

The Maple Commands conformal, complexplot and complexplot3d

Mth 417 Complex Analysis
Spring 1998 - Bent E. Petersen
April 6, 1998

Maple has a rich selection of plot commands. Three of them, conformal, complexplot, complexplot3d, are designed specifically for dealing with complex numbers and functions. Many of the others could be pressed into service with some ingenuity, for example, contourplot and contourplot3d.

```
[ > restart;
```

We must load the plots package in order to have all plot features enabled. We use a colon to suppress the output, though if you want to see what commands are enabled by the plots package, you can terminate the command with a semicolon instead.

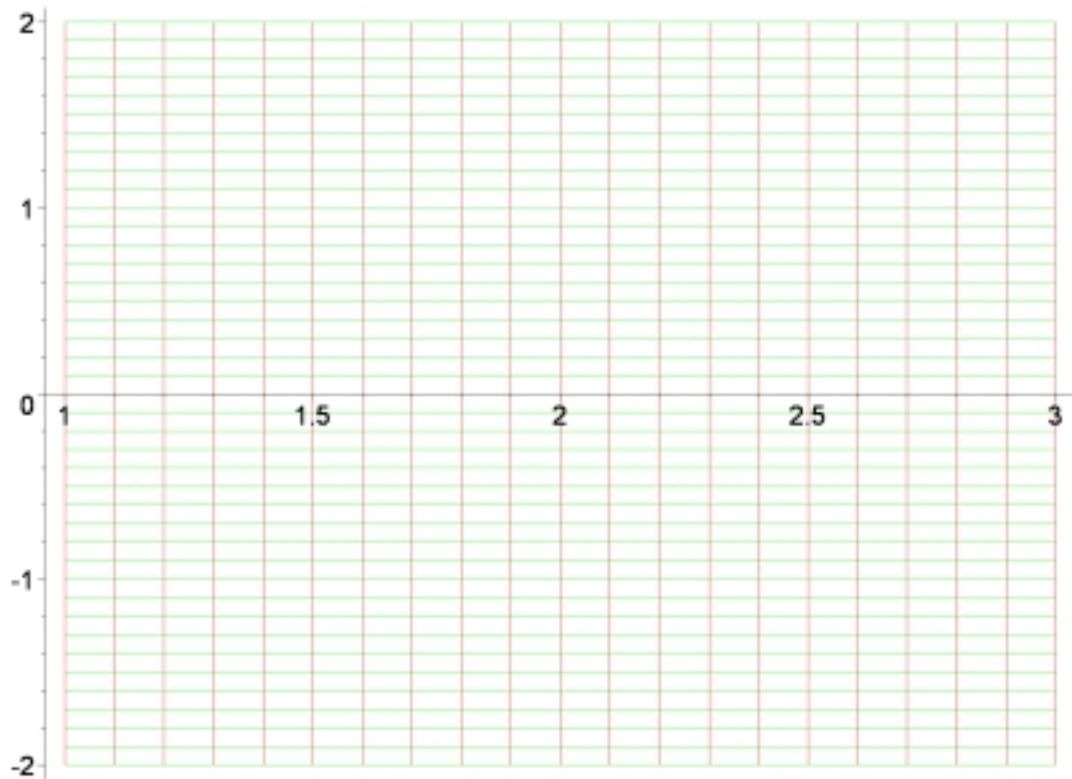
```
[ > with(plots):
```

The conformal command

The conformal command plots the image of a rectangular grid under a complex function.

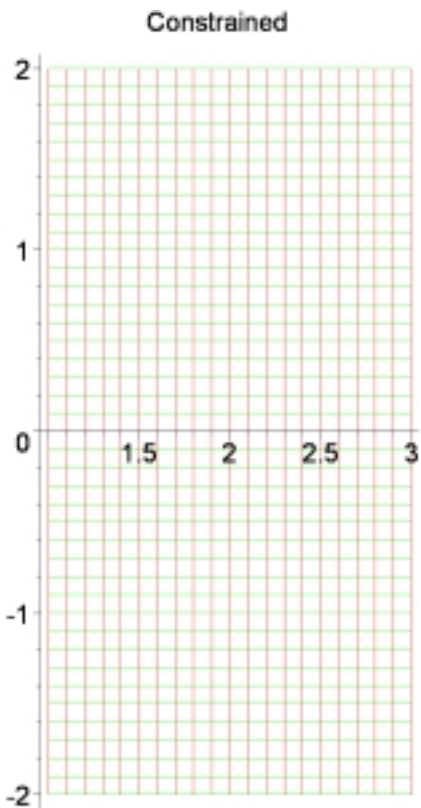
Let's start with something really simple - the identity:

```
[ > conformal(z, z=1-2*I..3+2*I, grid=[21,41]);
```



We see Maple plotted the values of the identity function in a square with lower left corner $1-2i$ and upper right corner $3+2i$ along 21 vertical lines and 41 horizontal lines. The scales along the two axes are different. We can have the same horizontal and vertical scales by adding an option:

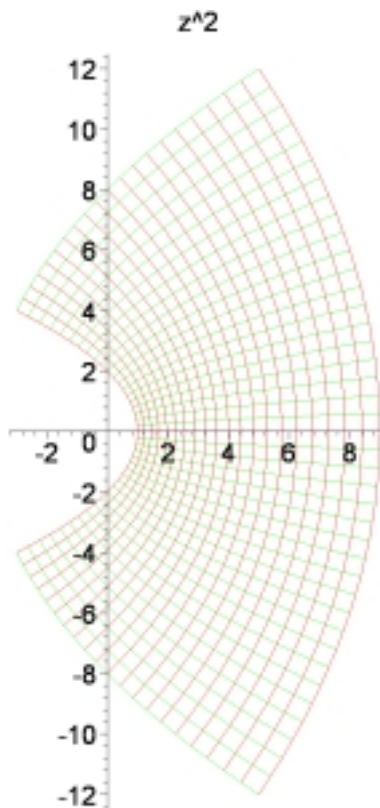
```
> conformal(z, z=1-2*I..3+2*I, grid=[21,41],
  scaling=constrained, title='Constrained');
```



There are numerous other options.

Well, lets plot something a bit more interesting than the identity. We will plot the polynomial z^2 evaluated on the lines indicated above. What will result will be a family of curves which we think of as the image under the squaring map of the lines indicated above.

```
> conformal(z^2,z=1-2*I..3+2*I, grid=[21,41],  
scaling='constrained',title='z^2');
```

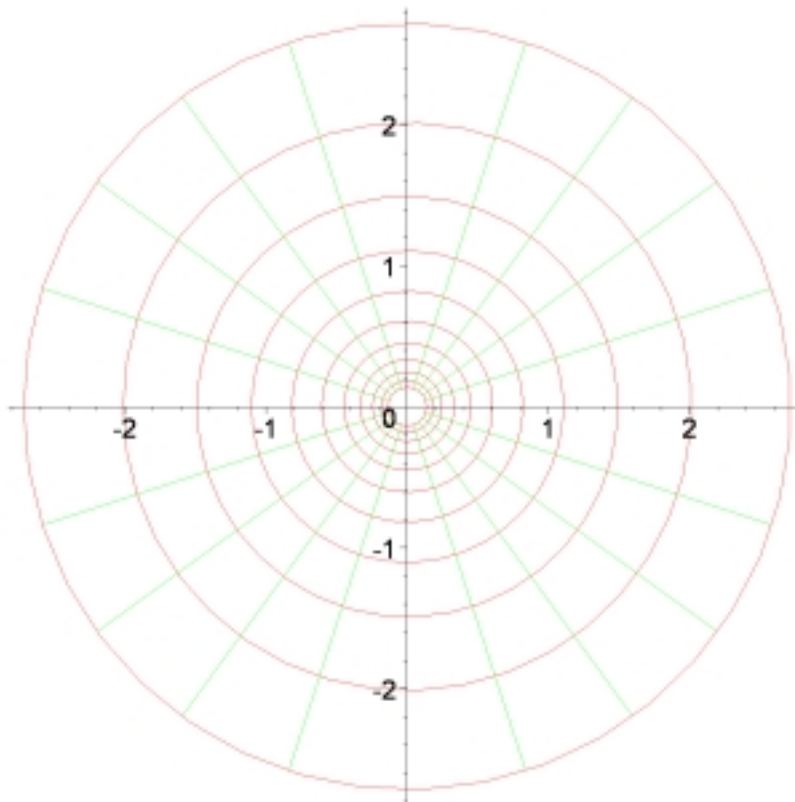


Here we have used constrained scaling so we can see that the contours are orthogonal. We can see quite well how the real part of z^2 becomes increasingly negative as the imaginary part of z tends to plus or minus infinity.

It is unfortunate that conformal is restricted to plotting the images of vertical and horizontal lines, since often we want to consider are the images of other curves, frequently circles. One way to do this is to use `complexplot` (see next section) to plot individual curves. Another way is to plot a suitable composition, where the first function maps the rectangular grid to the desired grid and the second one is the function we want to study.

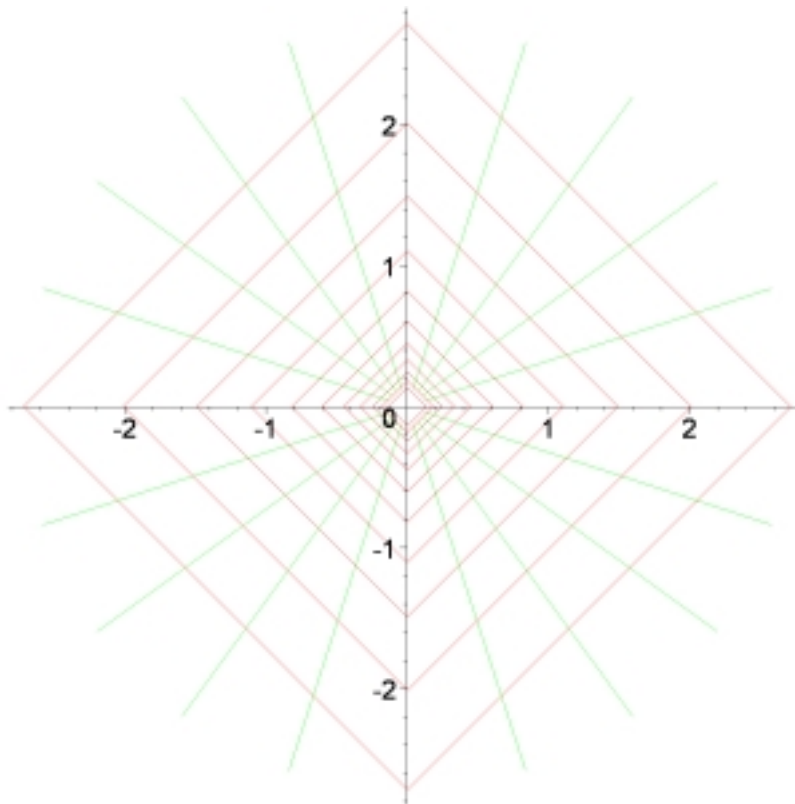
Let's look at the images of circles for example. First we define a family of circles with center at the origin - the exponential is convenient:

```
> h:=w->exp(w);
                                     h := exp
> conformal(h,-2-Pi*I..1+Pi*I,grid=[11,21],
  scaling='constrained',numxy=[15,40]);
```



The `numxy` command specifies the number of points to be plotted in each direction, that is, it is a measure of the resolution. If you do not specify it then you may get some circles with a lot of flat parts (the default is `numxy=[15,15]`). For example, let's try `[5,5]` above:

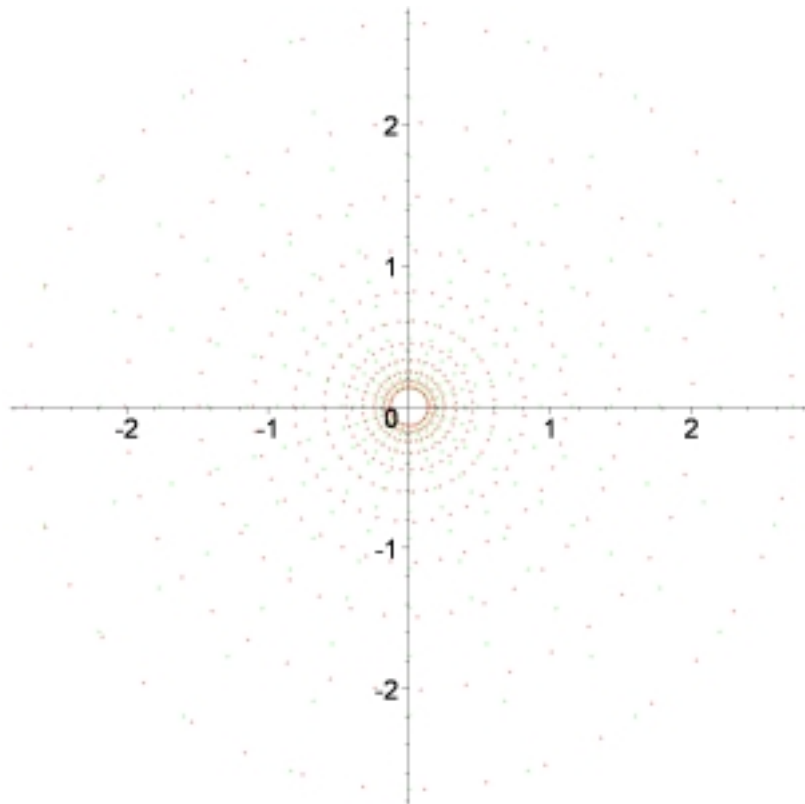
```
> conformal(h, -2-Pi*I..1+Pi*I, grid=[11,21],  
scaling='constrained', numxy=[5,5]);
```



Ouch! That is bad - but no worse than one might expect. One should usually set numxy to an appropriate value.

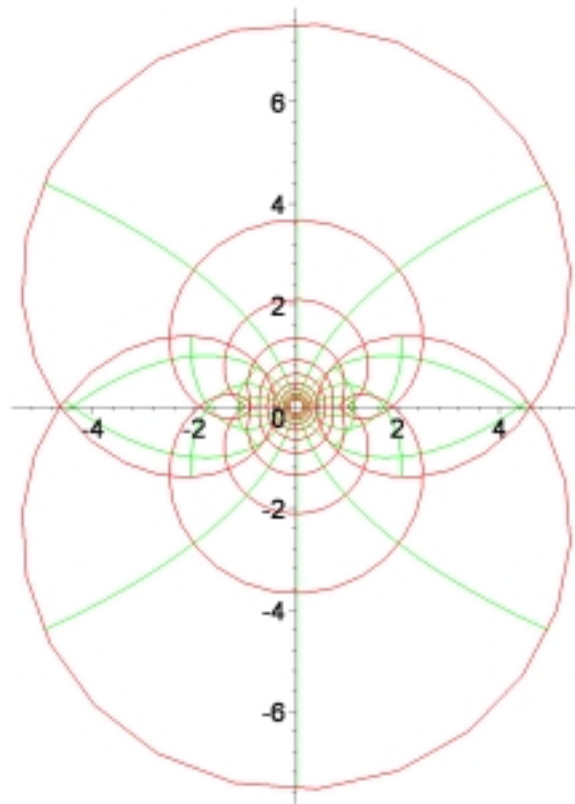
Note a point style graph is also available:

```
> conformal(h,-2-Pi*I..1+Pi*I,grid=[11,21],  
scaling='constrained',numxy=[15,40],style='point');
```



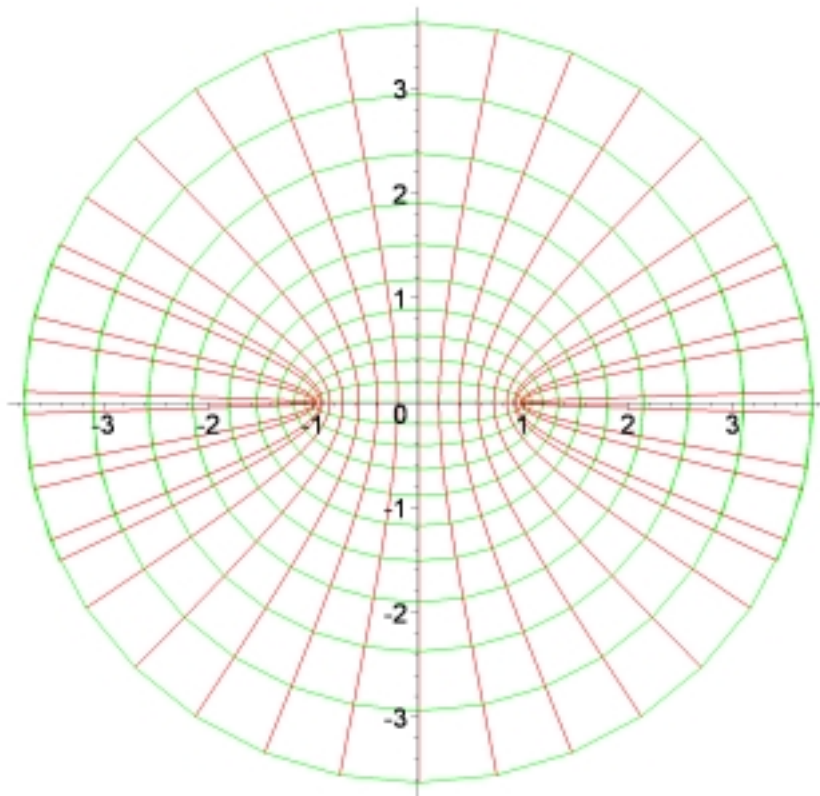
Now let's look at the values of $\sin(z)$ on circles as explained above:

```
> conformal(sin(h(w)),w=-2-Pi*I..1+Pi*I,grid=[11,21],
scaling='constrained',numxy=[80,80],thickness=2);
```



If you experiment you will see that we need to set `numxy` quite high to get an acceptable image. Thus the command will be very slow, but at least we are no longer limited to looking at conformal images of boring grid lines! Still, it might be interesting to look at what the sine function looks like on a rectangular grid:

```
> conformal(sin,-2-2*I..2+2*I, grid=[21,21], scaling='constrained',
thickness=2);
```



Note that

```
> Re(Sin(x+I*y))=evalc(Re(sin(x+I*y)));
  Im(Sin(x+I*y))=evalc(Im(sin(x+I*y)));
```

$$\Re(\text{Sin}(x + I y)) = \sin(x) \cosh(y)$$

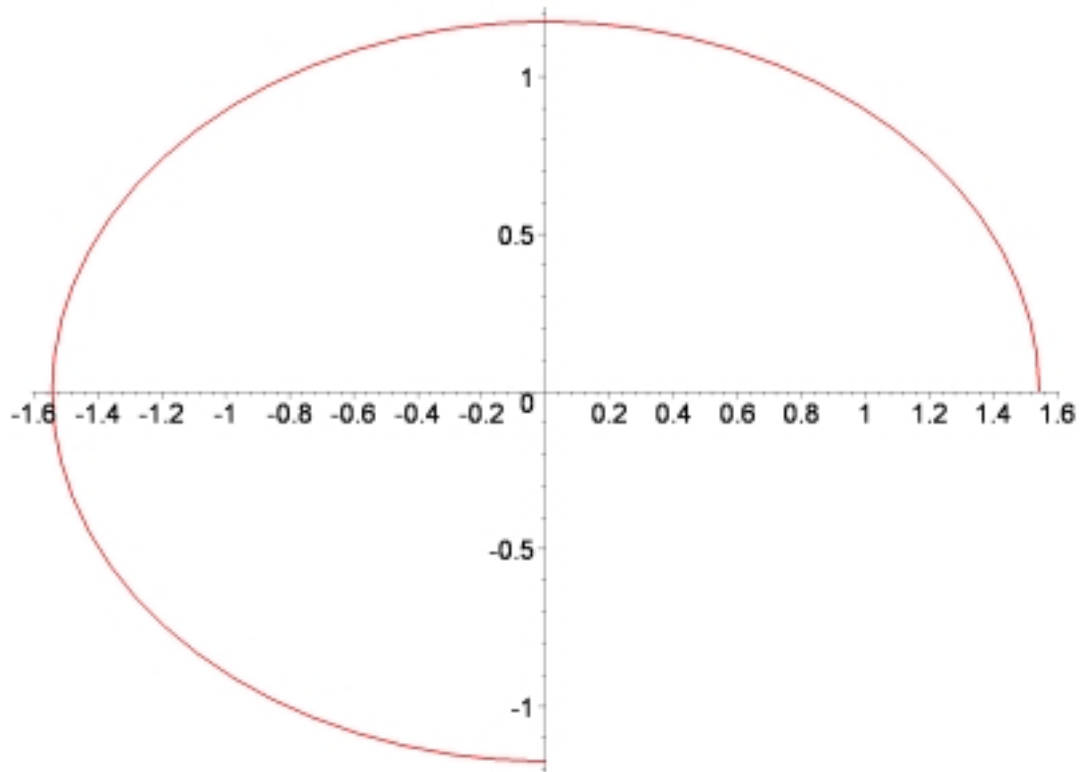
$$\Im(\text{Sin}(x + I y)) = \cos(x) \sinh(y)$$

Thus the closed curves above are ellipses and the open curves are hyperbolae.

The complexplot command

The complexplot command plots complex valued functions of a real variable, that is, complex curves.

```
> complexplot(sin(x+I),x=-Pi..Pi/2,thickness=3);
```

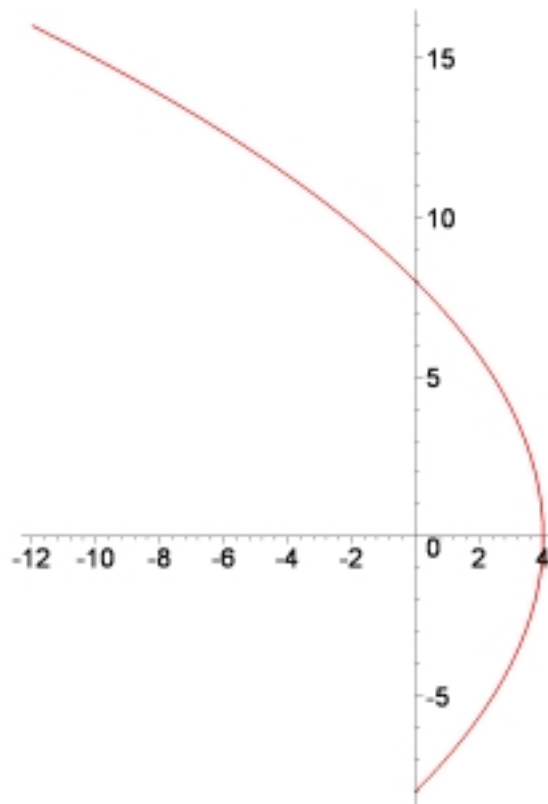


Let's look at some other complex curves:

```
> g:=t->(2+I*t)^2;
```

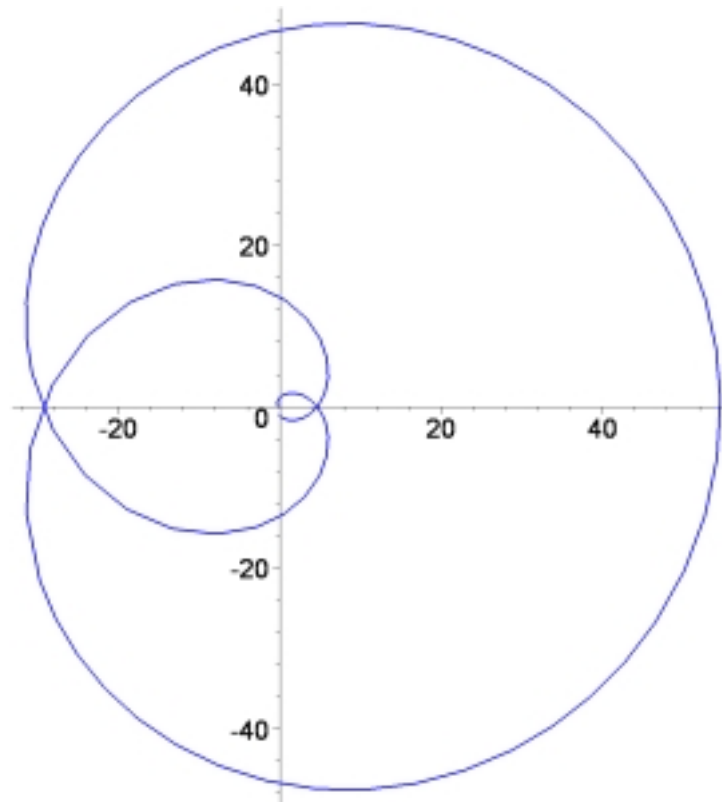
$$g := t \rightarrow (2 + It)^2$$

```
> complexplot(g(t),t=-2..4, scaling='constrained',thickness=3);
```



It is not difficult to see this curve is a parabola. Suppose we wanted to look at the exponential along this parabola. We have

```
> complexplot(exp(g(t)),t=-2..4,  
scaling='constrained',color=blue,thickness=3);
```



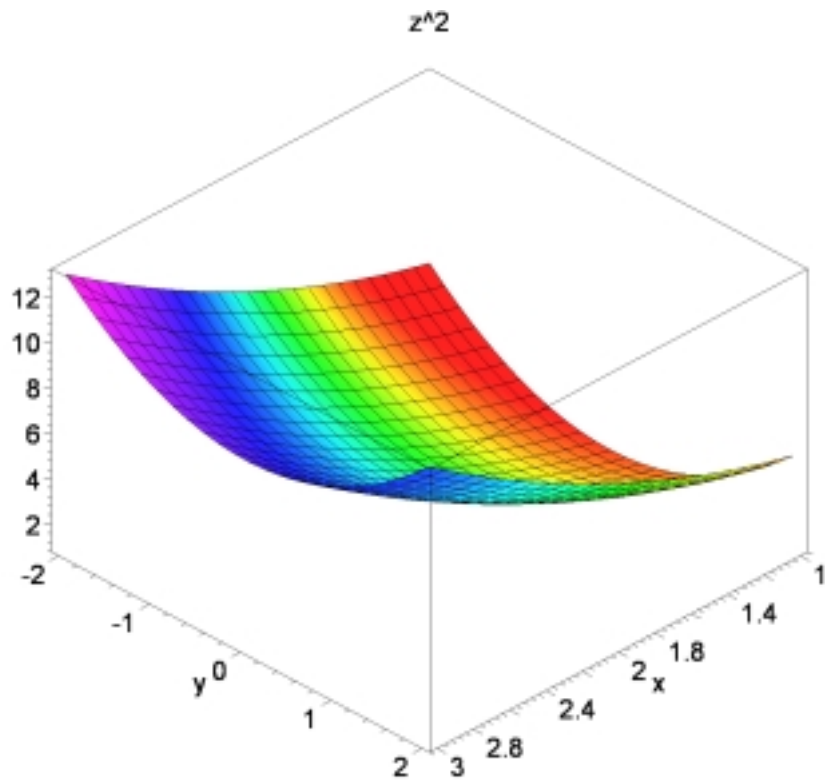
Very nice!

The complexplot3d command

The complexplot3d command draws 3 dimensional representations of complex functions of a complex variable.

>

```
> complexplot3d(z^2, z = 1 - 2*I .. 3 + 2*I, axes=boxed, title='z^2');
```



What is being plotted here is the modulus (absolute value or magnitude) of z^2 , but with an interesting twist - the graph is colored according to the values of the argument of z^2 . It's hard to interpret, but may be useful sometimes. This command has numerous options and is polymorphic, that is, alters its behavior when called with different types of arguments. See the Maple help.

[>