AM33 Brown University

Direction Fields

We can associate the solution of the differential equation with the trajectory of a particle starting from any one of its points and then always moving in the direction of the field. The path of such a particle is called a **streamline** of the field. Thus the function defined by a streamline is an integral of the differential equation to which the field applies.

A curve on the plane, at each point of which it has a constant slope, is called an **isocline** of the differential equation $y' = f(x, y)$. That is, an isocline is a set of all points (x, y) satisfying the equation $f(x, y) =$ `Constant`.

Available software packages are very helpful for practical drawings of direction fields and isoclines.

Matlab

In order to plot a direction field one must first create a file[1], say `function1.m`, containing the rate function $f(x, y)$. Let us take the example

$$f(x, y) = xy^2. \tag{1}$$

and consider the differential equation $y' = f(x, y) = xy^2$ which has the general solution $y(x) = -2/(x^2 + C)$, where C is an arbitrary constant. Thus, this file will contain the following three lines (including comments):

```
% Function for direction field:          (function1.m)
  function F=function1(x,y);
  F=x.*y.^2;
```

Notice that $.*$ is used for component-wise multiplication of the vector containers x and y. Now we are ready to proceed with the m-file which displays the direction field of the differential equation $y' = f(x, y)$ for $f(x, y) = xy^2$.

```
% Direction field of function1.m          (direction_field1.m)
clear; % clears the variables
N=15; % number of samples points
xmin=-2; %  |
xmax=10; %  |
ymin=-4; %  | domain specification
ymax=4; %  |
[x,y]=meshgrid(xmin:(xmax-xmin)/N:xmax,ymin:(ymax-ymin)/N:ymax);
dy=function1(x,y);      % y-component
dx=ones(size(x));     % x-component (=1)
L=sqrt(dx.^2 + dy.^2);        % length of vector
L=L+1e-10;                    % just in case L==0
dy=dy./L;                     % normalize dy
dx=dx./L;                     % normalize dx
quiver(x,y,dx,dy);            % matlab routine
title('Direction field');
```

The MATLAB command `meshgrid(a:dx:b,s:dy:d)` creates a grid of values in the rectangle $[a.b] \times [c, d]$ spaced dx units apart in abscissa and dy units apart in ordinate.
The MATLAB command `quiver(x,y,dx,dy)` draws the arrow (vector) (dx, dy) at each point (x, y) of a meshgrid.
The `inline`(*expr*) constructs an inline function object from the MATLAB expression contained in the string *expr*. The input argument to the inline function is automatically determined by searching *expr* for an isolated lower case alphabetic character, other than i or j, that is not part of a word formed from several alphabetic characters. If no such character exists, x is used. If the character is not unique, the one closest to x is used. If two characters are found, the one later in the alphabet is chosen.

Another way to plot the direction field is as follows:

[1] All files in MATLAB must have extension m.

```
clear all
[x,y]=meshgrid(-2:0.1:2,-2:0.1:2);
scale = 0.3;
u = scale +0.0*x;
v = scale*function1(x,y);
hold on
quiver(x,y,u,v)
y1 = inline('-2/x^2'), fplot(y1,[-2,2,-2,2])
hold off
```

For isoclines, the MATLAB routine `solve` will suffice.

Also many universities have developed matlab routines that facilitate drawing direction fields for differential equations. For example, John Polking at Rice University has produced DFIELD and PPLANE programs for such a purpose. He maintains the web site `http://math.rice.edu/~polking` with the matlab .m files at `http://math.rice.edu/~dfield`.

Maple

MAPLE is particularly useful for producing graphical output. It has a dedicated command for producing flow fields associated with a first order differential equation. For example, the commands

```
>> with(DEtools): with(plots):
>> dfieldplot(diff(y(x),x)= x*(y(x)^2), y(x), x=-1..1, y=-2..2, arrows=medium);
```

plot the direction field for the differential equation (1). You may draw a particular solution in the same picture by typing `DEplot(equation, x-range, y-range, [y(Pi/2)=1])`.

Isoclines can be found also using the `solve` command. For example

```
>> ? solve
```

generates the help page for `solve`, while the corresponding isoclines are found with

```
>> sol:=solve('x*y^2',y);
```

Then, `sol` can be plotted with respect to the independent variable, x.

Mathematica

To draw the direction field corresponding Eq. (1), the following commands in MATHEMATICA should be executed. First, we have to load the `Graphics` package by typing command

```
<< Graphics`PlotField`
```

and pressing `SHIFT-ENTER`. Then we call the function `PlotVectorField`. The following

```
PlotVectorField[f, { x, x0, x1, (xu) } , { y, y0, y1, (yu)}, (Options)]
```

produces a vector field plot of the two-dimensional vector function f. For example, we can write:

```
PlotVectorField[{Sin[x], Cos[y]}, {x, 0, Pi}, {y, 0, Pi}]
```

Look in the Mathematica help desk for some common options. For example, the `ScaleFunction -> (1&)` option in the statement rescales each vector to a unit vector at each point.

This direction field can be combined with one or several solutions of the corresponding differential equation by using the `Show` command.

Winplot

There is a friendly graphical program, **Winplot**, written by Richard Parris, a teacher at Phillips Exeter Academy in Exeter, New Hampshire. Mr. Parris generously allows free copying and distribution of the software, and provides frequent updates. The latest version can be downloaded from the web site:
`http://math.exeter.edu/rparris/winplot.html`.

Although the program is free, it is of top quality and easy to use. These instructions expand on those found in the program's help menu; they discuss some techniques of two-dimensional plotting, useful in an introductory course in differential equations. It is assumed that the reader has already installed the program in Microsoft Windows, and is familiar with the basic workings of that operating system.

MVT

MVT is an online Java application and can be found at `http://amath.colorado.edu/java/mvt3.0/`. Developed by the Department of Applied Mathematics at the University of Colorado, this program is able to analyze a number of different kinds of problems and does not require previous computing experience.

Examples

Example #1

Using `dfield6.m`, for the problem

$$y' = y^2 - x \tag{2}$$

we arrive at figure 1.

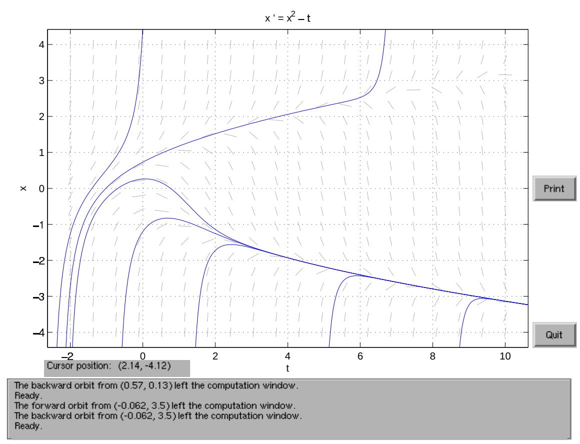

Figure 1: Example #1 with integral curves.

Example #2

Using the Matlab routine described above for the equation

$$y' = xy^2 \tag{3}$$

we arrive at figure 2.

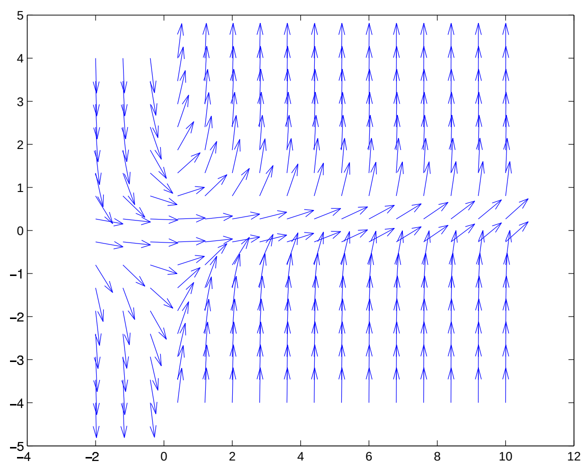

Figure 2: Example #2.