Implicit Differentiation

Douglas Meade, Ronda Sanders, and Xian Wu
Department of Mathematics

Overview

    This lab provides experience working with functions defined implicitly.

Maple Essentials

- Important Maple commands introduced in this lab are:

Command	Description	Example
`display`	display plots in a single plot (with plots package)	`display([F,G],title='Fig1');`
`implicitplot`	create graph of function defined implicitly (with plots package)	`implicitplot(x*y=0,x=0..1,y=0..1);`
`implicitdiff`	compute derivatives of functions defined implicitly	`implicitdiff(f,y,x);` `implicitdiff(f,x,y);`
`fsolve`	compute a solution of equations numerically	`fsolve({f=1,g=x^2},{x,y});` `fsolve({f,g},{x,y},{x=0..1,y=0..2});`
`with`	load a Maple package	`with(plots):  with(plots);`

- The *ImplicitDifferentiation* maplet is available from the course website:

    http://www.math.sc.edu/calclab/141L-F05/labs → ImplicitDifferentiation (TAMU)

Related course material

    §4.1 Implicit Differentiation (Pages 237-240) of the textbook.

Activities

1. Start a Maple session and type `restart;`.

2. Type `with(plots):`. This loads the maple `plots` package and allow us to plot points, use the display command, use the commands for implicitly-defined functions, and more. Notice that we used ':' instead of ';'. The difference is that the maple does not display the output with ':'.

3. **Example 5 on page 239:** a) Use implicit differentiation to find dy/dx for the Folium of Descartes $x^3 + y^3 = 3xy$. b) Find the equation of the tangent line to the Folium of Descartes at the point $(3/2, 3/2)$. (Then graph the curve, the point, and the tangent line with a viewing window of (-3,3)x(-4,3) as shown in Figure 4.1.5 on page 239.) c) At what point(s) in the first quadrant is the tangent line to the Folium of Descartes horizontal?

    We will redo this example together using Maple as follows:

- First, for part a), simply assign the Folium of Descartes to, say, `FD`, then use command `implicitdiff` to find dy/dx with:
  ```
  FD:=x^3 +y^3 =3*x*y;
  dydx:=implicitdiff(FD,y,x);
  ```
  (Notice that `implicitdiff(FD,x,y);` computes dx/dy.)

- Next, to find the tangent line, we need a point and a slope. We have the point $(3/2, 3/2)$. Since we already have the derivative, we find the slope `m` by evaluating dy/dx at this point with:
  ```
  m:= eval(dydx, {x=3/2,y=3/2});
  ```

- Therefore, by the point-slope formula, the equation of the tangent line is obtained with:
  ```
  L:= m * (x-3/2) + 3/2;
  ```

- To finish part b), we write commands that will plot the curve, the point, and the tangent line with:
  ```
  P1:= implicitplot(FD, x=-3..3, y=-4..3):
  P2:= pointplot([3/2,3/2], color=green, symbolsize=15):
  P3:= plot(L, x=-3..3,y=-4..3, color=blue):
  ```
  (Notice that we used ':' instead of ';' since we don't want any output at this point.)

- These plots can then be displayed on a single plot with:
  ```
  display([P1, P2, P3]);
  ```

- From the graph, we can see that the answer to part c) is a point located approximately at $(1.2, 1.5)$. Since this point is on the curve and the $dy/dx = 0$ at this point, we can find it's location by solving those two equations with:
  ```
  fsolve({FD,dydx=0},{x,y},{x=1..2,y=1..2});
  ```
  (For a numerical solution in a specified region, `fsolve` in general does a better job than `solve`.)

4. **Lab problem 1:** Find the equation of the tangent line to the curve $2(x^2 + y^2)^2 = 25(x^2 - y^2)$ at the point $(3, 1)$. Then graph the curve, the point, and the tangent line with a viewing window of (-5,5)x(-2,4).

5. **Lab problem 2:** Find all points where the tangent line to the graph of $x^2 y - xy^2 = 2$ is horizontal or vertical. (Hint: The tangent line is vertical where $dx/dy = 0$.)

Assignment

Exercises 17, 20, 26, 39, and 40 (pages 241-242).

Hint for 39 and 40: For part a), start with a big range for both x and y in `implicitplot` to see the size of the view window the graph will display and then re-plot the graph with that view window for a better plot. For part c), you also need to specify different regions in `fsolve` according to your graph to get all solutions, as `fsolve` only computes one solution at a time.