Maple Lecture 24. Two dimensional plots

Plotting is one of the main strengths of computer algebra. We use the package **plots**:

```
[> with(plots);
```

In this lecture we cover only the most essential topics, and only part of [1, Chapter 15]. The graphic tools in Maple are powerful enough for advanced mathematical visualization, see for instance [2] and the delightful [3].

24.1 Plotting Functions and Formulas

For example, suppose we wished to plot $f(x) = e^{-x^2}\sin(\pi x^3)$ over the interval $[-2, 2]$:

```
[> f := x -> exp(-x^2)*sin(Pi*x^3);
[> plot(f,-2..2);
```

When plotting a function, it suffices to give the range of the independent variable. For a formula, we must indicate the variable for the range, e.g.: `x = -2..2`, instead of `-2..2`. For example:

```
[> plot(f(z),z=-2..2);
```

Observe that with `f(z)` we obtained the formula defined by the function `f` evaluated at the symbol `z`.

24.2 Displaying and Animating Plots

We can assign plots to variables and save them for later display. Suppose we wish to add the amplitude e^{-x^2} to the plot above:

```
[> amp_plus := plot(exp(-x^2),x=-2..2,color=blue):
[> curve_plot := plot(f,-2..2):
[> amp_minus := plot(-exp(-x^2),x=-2..2,color=black):
[> display(amp_plus,curve_plot,amp_minus);
```

We can play with the frequency, adding an additional parameter k to f:

```
[> fxk := exp(-x^2)*sin(k*Pi*x^3);
[> movie := animate(fxk,x=-2..2,k=1..40,numpoints=200):
[> display(movie,insequence=true);
```

To play the movie, we click on the picture and we see a new toolbar appear. From this toolbar we select the play button. We can also let the movie repeat itself indefinitely.

24.3 Plotting around Singularities

To plot algebraic curves, we can use the command implicitplot:

```
[> implicitplot(x^3+y^3-5*x*y+1/5,x=-3..3,y=-3..3);
```

Not all curves will plot that easily:

```
[> f := (x^2+y^2)^3 + 5.12*(x^2+y^2)^2 - 5.15*(x^4-y^4) - 14.7456*y^2;
[> implicitplot(f,x=-3..3,y=-3..3);
```

As we adjust the range, Maple displays confusing pictures:

```
[> implicitplot(f,x=-11..11,y=-11..11);
```

This plot is made with hindsight (see below), but does still not look good:

```
[> implicitplot(f,x=-1.2..1.2,y=-1.2..1.2);
```

On this latest plot we see that the function does not seem to get drawn around the origin. The problem is the singularity at the origin $(0,0)$. In particular, it is easy to see that $\frac{\partial f}{\partial x}(0,0) = 0$ and $\frac{\partial f}{\partial y}(0,0) = 0$. To visualize the curve properly, we must convert to polar coordinates:

```
[> sf := subs({x=r*cos(t),y=r*sin(t)},f);
```

Since r=0 is just one point: the origin, we can divide out the common factor r^2:

```
[> nsf := normal(sf/r^2);
```

And now we solve for r:

```
[> sols := solve(nsf,r);
```

In this case, it suffices to take the first solution, to see the entire curve:

```
[> polarplot(sols[1]);
```

This curve is one of the curves "invented" by James Watt.

24.4 Drawing Sparse Matrices

Below we visualize a random matrix with entries consisting of zeros and ones.

```
[> randomize():                    # reset seed for random number generators
[> m := matrix(32,32,x->rand() mod 2):
[> plots[sparsematrixplot](m,axes=none,scaling=constrained,symbolsize=6);
```

With this facility we can make systematic patterns.

24.5 Making your own drawings

With cubic B-splines we can draw anything we want. For example,

```
[> x := [-1.0,0.0,1.0,2.0,1.6,1.2,0.0,-1.6,-2.1, \
         -2.8,-2.0,-0.9,0.0,1.0,0.0,0.0,-1.0];
[> y := [0.0,1.0,1.0,0.0,-1.5,-2.1,-2.8,-2.5, \
        -2.0,-0.7,2.3,3.7,5.0,8.0,9.0,-6.0,-7.2];
[> s := CurveFitting[BSplineCurve](x,y,t);
[> plot(s,scaling=constrained,axes=none);
```

The command **BSplineCurve** of the package **CurveFitting** creates a piecewise function through the given x and y coordinates. Since both lists for x and y must have the same length, it is often better to zip the lists, since BSplineCurve also accepts the data as a list of points.

24.6 Assignments

1. The "neoid" is defined by $r = at + b$ in polar coordinates. Give the Maple commands
   (a) to make a plot for $a = 0.2$ and $b = 0.5$, for $t = 0 \ldots 6\pi$; and
   (b) to produce an animation of 10 frames, for $a = 0.2$ and for b going from 0.1 to 1 (also for $t = 0 \ldots 6\pi$).

2. Do `f := 4*x*y^2 + 2*x^3 - x^2;` and give Maple commands
   (a) to convert the formula for `f` into an equation in polar coordinates;
   (b) to solve the equation obtained in (a) for the radius;
   (c) to make the plot using the solution(s) obtained in (b).

3. The folium of Descartes is defined by $x^3 + y^3 = 3xy$ and has tangent $x + y + 1 = 0$.
   (a) Draw the folium in rectangular coordinates taking the right-hand side of `numpoints=` high enough to see a nice smooth curve around the origin. Take $[-2, +2]$ as the range for both x and y.
   (b) Make a plot for the tangent $x + y + 1 = 0$, using a color and a linestyle different from the plot of the folium. Display both plots on the same figure window. You may want to extend the original range $[-2, +2]$ to see the folium better approaching the tangent.
   (c) Convert the folium into polar coordinates and plot. Be careful in choosing a good range for the parameter. Compare the size of the plot data structure in polar coordinates with size of the data structure in rectangular coordinates, recalling the high value needed for `numpoints` to make the other plot in rectangular coordinates look good.

4. The (i, j)-th entry Pascal matrix is defined by $\begin{pmatrix} i+j-2 \\ j-1 \end{pmatrix}$.

   Create a sparse 32-by-32 matrix, setting the even entries in the Pascal matrix to zero, and the odd entries to one. The sparse matrix plot should show the Sierpinski gasket.

5. Cassini ovals are sets of points for which the product of the distances to two fixed points are constant. Taking the points on the x-axis with coordinates $(-a, 0)$ and $(+a, 0)$, then for a constant c, the Cassini ovals satisfy

$$((x-a)^2 + y^2) \times ((x+a)^2 + y^2) = c^4.$$

   (a) Make several plots for various choices for a and c. Distinguish the cases $0 < a < c$ and $c < a < c\sqrt{2}$. Display all plots in the same picture.
   (b) Find the representation of these curves in polar coordinates.
   (c) Concerning the quality of the plots, is it easier for Maple to work with the original rectangular coordinates or are the polar coordinates better suited?

6. Take a large piece of paper and draw a big letter k in your own hand writing. Impose a grid on the sheet and take sufficiently many points on your big letter k so that the B-spline curve passing through those points resembles your own k faithfully.

References

[1] A. Heck. *Introduction to Maple*. Springer-Verlag, third edition, 2003.

[2] G. Klimek and M. Klimek. *Discovering Curves and Surfaces with Maple*. Springer-Verlag, 1997.

[3] V. Rovenski. *Geometry of Curves and Surfaces with Maple*. Birkhäuser, 1999.