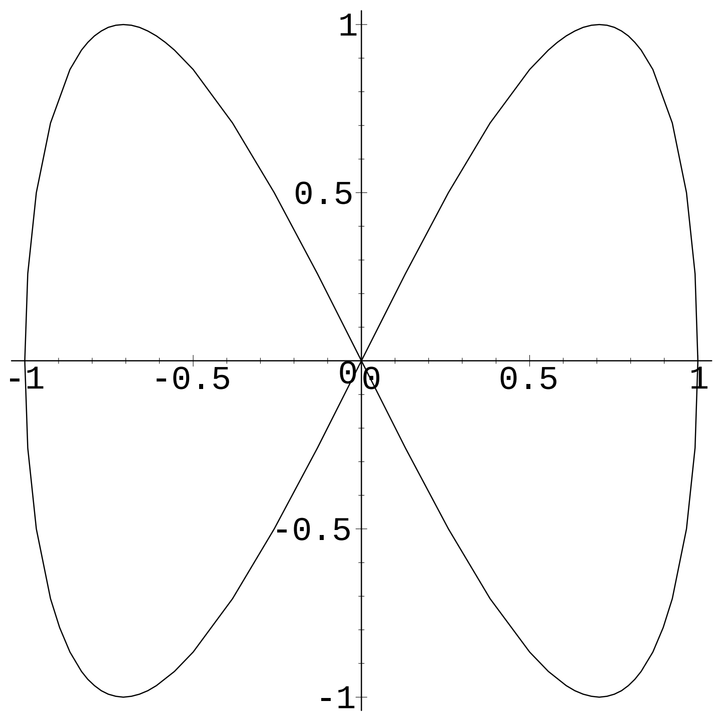
We can implement this in Maple with the following procedure:

```
rotate:=proc(x,y,a)
  cos(a)*x - sin(a)*y, sin(a)*x + cos(a)*y
end;
```

A Curve to Play With

Here's a pretty curve we can try rotating:

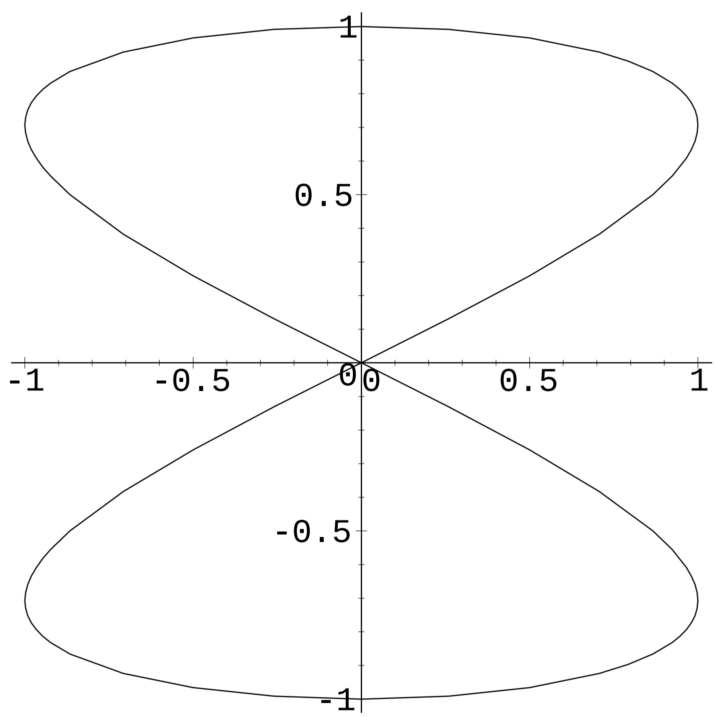
```
> plot ([sin(t),sin(2*t), t=-Pi..Pi],
>        scaling=constrained);
```



Rotating the Curve by $\pi/2$

Now we rotate by $\pi/2$ radians (90 degrees):

```
> plot ([rotate(sin(t),sin(2*t),Pi/2), t=-Pi..Pi],
>       scaling=constrained);
```



Maple does the rotation symbolicly, and manages to simplify the result:

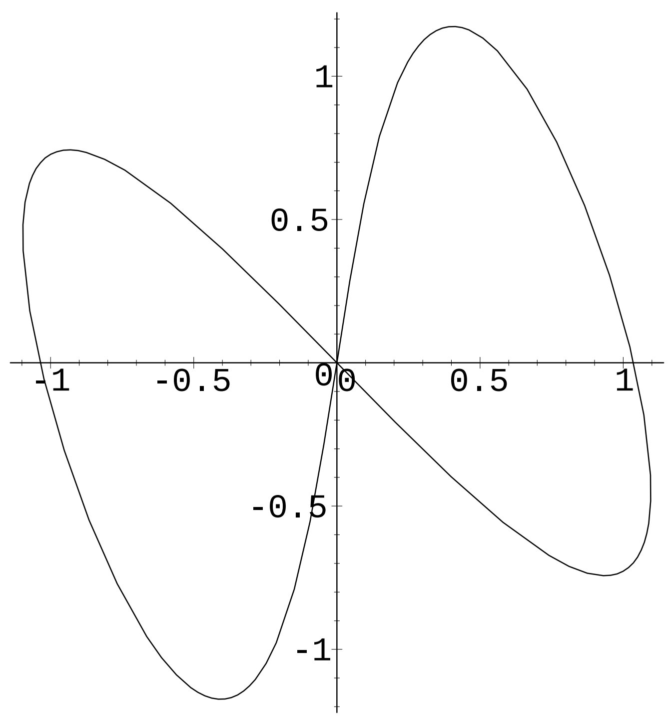
```
> rotate(sin(t),sin(2*t),Pi/2);
```

```
                    - sin(2 t), sin(t)
```

Rotating the Curve by 0.31

Now we try rotating by 0.31 radians:

```
> plot ([rotate(sin(t),sin(2*t),0.31), t=-Pi..Pi],
>        scaling=constrained);
```



Maple figures things out this time using floating point:

```
> rotate(sin(t),sin(2*t),0.31);

    .9523335699 sin(t) - .3050586364 sin(2 t),
         .3050586364 sin(t) + .9523335699 sin(2 t)
```