Tutorial 8 Differential Equations

This tutorial presents an introduction to ordinary differential equations (ODEs). The tools for studying ODEs have changed greatly in the past decades. The following examples have been taken from *Linear Algebra and Differential Equations using MATLAB* by Martin Golubitsky and Michael Dellnitz. The tutorial runs as a picture-book guide through many different types of DEs. While you may not understand the complete mathematical basis for the pictures, you should be able to understand what the pictures represent. Ask the TAs to explain the pictures if you are confused. Focus on the following concepts: a line field, a phase plot, autonomous DEs, asymptotic stability, equilibrium points, eigendirections. This tutorial requires three MATLAB functions developed at Rice University: dfield7, pline, pplane7. Download all the Rice ODE functions from the course website and install in the directory C:\work (note that information in this directory is not kept permanently). Versions of dfield and pplane are available for the different MATLAB releases. MATLAB's function ode45 is also described.

8.1 Single First Order Differential Equation

A single first-order differential equation (DE) has the form:

$$\frac{dx}{dt}(t) = f\left(t, x(t)\right).$$

This equation is first-order since only the first derivative of the function $x(t)$ appears in the equation. If the second derivative appeared in the equation, then it would be second-order. Frequently, the above equation is written in the simplified form:

$$\frac{dx}{dt} = f\left(t, x\right).$$

The most important point is that $x(t)$ is a function of t (which is referred to as the time variable). When the right-hand side, the function f, does not depend *explicitly* on the time variable, t, the equation is called **autonomous**. An example of an autonomous DE is:

$$\frac{dx}{dt}(t) = l\, x(t).$$

An example of a non-autonomous DE is:

$$\frac{dx}{dt}(t) = x(t)^2 - t.$$

Line Fields or Direction Fields

The following material requires MATLAB. Start by typing (7 refers to the MATLAB version):

```
>> dfield7
```

The DFIELD Setup window should appear. In the window, find the differential equation:

$$\frac{dx}{dt} = x(t)^2 - t.$$

We would like to start with the differential equation:

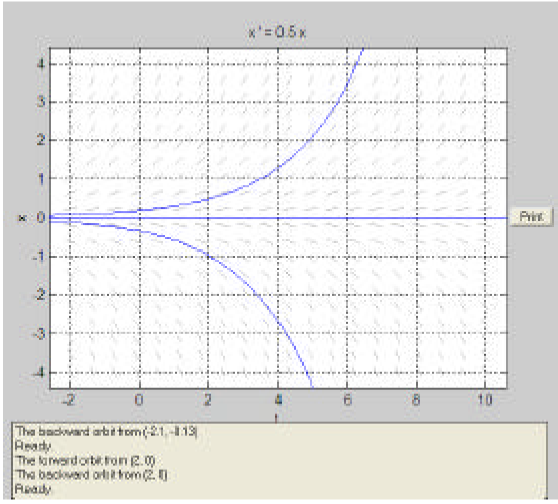
$$\frac{dx}{dt}(t) = 0.5x(t).$$

To change the differential equation, click in the Setup window where the right-hand side $x^2 - t$ is located and type $0.5 * x$. Now click on the button *Proceed*. This will bring up the window DFIELD Display showing the line field for the DE (shown in the figure below). In this figure, the x-axis is the vertical axis and the t-axis is the horizontal axis. In the DE

$$\frac{dx}{dt}(t) = 0.5x(t),$$

the slope of the tangent line to the solution is given by the right-hand side of the DE. This information is used to sketch the tangent lines at each point (t, x) in the tx-plane. More specifically, a small line segment is drawn at each point with the slope determined by the right-hand side of the DE. The resulting graph is called a **line field** or **direction field**.

Solution curves going through a specific point (t_0, x_0) in the tx-plane can be shown by clicking with the mouse button on the corresponding point in the graph. The solution curve is first computed forward in time and then backward in time. Click on a point near $(t, x) = (0,0)$ with $x > 0$ and then click on a point near $(t, x) = (0,0)$ with $x < 0$. Solution curves through these points should then be drawn. To compute a solution curve corresponding to the special case where $x_0 = 0$, go to the DFIELD Display window and click on the *Options* menu and select *Keyboard input*. Type in the initial values $t = 2$ and $x = 0$. Click on *Compute* and the solution curve will be drawn. It is important that you understand how it is that this line field corresponds to the correct solution of the DE which is $x(t) = x_0 e^{0.5t}$.



The Difference between Autonomous and Nonautonomous DEs

Autonomous DEs do not depend explicitly on the time variable and may be written

$$\frac{dx}{dt} = f(x).$$

Suppose there is one solution, $x_1(t)$, to such a DE and then consider a time-shifted version of $x_1(t)$ which we shall call $x_2(t)$: $x_2(t) = x_1(t - t_0)$. It is important that you understand the steps in the following equation:

$$\frac{dx_2}{dt}(t) = \frac{dx_1}{dt}(t - t_0) = f\left(x_1(t - t_0)\right) = f\left(x_2(t)\right).$$

The above equation shows that $x_2(t)$ is also a solution to the DE. In other words, $x_1(t)$ with a time-shift also satisfies the DE. This time-shift property of autonomous DEs does not generally apply to nonautonomous DEs.

Visualising the Difference

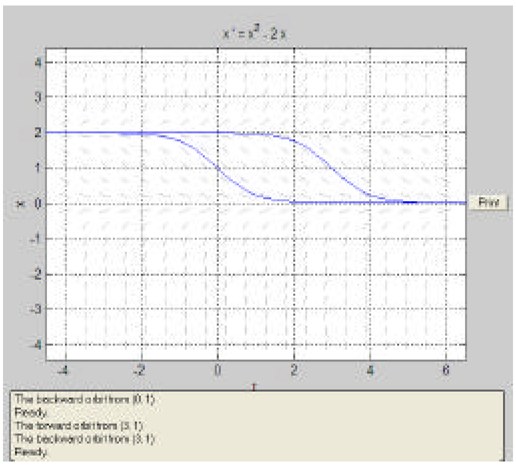
We will now visualize the difference between autonomous and nonautonomous DEs. Run DFIELD at the MATLAB prompt and change the differential equation to:

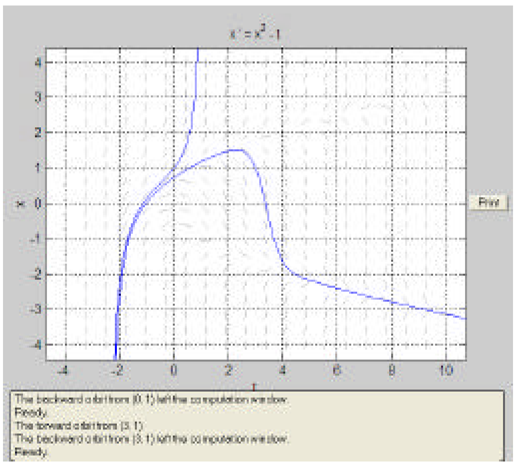
$$\frac{dx}{dt} = x^2 - 2x.$$

Note, you will have to type $2*x$ not $2x$. This equation is an autonomous DE. In the Setup window, choose a minimum value for t of -4 and a maximum value for t of 6. Using *Keyboard Input*, plot solution curves for the initial conditions $(t_0, x_0) = (0,1)$ and also $(t_0, x_0) = (3,1)$. The results show that these two solutions are time-shifted versions of each other. Go back to the Setup Window and change the DE to:

$$\frac{dx}{dt} = x^2 - t.$$

This DE is a nonautonomous. Choose a minimum value for t of -4 and a maximum value for t of 10. Using *Keyboard Input*, plot solution curves for the initial conditions $(t_0, x_0) = (0,1)$ and also $(t_0, x_0) = (3,1)$. The graphs of the two solution curves are not related by a time-shift. See the figures below for examples.





Exercise 1

Use DFIELD to compute the line field for the following equations and record whether the equations are autonomous or nonautonomous:

(a) $\frac{dx}{dt} = xt \quad \left(0 \le t \le 2,\ -1 \le x \le 3\right)$

(b) $\frac{dx}{dt} = x\sin(x) \quad \left(-2 \le t \le 10,\ -4 \le x \le 4\right)$

(c) $\frac{dx}{dt} = x\cos(t) \quad \left(-2 \le t \le 6,\ -4 \le x \le 4\right)$

Exercise 2

Use DFIELD to compute the line field for the DE:

$$\frac{dx}{dt} = x^3 - 2t^2x - t \quad \left(-2 \le t \le 2,\ -2 \le x \le 3\right).$$

Based on the line field, draw by free-hand your expected solution for the curve through (-2,1). Use *Keyboard Input* to check your free-hand solution. Now use the zoom feature to determine the solution (the x-value) for this curve at $t = 1$. In the DFIELD Display window, select *Edit* and then *Zoom In* . Draw a small box on the curve at $t = 1$. Recursively repeat this procedure until you can read three significant figures. Answer: 0.561.

Phase Space and Equilibria

Consider the equilibrium points for solutions of autonomous, first-order DEs. These DEs have the form:

$$\frac{dx}{dt} = f(x).$$

At the MATLAB prompt, type:

```
>> pline.
```

If you cannot fully see the window, type:

```
>> set(gcf,'position',[100 250 500 300]).
```

Since the pline function assumes autonomous DEs, there is no time variable. Instead, there is only the integration time. Note that pline adopts the convention that all solutions start at $t = 0$. There is also one free parameter, lambda. In the PLINE Setup window, change the default equation to $\frac{dx}{dt} = x\left(l - x^2\right)$ by typing: $x*(lambda - x\wedge 2)$. Change the minimum and maximum values of x to -2 and 2. Change the integration time to 2 and the value of lambda to +1. Click on the *Proceed* button. All that you will see is the x-axis. Click on the x-axis near -2 and then near -1.5. You will see coloured balls that track the motion of the solution, $x(t)$, for $0 \le t \le 2$. $x(0)$ is the point that you clicked on. Click on the x-axis near -0.5 and -0.1 and then click on the x-axis near 0.5, 1.5, and 2. Clearly the solutions to the DE move toward +1 and -1. Based on the functional form of the DE,

$$\frac{dx}{dt} = x(1-x^2) ,$$

it should be clear that $\frac{dx}{dt}$ will be zero when x is 0, +1, -1. This means that there is no motion at these points and 0,+1,-1, are called **equilibrium points**. Equilibrium points such as +1 and -1 are **asymptotically stable** because points near them move back to the equilibrium point. Equilibrium points such as 0 are **asymptotically unstable** because points near them move away from the equilibrium point. For the DE, $\frac{dx}{dt} = f(x)$, we say that the equilibrium point, x_0, (with $f(x_0) = 0$) is hyperbolic if $\frac{df}{dx}(x_0) \neq 0$. Using the approximation,

$$f(x) \approx \frac{df}{dx}(x_0)(x - x_0) ,$$

you should be able to explain the rule that if $\frac{df}{dx}(x_0) < 0$, then the equilibrium point is asymptotically stable and if $\frac{df}{dx}(x_0) > 0$, then the equilibrium point is asymptotically unstable.

Exercise 3

(a) The DE, $\frac{dx}{dt} = x^2$, has an equilibrium point at the origin. Use pline to determine for which initial starting points, $x(0) = x_0$, the motion is toward the origin.

(b) Determine the equilibrium points for $\frac{dx}{dt} = x^3 - 2x^2 - 8x$ and their stability.

8.2 Linear Systems with Two Equations

An autonomous system of two differential equations has the form:

$$\frac{dx}{dt}(t) = f\left(x(t), y(t)\right)$$
$$\frac{dy}{dt}(t) = g\left(x(t), y(t)\right) .$$

An autonomous system of two *linear* differential equations has the form:

$$\frac{dx}{dt}(t) = Ax(t) + By(t)$$
$$\frac{dy}{dt}(t) = Cx(t) + Dy(t) \ .$$

The simplest system of two linear DEs is uncoupled, which means that the equation for $\frac{dx}{dt}$ does not depend on y and the equation for $\frac{dy}{dt}$ does not depend on x. Uncoupled systems of two autonomous and linear DEs have the form:

$$\frac{dx}{dt}(t) = Ax(t)$$
$$\frac{dy}{dt}(t) = Dy(t) \ .$$

The solution for this system of two uncoupled linear DEs is:

$$x(t) = x_0 e^{At}$$
$$y(t) = y_0 e^{Dt}$$

Phase Space Plots

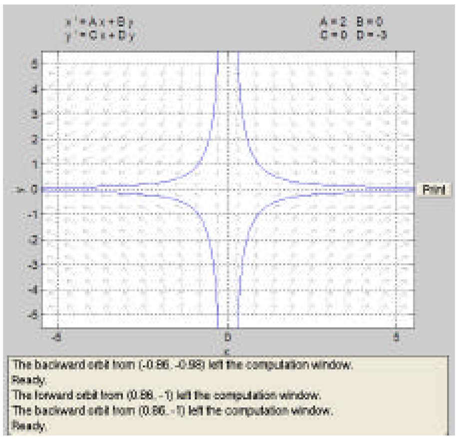
We can explore the geometry of the solutions to two linear DEs using a phase space plot in MATLAB. We treat $\left(x(t), y(t)\right)$ as the position of a particle in the xy-plane at time t. We then graph the point $\left(x(t), y(t)\right)$ in the plane as t varies. When looking at the phase space plots, we call the solution curves trajectories because we are thinking in terms of a particle moving in the plane as time changes. Run the correct version of PPLANE at the MATLAB prompt:

```
>> pplane7.
```

(There are versions of PPLANE available for the different versions of Matlab.)

Click on *Gallery* and choose *linear system.* Set the parameters B and C to 0, making the system uncoupled. Set A to 2 and D to -3. Click on *Proceed* . The PPLANE Display window should open up. The figure shows that the horizontal axis is $x(t)$ and the vertical axis is $y(t)$. Click at close as you can to the four points $\left(x(0), y(0)\right) = \left(\pm 1, \pm 1\right)$.

PPLANE will draw the trajectories through these points. Note that as $t \to \infty$, all solutions approach the x-axis and that as $t \to -\infty$, all solutions approach the y-axis (see figure below). You should be able to see this from the directions the arrows point. For this system, the origin is called a *saddle* point. If $A < 0, D > 0$ the origin would still be a saddle point, only the role of the x-axis and y-axis would be reversed. Change parameter A to -1 and click on *Proceed.* Click again on the four points $\left(x(0), y(0)\right) = \left(\pm 1, \pm 1\right)$. As $t \to \infty$, all trajectories go to the origin and in this case the origin a sink point. Change parameter A to 2, parameter D to 3, and click on *Proceed*. This time as $t \to \infty$, all trajectories go away from the origin toward infinity. The origin is a source point.



Time Series Plots

After showing several solution curves for the linear system with the origin as a source point, return to the PPLANE Display window, select *Graph*, and then click on *Composite*. A cross-hair cursor will appear. Click with the cross-hair cursor on one of the solution curves. A PPLANE t-plot window should then appear showing a time series plot for the solution curve. By selecting different options under the *Graph* heading in the PPLANE t-plot window, different types of time series plots can be shown. The time series plots show $x(t)$ and $y(t)$ as a function of *t*.

Coupled Linear Systems

Use pplane7 to display the phase space plot for the following system of DEs:

$$\frac{dx}{dt}(t) = -x(t) + 3y(t)$$
$$\frac{dy}{dt}(t) = 3x(t) - y(t) \; .$$

Display several solution curves by clicking on points in the xy-plane. Note that as $t \to \infty$, the solution curves approach the diagonal line $x = y$ and that as $t \to -\infty$, the solution curves approach the diagonal line $x = -y$. These lines are invariant in the sense that solutions starting on these lines remain on them for all time. Verify that this is the case by first selecting *Solutions* in the PPLANE Window and then *Keyboard Input* and then selecting the point $(x(0), y(0)) = (1,1)$. An invariant line such as the ones described here for a linear system of DEs is referred to as an **eigendirection**.

We now raise two very important questions:

(1) Do eigendirections always exist?

(2) How can we find eigendirections?

Let's answer the first question. Use pplane7 to display the phase space plot for the following system of DEs:

$$\frac{dx}{dt}(t) = -x(t) - 2y(t)$$
$$\frac{dy}{dt}(t) = 3x(t) - y(t)\ .$$

Click on several points in the xy-plane to show several solution curves. Clearly there is no eigendirection.

Exercise 4

Choose the linear system in PPLANE and set A=0, B=1, and C=-1. Find values for D such that (excepting the origin itself) all solution curves appear to :

(a) spiral into the origin

(b) spiral away from the origin

(c) form circles around the origin

8.3 Linear DEs and Matrix Exponentials

In order to answer questions about eigendirections for autonomous systems of linear DEs matrix theory is required. Indeed, the solutions can be written down elegantly using matrix exponentials. To begin with, we can write down an autonomous system of linear DEs using matrix notation. Let $\mathbf{X}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$ and $\mathbf{M} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$. The system of DEs

$$\frac{dx_1}{dt}(t) = Ax_1(t) + Bx_2(t)$$
$$\frac{dx_2}{dt}(t) = Cx_1(t) + Dx_2(t)$$

can then be written more compactly as

$$\frac{d\mathbf{X}}{dt} = \mathbf{M} \cdot \mathbf{X}\,.$$

The solution to the above equation can be written in terms of matrix exponentials:

$$\mathbf{X}(t) = e^{t\mathbf{M}}\mathbf{X}_0\,.$$

This may be your first time to encounter matrix exponentials. In order to define a matrix exponential, we use the power series for e^t:

$$e^t = 1 + t + \frac{1}{2!}t^2 + \frac{1}{3!}t^3 + \cdots\ .$$

In other words, we *define* the matrix exponential $e^{\mathbf{M}}$ by:

$$e^{\mathbf{M}} = \mathbf{I} + \mathbf{M} + \frac{1}{2!}\mathbf{M}^2 + \frac{1}{3!}\mathbf{M}^3 + \cdots\ .$$

The above sum is well-defined in terms of matrix multiplications and it turns out that the infinite series of matrix exponentials converges for all n by n matrices. We can now define $e^{t\mathbf{M}}$ as:

$$e^{t\mathbf{M}} = \mathbf{I} + t\mathbf{M} + \frac{t^2}{2!}\mathbf{M}^2 + \frac{t^3}{3!}\mathbf{M}^3 + \cdots \ .$$

With this definition it follows that:

$$\begin{aligned}
\frac{d}{dt}e^{t\mathbf{M}} &= \frac{d}{dt}(\mathbf{I}) + \frac{d}{dt}(t\mathbf{M}) + \frac{d}{dt}\left(\frac{t^2}{2!}\mathbf{M}^2\right) + \frac{d}{dt}\left(\frac{t^3}{3!}\mathbf{M}^3\right) + \cdots \\
&= \mathbf{0} + \mathbf{M} + t\mathbf{M}^2 + \frac{t^2}{2!}\mathbf{M}^3 + \frac{t^3}{3!}\mathbf{M}^4 + \cdots \\
&= \mathbf{M} \cdot \left(\mathbf{I} + t\mathbf{M} + \frac{t^2}{2!}\mathbf{M}^2 + \frac{t^3}{3!}\mathbf{M}^3 + \cdots\right) \\
&= \mathbf{M} \cdot e^{t\mathbf{M}}.
\end{aligned}$$

Therefore, as claimed earlier the solution to $\frac{d\mathbf{X}}{dt} = \mathbf{M} \cdot \mathbf{X}$ is $\mathbf{X}(t) = e^{t\mathbf{M}}\mathbf{X}_0$.

We can use MATLAB's matrix exponential command expm to compute matrix exponentials. For example, to compute $e^{t\mathbf{M}}$ for $\mathbf{M} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ and $t = \frac{p}{4}$ use:

```
>> M = [0 -1;1 0];
>> t = pi/4;
>> expm(t*M)
ans =
   0.7071   -0.7071
   0.7071    0.7071
```

Exercise 5

(a) Let $\mathbf{M} = \begin{bmatrix} 2 & 4 \\ -1 & 2 \end{bmatrix}$. Using MATLAB, write a for loop to sum the first 10 terms in the exponential $e^{\mathbf{M}}$. Compare the result with expm(M).

(b) Compute $e^{t\mathbf{M}}$ for $\mathbf{M} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ and t = pi, pi/2 , pi/3. Based on your results, write $e^{t\mathbf{M}}$ in terms of $\cos(t)$ and $\sin(t)$.

(c) Compute $e^{t\mathbf{M}}$ for $\mathbf{M} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$ and $t = -2, -1, 0, 1, 2$. Do your results show that $e^{t\mathbf{M}} = \begin{bmatrix} e^{2t} & 0 \\ 0 & e^{1t} \end{bmatrix}$?

(d) Compute $e^{t\mathbf{M}}$ for $\mathbf{M} = \begin{bmatrix} 1 & 1 \\ 2 & -1 \end{bmatrix}$ and $t = 1.0, 1.5, 2.5$. Does $e^{\mathbf{M}} e^{1.5\mathbf{M}} = e^{2.5\mathbf{M}}$? Does $e^{1.5\mathbf{M}} = 1.5 e^{\mathbf{M}}$? (It better not!)

8.4 Phase Portraits of Planar DEs

The phase plots for various autonomous systems of two linear DEs with constant coefficients are explored. The phase plots show that the qualitative behaviour of the DEs is entirely determined by the eigenvalues and eigenvectors of their coefficient matrix. The set of phase plots for planar DEs divides into two kinds: hyperbolic and nonhyperbolic. Hyperbolic systems consist of saddles, sinks, and sources, while nonhyperbolic systems consist of centers, saddle-nodes, and shears. It is not possible in this tutorial to look at all of these phase portraits. However, we will look at three phase portraits: a saddle, sink, and center.

To begin, let $\mathbf{M} = \begin{bmatrix} 1 & 2 \\ 4 & 1 \end{bmatrix}$. Compute the eigenvalues and eigenvectors of $\mathbf{M}$ in MATLAB as follows:

```
>> M = [1 2;4 1];
>> [V,D] = eig(M)
V =
   0.5774  -0.5774
   0.8165   0.8165
D =
   3.8284        0
        0  -1.8284
```

The columns of $\mathbf{V}$ are the eigenvectors and the values along the diagonal of $\mathbf{D}$ are the eigenvalues. Each eigenvalue corresponds to an eigenvector. The eigenvalues and eigenvectors in the same column belong together. In the above case, the eigenvalue 3.8284 corresponds to the vector $\begin{bmatrix} 0.5774 \\ 0.8165 \end{bmatrix}$. Note that eigenvalues can be complex numbers.

Hyperbolic DEs

Hyperbolic DEs have a coefficient matrix, $\mathbf{M}$, in which all eigenvalues have nonzero real parts.

Saddle

Use pplane7 to draw the phase plot for the following system of DEs:

$$\frac{d\mathbf{X}}{dt} = \mathbf{M} \cdot \mathbf{X}, \text{ where } \mathbf{M} = \begin{bmatrix} 2 & 1 \\ -1 & -4 \end{bmatrix}.$$

This DE has the phase portrait of a linear saddle. There are two eigendirections. Draw several solution curves until the eigendirections become obvious. Compute the eigenvalues and eigenvectors of $\mathbf{M}$ in MATLAB:

```
>> M = [2 1;-1 -4];
>> [V,D] = eig(M)
```

The eigendirections of the phase plot correspond to the eigenvectors of the matrix **M.** The eigenvector with a positive eigenvalue corresponds to the eigendirection for $t \to \infty$. The eigenvector with a negative eigenvalue corresponds to the eigendirection for $t \to -\infty$. The positive and negative eigenvalues identifies the saddle portrait. PPLANE has the ability to find the stable and unstable trajectories of a saddle (mainly used for nonlinear DEs, but we will use it anyway). In the PPLANE Display window, click on *Edit* and then click on *Erase all solutions.* Click on *Solutions* and then click on *Find an equilibrium point.* Cross-hairs will appear; click on the graph close to the point (0,0). PPLANE7 should put a dot at the saddle point. Click on *Solutions* and then click on *Plot stable and unstable orbits.* Cross-hairs will appear; click on the saddle point. PPLANE7 should draw the stable and unstable trajectories of the saddle point.

Spiral Sink

Use PPLANE to draw the phase plot for the following system of DEs:

$$\frac{d\mathbf{X}}{dt} = \mathbf{M}\cdot\mathbf{X}, \text{ where } \mathbf{M} = \begin{bmatrix} -1 & 2 \\ -5 & 0 \end{bmatrix}.$$

Compute a few trajectories by clicking on a few points in the xy-plane. There is no visible sign of an eigendirection. The phase portrait is characterized by spiraling trajectories approaching the origin. Select *Graph* from the menu bar and the click on *Composite* to show a time series plot for one trajectory. (Use the cross-hair cursor to click on a trajectory.) The spiraling is realized as a damped oscillation in either the x or y-directions.

NonHyperbolic DEs

Nonhyperbolic DEs have a coefficient matrix, $\mathbf{M}$, in which there is a zero eigenvalue or an eigenvalue that is purely imaginary.

Center

Use PPLANE to draw the phase plot for the following system of DEs:

$$\frac{d\mathbf{X}}{dt} = \mathbf{M}\cdot\mathbf{X}, \text{ where } \mathbf{M} = \begin{bmatrix} 0 & -3 \\ 3 & 0 \end{bmatrix}.$$

Compute a few trajectories by clicking in the xy-plane. This is an entirely new kind of solution: it is a time periodic solution referred to as a center. The trajectories circle about the origin in a couterclockwise fashion. Select *Graph* from the menu bar and click on *Composite* to show a time series plot for one trajectory. There is a periodic oscillation along both the x and y-directions.

Exercise 6

Let $\mathbf{M}$ be the following coefficient matrices. Calculate the eigenvalues and eigenvectors for the matrices and determine whether the origin is a saddle, sink, source, or center. Use pplane7 to verify your answer.

(Hint: If the eigenvalues both have a positive real part, then the exponential will grow; if the eigenvalues both have a negative real part, then the exponential will shrink; if one eigenvalue has a positive real part and the other a negative real part, then the exponential will grow in one direction and shrink in the other; if the eigenvalues both have zero real part, then the exponential will not grow or shrink.)

(a) $\begin{bmatrix} -2 & 2 \\ 0 & -1 \end{bmatrix}$ (b) $\begin{bmatrix} 1 & -8 \\ 2 & 1 \end{bmatrix}$ (c) $\begin{bmatrix} 9 & -11 \\ -11 & 9 \end{bmatrix}$ (d) $\begin{bmatrix} 8 & 0 \\ -5 & 3 \end{bmatrix}$ (e) $\begin{bmatrix} -2 & -2 \\ 4 & 1 \end{bmatrix}$ (f) $\begin{bmatrix} 2 & -2 \\ 4 & -2 \end{bmatrix}$

8.5 Higher Dimensional Systems and ode45

The MATLAB functions DFIELD, PLINE, and PPLANE compute solutions to 1-D and 2-D differential equations. To compute solutions in higher dimensional systems MATLAB provides the command ode45. The command ode45 is used to solve initial value problems. Suppose we want to solve the following linear system of DEs:

$$\frac{d\mathbf{X}}{dt} = \mathbf{M} \cdot \mathbf{X}, \text{ where } \mathbf{M} = \begin{bmatrix} -0.25 & 3 & 0 \\ -3 & -0.25 & 0 \\ 0 & 0 & -0.2 \end{bmatrix}.$$

In order to use ode45, we must write a MATLAB function that calculates the right-hand side of the linear system of DEs:

```
function f = myodefun(t,x)
A = [-0.25 3 0;-3 -0.25 0;0 0 -0.2];
f = A*x;
```

Note that the first argument of the function must be t even though in this example the time variable does not explicitly appear. In order to use ode45, we must specify the time interval over which to integrate a solution and the starting point. In this case, we choose [0,100] as our time interval and $\left(x_1(0), x_2(0), x_3(0)\right) = [2,-1,-1]$ as the starting point (note that the starting point must be entered as a column vector):

```
>> [t,x] = ode45('myodefun',[0 100],[2,-1,-1]. '); .
```

We can view the phase plot for this single solution as follows:

```
>> plot3(x(:,1),x(:,2),x(:,3));
>> xlabel('x1'); ylabel('x2'); zlabel('x3');
```

The most important error criterion for an ode solver is the *relative error* which is defined as the absolute error between the numerical and exact solution divided by the size of the exact solution. Mathematically the relative error is written:

$$\text{relative error} = \frac{\left\| \vec{x}_{num}(t_0) - \vec{x}_{exact}(t_0) \right\|}{\left\| \vec{x}_{exact}(t_0) \right\|} .$$

We can set the tolerance on the relative error using the odeset command in MATLAB:

```
>> [t,x] = ode45('myodefun',[0 100],[2,-1,-1]. ',odeset('RelTol',1e-7)); .
```

Exercise 7

(a) ode45 solves ODEs using the Runge-Kutta method. Read the text, IM6 p451-453, IM7 p496-499 and explain what it means to use a fourth-order Runge-Kutta method.

(b) Read the text, IM6 p463-464, IM7 p508-509. Explain in your own words how ODE solvers are used to solve equations with order higher than two. Explain what it means if the equation has been expressed in state-variable form (a.k.a. Cauchy form).

Exercise 8

(a) Asymptotically stable solutions to systems of two DEs include equilibrium points and centers. With systems of higher dimensional DEs, asymptotically stable solutions can take more complicated forms. However, there are no analytical formulas expressing these “quasiperiodic” solutions. Therfore, numerical methods are required to view such solutions. Recently, it was discovered that nonlinear systems of three DEs can have a two-frequency quasiperiodic solution. Without computers, such solutions would never be seen. Consider the following system of three DEs:

$$\frac{dx_1}{dt} = (x_3 - 0.7)x_1 - 3.5x_2$$
$$\frac{dx_2}{dt} = 3.5x_1 + (x_3 - 0.7)x_2$$
$$\frac{dx_3}{dt} = 0.6 + x_3 - 0.33x_3^3 - \left(x_1^2 + x_2^2\right)\left(1 + 0.25x_3\right)$$

Use ode45 to find a trajectory for this system of DEs. Integrate over the time interval [0,100] and use [0.1,0.03,0.001] . ' as the starting point. The phase space plot should show motion on a torus.

(b) Classifying the kinds of solutions that can occur asymptotically in autonomous systems of first-order DEs is a difficult task and a topic for current research. Recently, it was found that there is a type of solution that exhibits a sensitive dependence on initial conditions. This type of solutions is referred to as chaotic. The classic example of chaos is the Lorenz system:

$$\frac{dx_1}{dt} = s\ (x_2 - x_1)$$
$$\frac{dx_2}{dt} = r\, x_1 - x_2 - x_1 x_3$$
$$\frac{dx_3}{dt} = -b\, x_3 + x_1 x_2$$

where s, r, and b are real constants. Compute a solution trajectory for the Lorenz system with ($s = 10, b = \frac{8}{3}, r = 28$), starting point [5,5,30] . ', and time interval [0 40]. Plot the time series solution for $x_1(t)$; it should be chaotic. Plot a phase portrait using

```
>> plot3(x(:,1),x(:,2),x(:,3))
>> axis([-20,20,-20,20,0,60])
>> view(125,20)
```