Using Math Software Under UNIX

Clinton Wolfe (April 1999)
with material from Dave Hart

Introduction

Purpose

Distributed computing puts into practice a favorite Hoosier proverb: "Use the right tool for the job." As distributed computing continues to grow, the network is becoming one vast virtual machine, with different parts performing different functions. UITS provides a variety of hardware platforms, from large shared computers to desktop workstations in Student Technology classrooms and labs. Workstations typically provide superior graphics and a more convenient user interface, and offer powerful logical and arithmetic processing capabilities, but generally lack the large memory that shared computers provide. On a shared computer, you can use Maple to manipulate formulas thousands of lines long, or use Matlab to perform billions of computations -- all within minutes. Using workstations in conjunction with multiuser systems, via the network, can combine the best functionality of both.

Who this document will help

This document is intended for users of Maple, Matlab, and Mathematica who need the computing power of very large computers. This is *not* a tutorial for any of these software packages; instead see Getting Started with Maple, Matlab, or Mathematica. We assume that you are fairly familiar with one of these packages. We also assume you are comfortable using the basic functionality of network utilities such as FTP and Telnet (or SSH). Our goal is to ease the transition from a window-style desktop environment to a command-line based UNIX environment.

The standard math tools at IUB -- Maple, Mathematica, and Matlab -- have versions that can run on many different platforms. The microcomputer version offers a relatively friendly interface, while the mainframe version offers raw power. For everyday problems, you'll find the microcomputer version fast and easy to use. Even when you need the power of the mainframe, you may still want to develop algorithms and debug code on your desktop, then turn to shared systems for production runs that require large storage or extended number-crunching.

This document is designed for users at Indiana University (Bloomington or IUPUI campuses), who have accounts on either the SP system, Steel, Cobalt, Nations, DaVinci, or Ships. The information contained herein may not be appropriate for your specific circumstances (especially when refering to characteristics of specific computers).

How to use this document

Everyone should read the UNIX - The Least You Need to Know and Starting the Math Software sections, unless you already know this information. You may then pick among the remaining sections to fit your needs.

Please send errors, omissions, comments, etc. to statmath@indiana.edu, or post to iu.statmath.

UNIX - The absolute least you need to know

UNIX is an operating system (like Windows, DOS, or Macintosh System 8) that is designed for very large computers. More than one person (in fact, ususally hundreds!) can use a UNIX machine at the same time.

Accounts

Every person who uses a shared computer has an *account*. An account means that you have a place to put files (your "home directory"), permission to use the machine (to "log in"), and a way to identify yourself (your "username"). For example, the Shakespeare email system at IUB is a UNIX system.

You will need an account on one of the following systems: DaVinci, Ships, Nations, the SP (formerly STARRS), Cobalt, or Steel to use this guide. If you don't have one, you may get one from The UITS Accounts System. Or, if you have your own UNIX machine (such as a Sun, Linux, FreeBSD, etc.) and you have a copy of the math software, you may use your own machine.

Choosing a machine

As of now (August 1999), if you want to run math software on a large computer at IUB, you have the option of using the SP System , Cobalt, or Steel.
If you just want to run math software under UNIX, you could also use the DaVinci, Nations or Ships cluster, or your own machine.

You must be faculty (or have a faculty sponsor) to get an account on the SP, since it is a research-only system. In addition, the SP is really only designed for batch processing (which we discuss in the Batch Processing section), and logging in is a little complex. More information on the SP is availible here.

Steel is open to all IUB faculty and students, as well as some members of the IUPUI community. Its is a "general purpose" system; in other words, you aren't required to do batch processing. If you'd like to know Steel's specifications, you can find it here.

The DaVinci cluster is a cluster of SGI O^2 workstations located in FA 215. While not designed for heavy computational use, you can still use the DaVinci cluster to get a feel for using Mathematica under UNIX. To connect to any DaVinci machine, telnet to `davinci.ucs.indiana.edu.` All IUB faculty, staff and students may apply for an account on the DaVinci cluster.

To simplify this document, we assume you will be using Steel unless otherwise noted.

Keep in mind that you will be running the math software on the machine you connect to, *not* the machine you are sitting at.

Logging in

Since you are (presumably) not in the same physical room as Steel, you will need to log on using Telnet. The address is
`steel.ucs.indiana.edu` .

- From a UITS NT cluster, go to Start Menu - Programs - Communications - Other WCC Computers - Steel.
- From a UITS Mac cluster, go to Apple Menu -Telnet-Steel.

Enter your username and password. If this is your first time logging in, you may be presented with some information about the system. You should eventually see the UNIX prompt, which should be a $ or a %.

Learning more about UNIX

At this point, you should have a UNIX prompt in front of you. This is really all you need to get the basic functionality of the math software.

However, your UNIX experience will be much more satisfactory if you follow an in-depth tutorial. We recommend:

- Research Computing at IU: The Basics. A comprehensive guide to using research computers at IU. It is IU-specific, but not campus specific.
- UnixTools.com. A good web directory of UNIX help. Includes history, flavor-specific help, and tutorials.

Starting the Math Software

Modes of Use

Before you use the math software, you should be aware of the different ways of using it. On a desktop computer, it is only possible to use it in a fully interactive setting. On UNIX, it may be preferable to use it in a different way. There are three ways of running most software under UNIX:

- Interactive mode. You type in input directly, and output is displayed on your screen. The interface may be a full window with pretty graphics, or it could be a terminal-based system.
- Non-interactive mode. Over a terminal, you can put all of your input into one file, then have the software process the commands and save the output to a file. You can be doing other tasks while it runs. Graphics output is limited to saving plots as images.
- Batch mode. You create an input file as for non-interactive mode, but you also create a "job script", which tells the system software to execute the math software (with your input) at a later time. Jobs are queued, so you may have to wait a while for your results. Batch jobs allow you access to more memory and processor power than the other modes, so it is best for extensive number crunching.

At IU, most batch jobs should be done on the SP system. We'll discuss batch mode later, in the Batch Processing Section. In addition, we'll discuss only terminal-based modes here, and we'll cover interactive graphics later, in the X Windows Section.

Maple

Interactive Mode
To start maple, just type `maple` at the UNIX prompt. Maple will start, and you will see the familiar "" Maple prompt.
To leave Maple, type `quit;` at the Maple prompt.
All of the standard packages are available, and you should find that Maple behaves just like it does on your desktop.

Non-interactive Mode
You can load plain text files with maple commands (one per line, ending in a semicolon - the usual). Be sure the last command is `quit;`. Then type at the UNIX prompt:

```
maple filename results.txt
```

This will return you to the UNIX prompt while Maple executes the commands in `filename` and saves the results in `results.txt` .
To learn more about Maple's file input and output abilities, run Maple interactively and type the following:

```
?writeto
?echo
```

```
?save
?read
?interface
```

Matlab

Interactive Mode
To start Matlab, type `matlab`. Matlab will start, and you will see the familiar "" prompt.
Type `quit` at the Matlab prompt to exit.
IUB has licenses to most of the Matlab toolboxes. Use them as you would at your desktop. Be forewarned that most graphics functions simply won't do anything if your terminal doesn't support graphics. Matlab always looks for m-files in the directory from which it started (as well as the standard library locations).

Non-interactive Mode
Put all of your commands in a plain text file as if you had typed them into Matlab. Be sure the last command is `quit`. Then type

```
matlab results.txt &
```

This runs Matlab, executes the commands in `filename` and saves the results to `results.txt` , while you are returned to the UNIX prompt.
To load and save Matlab variables, type the following at the Matlab prompt:

```
help load
help save
help diary
```

Mathematica

Interactive Mode
To start Mathematica, type `math`. You should see the familiar Mathematica prompt, "In[1]=". To exit Mathematica, type `Quit`.

This actually starts the Mathematica Kernel and lets you type directly into it. There is no terminal-based front-end, since the kernel itself can function in this manner. However, you will have primitive graphics and plotting capabilities (but you will not have the ability to export to an image file). **Non-interactive Mode**
Save all your commands, one per line, in a file. Be sure the last command is `Quit`. Then type

```
math results.txt &
```

This runs Mathematica, executes the commands in `filename` and saves the results to `results.txt` , while you are returned to the UNIX prompt.

For more information on saving and loading work, type at the Mathematica prompt:

```
?Save
?
?<<
```

Using Graphics Over the Network

Advice on graphics

The goal of this section is to help you use the graphics features of the math packages. There is no perfect way of doing this. Before you attempt any method other than typewriter graphics, you should ask yourself if you really need graphics output from your UNIX session. UNIX machines are designed to crunch a lot of numbers quickly,

but they are *not* designed to draw pretty pictures. Desktop versions of the math software *are* designed to draw pretty pictures.

One good way out of this is to use the UNIX machine to do all the hard work, save the results in a file, transfer the file to your desktop machine, and then use your desktop package to visualize and interpret the results.

This area of this document is still changing as we perfect the methods. If you have any problems, suggestions, etc. with the techniques used here, please post to iu.statmath, or email statmath@indiana.edu.

Typewriter Graphics

Typewriter graphics is the default behavior for all three packages (but if you are running an X-server and have set the DISPLAY variable, all three default to X graphical output). Typewriter graphics simply means that the program will try to draw a picture using ASCII characters. While not pretty (or accurate, for that matter), typewriter graphics are good for drawing formulae with a bit more clarity than straight text.

Postscript, VRML, and GIF Output

All three major math packages offer graphics output to other file formats. This may be your best graphics option if you plan to publish your results on the web, use graphics in other applications (such as word processors), or if you are doing batch processing. All three packages will place output files in the current directory if a full path is not given, so make sure that you have write permission on the current directory. After you are done, you will need to retrieve your files using FTP.

Maple

Maple provides good export capability. See
`?plottools,vrml` - export 2d or 3d graphics to vrml
`?plot,device` - export to gif, jpeg, Postscript, and other formats.
`?plot,device,ps` - options for Postscript output.
`?plotsetup` - output options - especially see section on `plotoutput` variable.

Matlab

Matlab provides extensive export capability. Type the following in Matlab for more information:
`help vrml` -export a graphics object to VRML 2.0
`help imwrite` -write out a graphics object to a wide variety of formats
`help saveas` -save a graphics object to Postscript
`help print` -save a graphics object to Postscript

Mathematica

Graphics output requires the use of a front-end. Since the only front-end availible for UNIX is the X-windows client, this method is less useful (since you would already have graphics capability directly from within Mathematica). However, you can still redirect output to other file formats. You can use the following command to redirect plot output to a GIF file.

```
    Plot3d[[Sin[x y],{x,0,4},{y,0,4}, DisplayFunction-
Function[Display["foo.gif",#,"GIF"]]
```

You use this option with all plot, play, and show commands. Be sure that your current working directory is something reasonable, since that is where the file will end up. You may also use a full path for the filename. Read about other file formats by using the Mathematica help browser, under "Built-in Functions": Graphics and Sound - Graphics Exporting - Display.

X Windows

X Windows is a distributed Graphical User Interface (GUI) environment. In other words, you can run a program on a UNIX machine, but have the window appear on your local machine. This gives you the full functionality of using the graphical interface, while still having the power of a large UNIX computer. The program you are running is called an X-windows *client*. You local machine must be able to interpret and display the windowing commands that your client sends - in other words, your local machine must act as an X-windows *server*. Most Solaris, Linux, and IRIX (SGI) workstations are capable of acting in this manner directly. For Mac and Windows users, you must run a seperate program that will provide this capability. Please note that using X-windows requires a fast network connection, so using this option over a dial-up line is not recommended.

- For Mac users, you will have to install a third-party X-windows server (this is different from the Mac OS X Unix Server). UITS does not support or endorse the use of these programs in the public clusters (STCs). However, you may purchase, install, and use these programs on your own Mac. MI/X is a free X-server for the Mac. UITS provides this link as a service to customers and in no way endorses or supports this product.

- For Windows (3.x, 9x, and NT) users, you will have to install a third party X-windows server. UITS does not support or endorse the use of these programs in the public clusters (STCs). However, you may purchase, install, and use these programs on your own PC. Popular X-windows servers include X-WinPro , Exceed, and MI/X (free). UITS provides these links as a service to customers and in no way endorses or supports these products.

- For UNIX workstation users, the Ships and DaVinci machines have X already installed.

To use math software as an X-windows client, follow these steps.

Setup the X-windows server on your local machine.

If you are using a UNIX local machine, you will have to give the client machine permission to connect to your X-server. Do this by issuing the command `xhost +steel.ucs.indiana.edu`

PC and Mac users should start their X-server software. Refer to the documentation of the software.

Connect to the remote machine using telnet.

You need to tell the remote machine where you are. To do this, you must set the DISPLAY environment variable to your local hostname.

Find out what shell you use by typing `echo $SHELL`

If you use csh, type `setenv DISPLAY hostname:0`

If you use bash, type `DISPLAY=hostname:0;export DISPLAY`

Start the X-client version of the math software from your telnet session.

Maple - issue the command `xmaple &`

Matlab - issue the command `matlab`. If your X-server is setup properly, you should be able to enter matlab commands in the telnet window, but have graphics displayed in a "pretty" new window. Type `demo` to try this out.

Mathematica - issue the command `mathematica &`. You may see errors about fonts. If so, follow these instructions.

Do your work, and exit from the math software.

Un-setup your X-server

For UNIX machines, make sure you issue the command `xhost -steel.ucs.indiana.edu` after you are done.

For PC and Mac users, consult your X-server's documentation.

Please note that using the X-windows system may open security holes on your system (more info).
More information on the X-windows system may be found at www.x11.org.

Mathlink

Note: This information applies to the Mac edition of Mathematica 4.0.

Mathlink is a feature of Mathematica. Users familiar with Mathematica will recall that it is made up of two parts:

- The Kernel - does all processing, manages variables, and loads packages
- The Front-end - does all graphics work, input, and output.

Mathlink is a proprietary protocol that allows the kernel and front-end to run on seperate machines. (It also allows compiled programs to access a remote kernel - but that topic is well outside the scope of this document). To use Mathlink, do the following steps:

- Go to the Kernel menu and select Kernel Configuration Options,
- Select Kernels and Tasks,
- Click the Add button to add a new kernel
- Assign the new kernel a name. For example, if using Mathematica on DaVinci, you may want to use DaVinci as the name of the new kernel.
- Under Basic Options, click on Remote Machine.
- Click the Connections button and under Connection Settings Method, select VersaTerm Telnet Tool.
- In the Host TCP/IP Address box, type in either the hostname or IP address of the remote host and click OK.
- If connecting to a Unix system, click on the Terminal button and select VT102 under Terminal Settings Emulation. You may also customize your keyboard mapping here.
- Note, by default Mathematica looks for the symbol before launching the program on a remote system. If the default prompt on the remote system is not , then you must either change the default prompt or the start up option in the new kernel settings. To change the prompt locally, click on Advance Options and in the Communications Toolbox Login box, change it to whatever the last character in your login prompt is. For example, if you were accessing Mathematica on a Unix system with the default prompt for Korn shell, you would change to **$**.
- Once all the settings are correct, click OK twice.
- In the Kernel menu, choose Notebooks Kernel and select the Connection from the list of available kernels.
- Again from the Kernel menu, choose Start Kernel and select the Connection from the list of available kernels.
  You will then be asked to supply a username and password for the remote system.

Using Batch Processing

Using the SP

Batch processing is an extremely powerful way to do computationally-intensive work. The main system for doing batch processing at IU (for research purposes) is the IBM RS/6000 Scalable POWERParallel System, known as the SP.

The SP is made up of 47 different nodes. Maple and Matlab are available on sp09, sp10, and sp11. Mathematica is only available on sp09. The first time you log in to the SP, log into `libra.ucs.indiana.edu`. Libra manages the SP's password records, and will initialize your account on all the nodes. Then you may telnet or SSH to `spXX.ucs.indiana.edu` (replace XX with the node number you want). If you're doing batch processing, you can log into any node and (assuming you follow the instructions) the job will execute on the right node.

Please note that the SP is intended for use only as a batch processing system. All processes that accumulate more than 20 minutes of CPU time in an interactive (non-batch) setting will be terminated.

Preparing to batch process

If this is your first time batch processing, we recommend you first attempt a "smaller" version of your task. Don't try to solve your research problem on the first attempt.

Create a text file that has the exact command(s) you wish to run. This file will be fed into your math package as if you typed it in. In other words, each line in the text file should be the exact same thing as what you would type at the Maple, Matlab, or Mathematica prompt. You are free to have your text file load other files or worksheets, but be sure to use the *full path* when specifying filenames. You can find out the full path by typing `pwd` . This tells you the "present working directory". For example, if you want to load a file in your home directory named "work.mws", you should give the filename as "/N/u/john_smith/work.mws". The same goes for all output commands.

Remember, you won't be able to interact with the math software while it is running. This means you need to save the results of your work in some way. You should probably use the math package's built-in save function. For graphical output, we recommend outputting to a file format such as VRML, GIF, or JPEG.

Also, you *must* test your plain text file before executing it on the SP. We recommend testing it on Steel. If you saved your text file as "foo.txt", you can test it using the command `maple foo.txt foo.out` (or matlab, or math). This feeds foo.txt to Maple one line at a time, and any output that would normally go to the screen is redirected to a new file called foo.out. Make sure that your list of commands does exactly what you think it will.

After you have perfected your math package command script, you are ready to batch proccess it.

Writing a LoadLeveler Script

The SP uses a program called LoadLeveller to do batch processing. You wrap up all the commands neccesary to do your job in a file called a LoadLeveller Script.
A good tutorial on using LoadLeveller is The UITS Research SP System - Submitting Batch Jobs .

Sample scripts may be found in the `/N/u/statmath/SP/scripts` directory on the SP.

Further Reading

For general questions, including questions about various platforms and their terminal and X window software, call the UITS Support Center at 855-6789, or use the WWW to browse the UITS Knowledge Base.

More information on the X Window System is available from the Unix Workstation Support Group. If you have questions about statistical or mathematical software, call the UITS Stat/Math Center at 855-4724, or send e-mail to statmath@indiana.edu.
