

## Chapter 7

# Using AUTO: Bifurcation and Continuation

For many people, the main reason to use *XPPAUT* is because it provides a fairly simple interface to the continuation package, AUTO (see Doedel, [9]). AUTO now has a graphical user interface in the distributed version, but still requires that you write FORTRAN code to drive it. *XPPAUT* allows you to use most of the common features of AUTO, including following fixed points, periodic orbits, BVPs, homoclinic orbits, and two-parameter continuations. This chapter will show you how to solve a variety of problems with AUTO. Keep in mind that AUTO is a tricky program and can fail in a dramatic fashion. Nevertheless, it is a powerful tool and, even though it is 15 years old, it is still better than most other similar programs when it comes to tracking periodic solutions and solving BVPs.

Many physical and biological systems include free parameters. One of the goals of applied mathematics is to understand how the behavior of these systems varies as the parameters change. This is a numerically challenging task and there is generally no way to systematically explore a system as a function of all its parameters. One of the first things a modeler should do is reduce the number of parameters either by fixing those which are best known (e.g., gravity) or by making the problem dimensionless. Dimensional analysis is very useful and the reader can consult any number of books which describe this technique (see, for example Edelstein-Keshet [10], and Murray [31]). Assuming that you can reduce the free parameters to a manageable number, there are some useful tools for exploring how a dynamical system changes as these parameters vary. The most straightforward technique is called *continuation* in which a particular solution (such as a fixed point or limit cycle) is followed as the parameter changes. AUTO (which is described in this chapter) provides some very powerful algorithms for the continuation of fixed points and periodic solutions to differential equations. The stability of the particular *branch* of solutions is readily obtained by analyzing the linearization. This, too, is automatically accomplished by AUTO. If a fixed point or limit cycle exhibits a change in its stability, this is often a sign that new qualitatively different behavior could occur. For example, new fixed points might emerge from the branch of solutions, or a limit cycle may emerge. These qualitative changes in the local and global behavior are called *bifurcations* and their detection during continuation is the subject of much mathematical research. AUTO provides a number of tools for the automatic detection of bifurcations of fixed points and limit cycles.

A thorough description of local and global bifurcations can be found in the excellent book by Kuznetsov [25].

Consider a differential equation of the form

$$\frac{dx}{dt} = F(x, \mu), \quad x \in \mathbb{R}^n, \quad (7.1)$$

where  $\mu$  is a parameter. Suppose that (7.1) has a fixed point,  $(x_0, \mu_0)$ . Let  $A(\mu_0) = A_0$  be the linearized matrix for  $F$  with respect to  $x$  about the fixed point  $(x_0, \mu_0)$ . There are two important insights gained by looking at  $A_0$ . First, if  $A_0$  is invertible, then we can follow the fixed point  $x_0$  as a function of the parameter  $\mu$  in some neighborhood of  $\mu_0$ , and this is the *only* fixed point near  $x_0$ . This fact is a consequence of the implicit function theorem. The second point is that if none of the eigenvalues of  $A_0$  lie on the imaginary axis, then the behavior of (7.1) near the fixed point  $x_0$  is the same as the behavior near the origin of the linear differential equation

$$\frac{dy}{dt} = A_0 y. \quad (7.2)$$

In particular, the stability of the two systems is the same. This is why we can determine stability of fixed points by linearizing. However, if one or more eigenvalues of  $A_0$  has a zero real part, then the local behavior of the nonlinear system is not necessarily the same as the linear system. Convince yourself of this fact by looking at the following three systems which all have the same linearization:

$$\begin{aligned} x' &= -y - y^3, & y' &= x, \\ x' &= -y - x^3, & y' &= x, \\ x' &= -y + x^3, & y' &= x. \end{aligned}$$

Thus, if we vary  $\mu$ , and if one or more eigenvalues of  $A(\mu)$  crosses the imaginary axis, we expect that there will be a *qualitative* change in the behavior of (7.1). The analysis of this change in behavior near the fixed point is the goal of *local bifurcation theory*.

Similar points can be made for discrete dynamical systems

$$x_{n+1} = F(x_n, \mu), \quad (7.3)$$

but the critical set for the eigenvalues of  $A(\mu)$  is the unit circle rather than the imaginary axis. Consider a limit cycle solution to (7.1) and take a local Poincaré section. Then, this defines a map and the limit cycle is a fixed point of the map. Thus, the local bifurcations of limit cycles are found by studying the local bifurcations of the associated Poincaré map.

One of the beauties of local bifurcation theory is that the behavior of the systems (7.1) and (7.3) is the same as certain low-dimensional polynomial systems called *normal forms*. We briefly mention the main bifurcations of interest:

- *Bifurcations at a zero eigenvalue for differential equations.* When  $A_0$  has a zero eigenvalue, then new fixed points often arise. These can be transcritical, pitchfork, or saddle-node bifurcations. Only the latter is generic; the other two occur when there

is symmetry. The transcritical and pitchfork are *branch point* bifurcations and AUTO will automatically detect and follow them. The saddle-node, fold, and limit-point bifurcations occur when a branch of solutions bends around.

- *The Hopf bifurcation.* This occurs when  $A_0$  has a pair of imaginary eigenvalues. Generically, a small amplitude limit cycle will emerge from the branch of fixed points.
- *Period-doubling bifurcation.* This occurs for maps when an eigenvalue of  $A_0$  is  $-1$ . For a limit cycle, the result is that the period of the limit cycle doubles.
- *Torus bifurcation.* When the eigenvalues of  $A_0$  for a map cross the unit circle at values other than  $\pm 1$ ,  $\pm i$ , or the cube roots of 1, then a torus bifurcation arises. Chaos and other complex behavior often occur.

AUTO detects all these bifurcations and in some cases can create a two-parameter curve along which the bifurcation occurs.

In the next few sections, we show how to use *XPPAUT* and AUTO together on a variety of differential equations, maps, and BVPs.

## 7.1 Standard examples

As a first example, we consider the normal form for a cusp bifurcation:

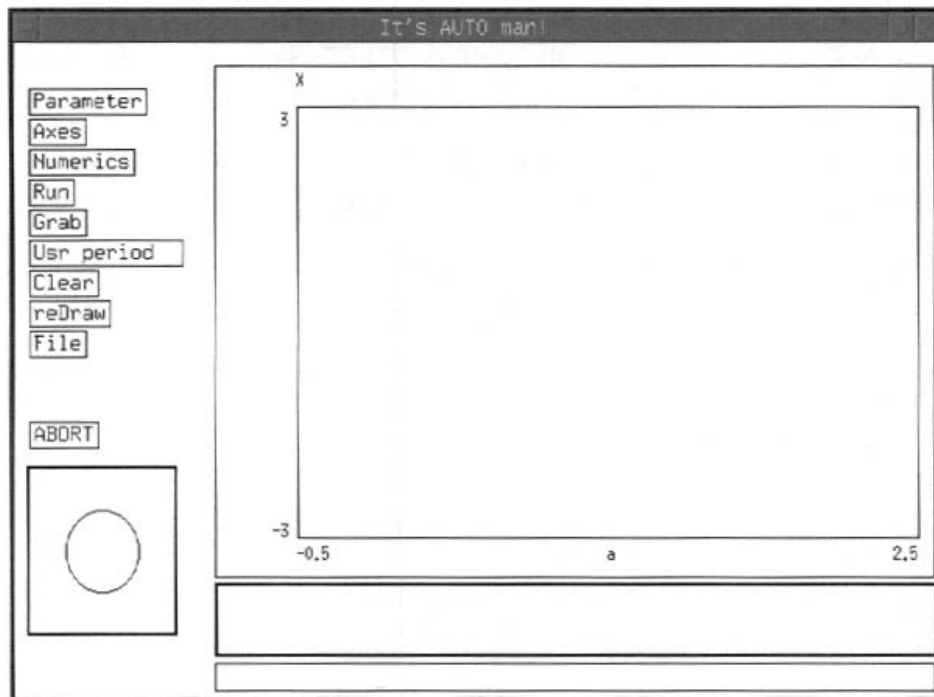
$$x' = a + bx - x^3.$$

Let's pick  $a = b = 1$  and let *XPPAUT* find the root by integrating the equation. Make an ODE file for this problem and run it. Integrate it and then integrate it again using the `Initialconds Last (I L)` command to pick up where you left off. You should have settled onto a fixed point of about 1.3247. Click on `File Auto` to bring up the **AUTO window**. (See Figure 7.1.) In this window, there are additional menu items and several windows. The lower left shows a circle. The number of crosses inside (outside) the circle corresponds to the number of stable (unstable) eigenvalues for a solution. The two windows at the bottom are, respectively, information about the computed points and a hints or tips window. These bottom windows also tell you the coordinates of the main graphics window. The main window is where the relevant bifurcation diagrams are drawn.

Before you can use AUTO, you must prepare your system for it. You must start your bifurcation analysis from either a fixed point of your model, a periodic orbit, or a solution to a BVP. We have already done this for the present example.

Once you have brought up the **AUTO Window**, you should do the following steps:

- Use the `Parameter` item to tell AUTO the list of five or fewer parameters that you will vary for the session. Choose the names of any of your parameters; the default names are the first five that you have defined in the ODE file.
- Use the `Axes` command to tell *XPPAUT* what parameters are varied and what is to be plotted, as well as the range of the graphs.



**Figure 7.1.** *The AUTO window.*

- Use the `Numerics` command to define all the AUTO numerical parameters such as the direction, output options, step size, etc.
- Use the `Run` command to run the bifurcation.

Once you have run a bifurcation, use the `Grab` command and the arrow keys to move around the diagram, and use the `File` command to save things or reset AUTO. The commands `Clear` and `reDraw` just erase and redraw the screen, respectively. The `Usr period` item allows you to tell AUTO to save certain points such as oscillations of a certain period or points when a parameter takes a particular value.

AUTO is essentially independent of *XPPAUT* but there is some communication between them. For example, if you grab a point in the **AUTO Window**, then the state variables are loaded as initial data into *XPPAUT* and the parameters are changed as well. Similarly, when you start fresh in AUTO, the initial fixed point or orbit is computed in *XPPAUT*. Finally, you can import a bifurcation diagram from AUTO into *XPPAUT* and plot this. We will use this feature later to show bursting in a biological model.

With these preliminaries, let's compute the bifurcation diagram for our cusp model. The main parameter is  $a$  and the secondary parameter is  $b$ . Since there are only two parameters, there is no need to invoke the `Parameter` item in the **AUTO Window**. We eventually want to compute a two-parameter bifurcation diagram; to do this, we must first compute a one-parameter diagram. From here on, all commands that I mention concern the items in the **AUTO Window**. Click on `Axes` and choose `HiLo`; this option plots both the maximum and minimum of orbits which, for the present problem, is irrelevant since there are no periodic orbits. In the resulting dialog box, tell AUTO the variable to plot, the main

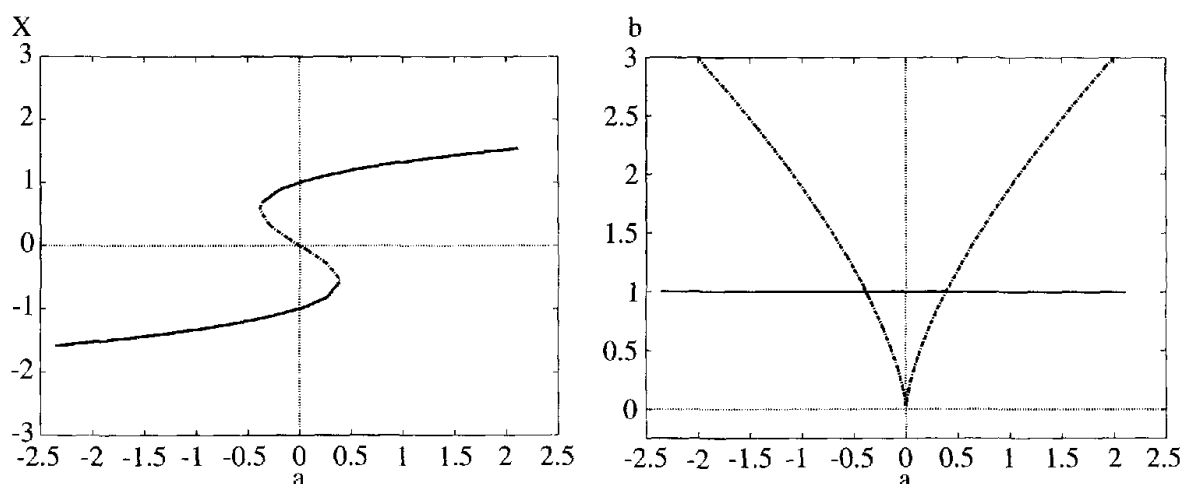
parameter (for one-parameter diagrams) and also the second parameter (for two-parameter diagrams) as well as the dimensions of the plot. Fill it in as follows:

Y-axis: X
Main Parm:: a
2nd Parm:: b
Xmin:: -2.5
Ymin:: -3
Xmax:: 2.5
Ymax:: 3

and then click Ok. We have told AUTO that the main bifurcation parameter is  $a$  and that  $X$  is the variable to plot. The graph will show  $a$  on the horizontal axis and  $x$  on the vertical axis with a range of  $[-2.5, 2.5] \times [-3, 3]$ . Next click on the Numerics item. Change only one item in this dialog box, Par Min, the minimum value of the parameter, which you should change from 0 to  $-2$ . Par Min and Par Max tell AUTO how far to continue solutions. Then click OK. Now you are ready to run. Click on Run Steady state. (This tells AUTO you are continuing along a steady state or fixed point of the system.) A thick black line will appear across your screen toward the right. If nothing happens, you probably forgot to set a good initial guess for the starting point. You should make sure that you are on a fixed point in *XPPAUT*; for  $a = 1, b = 1$  the fixed point is about 1.3247. Assuming that you have a curve, you can see that it is pretty boring and that the curve was produced for  $a$  increasing.

Click on Grab to move around in the bifurcation diagram. You should see a cross on the screen. (If you don't see a cross on the screen, then, next time you run *XPPAUT*, run it with the `-xorfix` option which should fix the lack of a cross. By default it runs without requiring this option on Linux and Windows.) Use the left and right arrow keys to move the cross around. The stability circle shows the eigenvalue status. The fact that the drawn curve is solid implies the fixed point is stable; unstable fixed points are drawn with thin lines. AUTO specially marks certain points with small crosses and numbers. You can jump to these by pressing Tab. As you move around the diagram, the text area beneath the diagram gives you a summary of information about the current location such as the value of the parameter, the state variable, the period, and the point type for special points. There are many point types in AUTO. In this example, you will see only EP which means endpoint. The one you never want to see is MX since this means AUTO has had trouble and maxed out on some sort of numerical iteration. Sometimes this can be rectified by changing the numerical parameters. You can exit from Grab mode by clicking Esc which does **not** grab the point, or by clicking Enter which loads the point into memory. *AUTO can generally be restarted only from special points.*

So, lets make this diagram extend to the left. Grab the first point in the diagram (Grab Enter). Click on Numerics and change Ds from 0.02 to -0.02 and click on OK. This tells AUTO you want to change the direction since the sign of the numerical parameter Ds determines the starting direction for AUTO. Now, click on Run. You should now see a sideways cubic curve. In particular, there are two points labeled 3 and 4 occurring at  $a = \pm 3.84$ . If you click on Grab, you will discover that the two special points are labeled



**Figure 7.2.** One- and two-parameter bifurcation diagrams for the cusp ODE,  $x' = a + bx - x^3$ . The one-parameter diagram has  $b = 1$ . The two-parameter diagram also shows the artifactual line at  $b = 1$  from the one-parameter continuation.

LP, which means that they are limit points, also known as fold points. Note that as you move past them, the eigenvalue goes through the unit circle. The curve of fixed points along the middle branch is unstable.

Grab the leftmost limit point 3. We will now track this fold point as a function of the second parameter,  $b$ , producing a two-parameter bifurcation curve. Click on **Axis** and select **Two param**. Change **Ymin** from  $-3$  to  $-0.5$ . Since the plot type is **Two Par** the vertical axis contains the secondary parameter ( $b$ ) and the horizontal axis contains the main parameter ( $a$ ). Click on **OK**. Click on **Run**. You should see a thin curve that rises and moves to the left. This is a curve of fold points; along this curve in  $(a, b)$  space, the original system has a fold.

Once again, invoke the **Grab** item and click on **Tab** until you have selected the limit point 3 again. (Look at the bottom of the **AUTO Window** to see which points you move through.) We now want to change the direction of bifurcation to get the rest of the curve. Click on **Numerics** and change **Ds** from  $-0.02$  to  $0.02$ , thus forcing AUTO to make the main parameter increase. Click on **Run** and you will get the rest of the curve; it is a nice cusp. For each  $(a, b)$  inside the cusp, there are three fixed points and, for  $(a, b)$  outside the cusp, there is only one fixed point.

Click on **Axes Last 1 Par** and then on **reDraw**. You will get the one-parameter diagram again. Click on **Axes Last 2 par reDraw** and you will see the cusp. This is a shortcut for selecting the most recently defined one- and two-parameter views. Click on **File Postscript** and you can make a PostScript version of the diagram. Figure 7.2 shows the one-parameter and two-parameter diagrams.

### 7.1.1 A limit cycle

We will start on a limit cycle in this example. The limit cycle is essentially what is called an *isola* since it is not connected to a main branch of fixed points. Here are the equations:

$$x' = xf(R) - y, \quad y' = yf(R) + x,$$