DsTool:
A Dynamical System Toolkit with an Interactive Graphical Interface

Based on the program **kaos** written by S. Kim and J. Guckenheimer

Tutorial

Center For Applied Mathematics
Cornell University, Ithaca NY 14853

May 24, 1995

Contents

1 Tutorial **4**

1.1 Preliminaries 4
1.2 The Lorenz Equations 6
1.2.1 Conventions 6
1.2.2 Opening a View Window 7
1.2.3 How to Display Orbits 9
1.2.4 Using a Mouse to Generate Orbits 9
1.2.5 Reading Coordinate Values 10
1.2.6 Changing the Plane of Projection 10
1.2.7 Changing the Scale 11
1.2.8 Other Mouse Button Functions in View Windows . . . 13
1.2.9 Quitting and Interrupting 14
1.2.10 Orbits Panel 14
1.2.11 How to Change Parameters 15
1.2.12 Loading Other Dynamical Systems and Restoring Defaults 17
1.2.13 Computing Equilibria 18
1.2.14 Computing Manifolds 19
1.2.15 Multiple Orbits 21
1.2.16 How to Print 23
1.2.17 Use of Color 23
1.3 A Torus Diffeomorphism 24
1.3.1 Choosing an Installed Dynamical System 25
1.3.2 Orbits of Maps 25
1.3.3 More on Fixed Points 25
1.3.4 More on Manifolds 26

1.3.5 Saving and Loading Data 29

List of Figures

1.1 The Command Window. 6
1.2 A 2-D Image Window 8
1.3 Orbits projected on the xy-plane 9
1.4 A different projection: the xz-plane 11
1.5 Zooming Out 12
1.6 Zooming In 13
1.7 The Orbits Window 14
1.8 The Selected Point Window. 16
1.9 Similar initial conditions with different fates 17
1.10 The Fixed Point Window. 18
1.11 Equilibria Found 19
1.12 The Full Fixed Point Panel. 20
1.13 Stable and Unstable Manifolds 21
1.14 The Multiple Orbits Window. 22
1.15 Evolution of an xy plane rectangle centered near the origin. . 23
1.16 Using Multiple Windows 24
1.17 Orbits of a torus diffeomorphism 26
1.18 Stable and Unstable Manifolds on the Torus 27
1.19 A homoclinic tangle 29
1.20 A phase portrait on the torus 30

Chapter 1

Tutorial

1.1 Preliminaries

This chapter is a tutorial for **DsTool**. Its purpose is to introduce the user to **DsTool** quickly. It is suggested that the user read this tutorial while actually running **DsTool**. The tutorial consists of three sections: this brief overview and two examples. In the first example, we will use **DsTool** to explore the Lorenz system; in the second, we will explore maps of the two-dimensional torus. We describe only basic features of **DsTool**. This is not a comprehensive description of the capabilities of **DsTool**. More complete documentation is presented in the **DsTool Reference Manual** [10] and the **DsTool User's Manual** [9]. In particular the **DsTool User's Manual** explains how users can easily install their own dynamical systems in **DsTool**.

Running windowed working environments such as `X11R4` (Release 4 of `X11`), `X11R5`, or `Open Windows` on a workstation generally involves invoking both a **server** and a **window manager**. In this context, a **server** is a program controlling one or more displays. Client programs such as **DsTool** make `X protocol` requests to the server, and it is up to the server to actually accomodate these requests. A **window manager** is another client program, which is given special authority to make decisions (using user and client preferences) about issues such as window placement and user interaction style. Due to the relatively low level of abstraction of some `X11 services`, most applications, including **DsTool** are built using an additional toolkit library layered on top of `X11`.

In the case of **DsTool** the additional toolkit library is the Sun developed `xview`. Sun donated the source for this library to the X Consortium, but compiling it on other platforms has been nontrivial. **DsTool** has been ported to most environments where `xview` is available. At the time of **Version 2.0**, installation was tested on Sun's running **SunOS 4.1.x** and **SunOS 5.4**. We used a binary distribution of **xview** under **Irix 5** on an Indigo 2 to successfully compile. It was also possible to test an earlier form of this version on a 386 PC running **Linux 1.1.59**. Earlier versions of **DsTool** ran on Decstations, RS 6000's, and some HP machines, but we are uncertain about the availability of suitable xview ports for current versions of their operating systems.

We will assume that the program has already been installed either by yourself or by a system manager. If **DsTool** has not already been installed, please refer to the **DsTool** installation instructions.

DsTool uses three environment variables which will typically be set in a users `.cshrc` file. The variable **DsTool** should point to the root of the local **DsTool** distribution - possibly a directory such as `/usr/local/dstool`. With this choice, the users `.cshrc` file should contain the line "setenv DSTOOL /usr/local/dstool". If you are unsure of this location, consult your system manager. The environment variable DSTOOL_COLOR_DIR gives the location of colormaps used by the system, when color is available. This directory will often be $DSTOOL/colormaps. The variable DSTOOL_DATA_DIR specifies a default location for data files to be saved to and loaded from.

Before accessing **DsTool**, start up your window system and window manager in the usual way. On a Sun running `Open Windows`, this might be done by using the script `openwin`. This will call `xinit`, and in the course of processing (or possibly creating) your `.xinitrc` file, the default window manager `olwm` will be started.

DsTool is controlled by a collection of panels in separate windows. Windows contain buttons, menus, text fields, and drawing canvases.

To access **DsTool** start the program by typing **dstool**. If the **dstool** executable has been correctly installed and lies in a directory mentioned in your `path`, the **DsTool command** window (*figure 1.1*) pops up.

The top line of the window is a header containing a title such as **DsTool (Version 1.0)**. On the left of the header line is a small button with an inverted triangle ($\bigtriangledown$) — this is an `Open Look` convention for an `abbreviated`

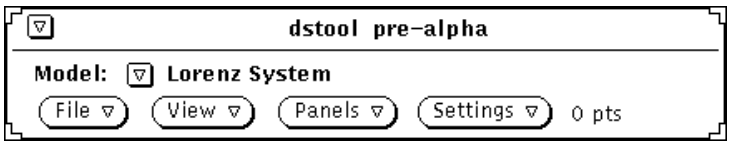

Figure 1.1: The Command Window.

`menu button`. Clicking and holding the MENU button of your mouse (usually the rightmost) over this button will display a menu whose bottom entry is **Quit** — selecting this is the easiest way to exit **DsTool** at any time.

Below the header of the **command** window is a control panel. The first line controls the selected model and the second line is a series of menu buttons. By default, **DsTool** comes up with the Lorenz system selected. You can use the abbreviated menu button between **Model:** and **Lorenz System** to choose other pre-installed dynamical systems. We'll see how to do so later.

1.2 The Lorenz Equations

In this section, we shall learn some of the basic **DsTool** operations by examining the Lorenz equations. The Lorenz equations are given by

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy \end{aligned}$$

These equations represent a truncation of a partial differential equation which models two-dimensional fluid convection. This system was first studied by Lorenz in the early 1960's. The reader is directed to [1], [2], or [3] for more information regarding the nature and behavior of this system.

1.2.1 Conventions

Throughout the tutorial, mouse operations are frequently used. In order to facilitate our subsequent discussion, we introduce the following conven-

tions. See the `Open Windows Users Guide` [7] for more information on `Open Windows` interaction conventions.

1. The three mouse buttons on the Sun mouse will be denoted SELECT, ADJUST, and MENU. By default, these are the left, middle , and right mouse buttons respectively.

2. *Shift-clicking* and *control-clicking* are defined as pressing a mouse button while **Shift** or **Control** are also depressed. We will use notations like Shift-MENU and Control-MENU to indicate these.

3. Buttons are activated by clicking SELECT on them.

4. Menus generally are opened by pressing MENU. Items on pop-up menus are selected by holding this mouse button while the pointer is dragged to the menu entry being chosen.

5. Text fields are edited by first clicking SELECT within the appropriate text field, and then using keyboard keys including **Delete** to change the entries.

6. One of the features of the `Open Look` environment is that menus may be opened for display without continuing to hold the mouse button. Clicking (without holding) MENU on a menu button is the procedure for doing this. One can then select a menu entry by clicking MENU or SELECT on it. The menu can also be dismissed by clicking MENU again outside the menu.

7. Windows are moved by clicking and holding SELECT on the window header while moving the pointer. When the window is in a satisfactory location, simply release SELECT.

1.2.2 Opening a View Window

The first thing to do after starting **DsTool** is to open a view window. Open the **view** menu on the **command** window and using MENU, choose the **2-D Image** entry. A window with a title such as **dstool:Lorenz System** (*figure 1.2*) appears. It may well obstruct some other important window – using SELECT on the header, drag the window to a convenient location. The

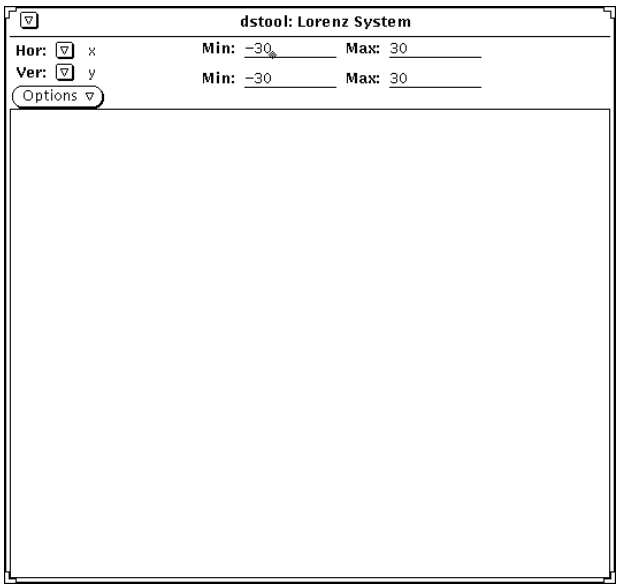

Figure 1.2: A 2-D Image Window

view window contains abbreviated menu buttons for selecting the horizontal and vertical coordinates being displayed, as well as text fields controlling the coordinate ranges of the viewport. (These coordinate ranges are expressed in the original world coordinates of the dynamical system.) Coordinates displayed in view windows can be phase space coordinates of the original problem, auxiliary functions, or parameters built into the dynamical system at creation. For the Lorenz system, the default display is to project the phase space coordinates (x, y, z) to (x, y) as horizontal and vertical coordinates of the view window.

Move the mouse within the canvas below the view window control panel. The canvas is the large boxed-in area where graphics are displayed. Note that the horizontal and vertical coordinates of the mouse position (in world coordinates) appear to the right of the **options** menu button.

Within the limits of system resources, you can open as many view windows as you like. So it's possible to view a number of two dimensional projections simultaneously. By default, the same orbit in different windows will be drawn in the same color. Different orbits will generally be drawn in different colors. The use of color is controllable as described in Section 1.2.17.

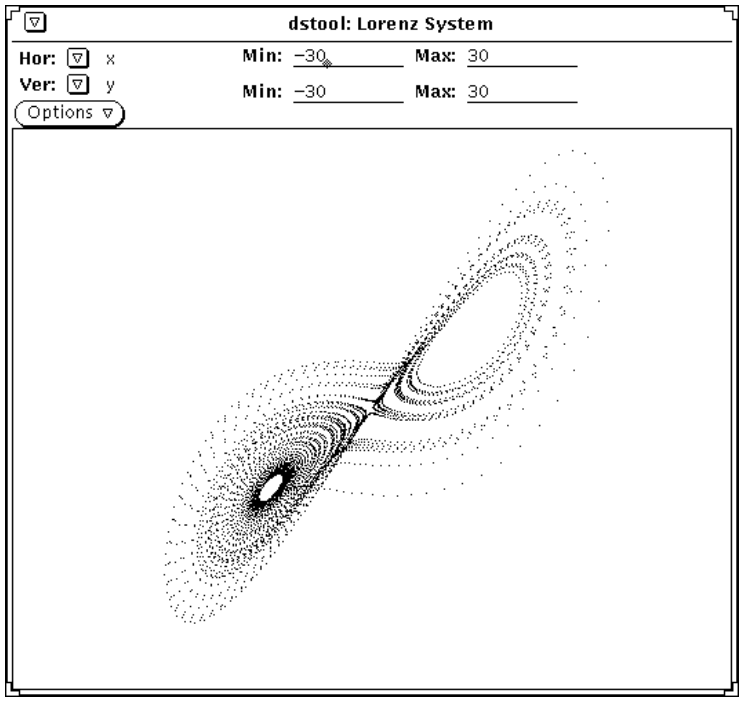

Figure 1.3: Orbits projected on the xy-plane

1.2.3 How to Display Orbits

1.2.4 Using a Mouse to Generate Orbits

Move your mouse into the view window and click SELECT. **DsTool** will remember this selected point until you explicitly choose another. Now click ADJUST in the view window. *Figure 1.3* shows a typical result. By default, **DsTool** will compute 5000 points forward along the trajectory starting at the previously selected point. The points will be equally spaced at time intervals of .01. We'll see shortly how to change these parameters. To the right of the **Settings** button on the **command** window, you'll see the number of points currently stored by **DsTool**. Each time you click ADJUST another 5000 points are computed and displayed.

If you're using a color monitor, all points of this orbit will be drawn in the first color of **DsTool's** palette. Now click SELECT at another location in the canvas of the view window. Clicking ADJUST will plot 5000 points

on the orbit from the newly selected point. On a color monitor, this orbit will automatically be displayed in the next color of **DsTool's** palette.

You can choose initial conditions for backwards orbits by clicking MENU on a location in the view window canvas. This also resets the current orbit direction from forward to backward. Now clicking ADJUST within the canvas will cause up to 5000 points along the backward orbit to be displayed. Your backward orbit will probably rapidly go off to infinity, so **DsTool** will display a message and stop computing somewhat earlier. On a color monitor, this backward orbit will automatically have been displayed in the next color of **DsTool's** palette.

DsTool only remembers one selected point and one orbit direction. So pressing SELECT or MENU within the canvas both sets the orbit direction and replaces any previously selected point. One of the menu entries under the **Options** menu button of the **2-D Image** window is **Display...** which will allow you to mark this selected point before you change it.

1.2.5 Reading Coordinate Values

Numerical values of phase space variables can also be inspected and adjusted directly. Going back to the **Settings** menu button in the **command** window, click the MENU mouse button on the entry **Selected....** The **Selected Point** panel which appears will then contain text fields with the coordinates of the last point you chose with SELECT or MENU as well as the last point computed on the current orbit. You can directly edit the values of the selected point (in the column labeled **Initial**) in these text fields. Or click on the **Copy Final to Initial** menu button to replace the selected point by the endpoint of the current orbit.

1.2.6 Changing the Plane of Projection

Those familiar with the Lorenz system may have noticed that the orbits which appear here do not look like the phase portrait normally associated with the Lorenz attractor. This is because **DsTool** is projecting this three dimensional system onto the xy-plane. To change the plane of projection, locate the abbreviated menu button labeled **Ver:** on the view window control panel. Open this menu and select the variable **z** to replace **y** as the vertical coordinate. Then open the **Options** menu button just below and select the

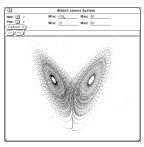

Figure 1.4: A different projection: the xz-plane

Refresh menu entry. The screen will redraw each orbit as projected into the xz-plane. *(See figure 1.4).*

In the same way as above, values of auxiliary user-defined functions can also be displayed in **2-D Image** windows. Or **time** may be chosen as one of the displayed coordinates to generate *time series.* The **DsTool Users Manual** explains how to add user-defined functions to a particular **DsTool** system.

1.2.7 Changing the Scale

To see fine structure, it is often desirable to zoom in on a particular portion of the phase space. Conversely, sometimes a larger phase space perspective is desired. Changing the scale of the view window can be accomplished by editing the **Min:** and **Max:** text fields to the right of the horizontal and vertical variables. Change the minimum `z` value to `-100` instead of `-5` and change the maximum to `100`. Be sure and enter a **Carriage–Return** after you edit each text field — then select **Refresh** from the **Options** menu button and the orbits will appear compressed and much further away. *(Figure 1.5)* This process can also be used to zoom in, or to view any rectangular portion of the xz-plane.

It is also possible to rescale using a "rubber-band" method. This is often preferred for zooming in, whereas the method described above is often preferred for zooming out. Position the mouse pointer in the view window near the upper left corner of the Lorenz attractor. Press Shift-SELECT and keep the SELECT button depressed. (You may release the **Shift** key.) Now drag down and to the right until the attractor is enclosed in a rectangle called the **rescaling box**. When the attractor is satisfactorily enclosed within the rescaling box, release SELECT. A notifier box offering the options of **zoom in**, **zoom out**, or **cancel** will pop up. Click SELECT on the button **zoom in**. The attractor will be displayed larger than it was before. *(Figure 1.6)*

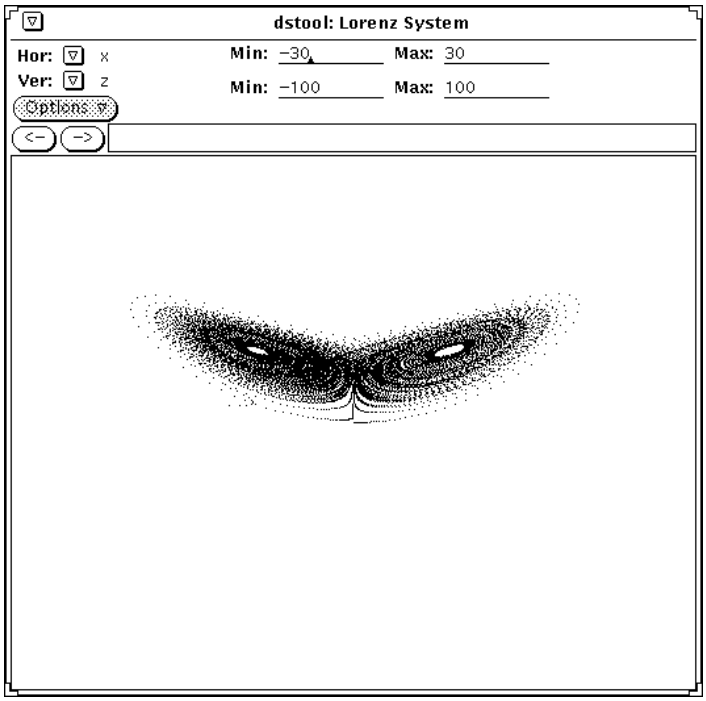

Figure 1.5: Zooming Out