Introduction to SCILAB

By

Gilberto E. Urroz, Ph.D., P.E.

Distributed by

*i*nfoClearinghouse.com

Introduction to SCILAB

This chapter is intended to get the user started using SCILAB through simple exercises in numerical calculations. The chapter starts by describing how to download and install SCILAB in a Windows environment. Installation of the software in other operating systems is very similar and is explained in detail in the SCILAB website.

What is SCILAB?
SCILAB is a numerical, programming and graphics environment available for free from the French *Government's "Institut Nationale de Recherche en Informatique et en Automatique - INRIA* (National Institute for Informatics and Automation Research)." It is similar in operation to MATLAB and other existing numerical/graphic environments, and it can be run using a variety of operating systems including UNIX, Windows, Linux, etc.

SCILAB is a self-contained package including a large number of intrinsic numeric, programming and graphics functions. Once unpacked and installed in your computer it will consume about 50 MB of your hard disk. Make sure you have at least that much memory in your hard disk before downloading and installing SCILAB.

Where to find SCILAB
SCILAB is available for free from the SCILAB web page: http://www-rocq.inria.fr/SCILAB/
Once, you have accessed this web page, follow the procedure described below to download and install SCILAB in your computer.

How to download the software
Click on the link labeled **Download Scilab 2.6** . This will take you to an information page. Next, click on the link labeled ftp.inria.fr in directory. This will send you to the ftp download page ftp://ftp.inria.fr/INRIA/Projects/Meta2/Scilab/distributions/. From that directory you need to click on the link Scilab26.exe,or on the link Scilab26.zip if you prefer a zip file containing the installation program. Save the program it in the directory of your choice.

How to install SCILAB
Unpack *SCILAB26.zip,* if you downloaded that version of the installation program. Run the program *SCILAB26.exe* and follow the instructions in your screen. In a *Windows* system, SCILAB 2.6 is typically installed in the sub-directory *c:\Program Files\Scilab-2.6*

How to get documentation for SCILAB
To get documentation on SCILAB scroll down the SCILAB main web page and click on the *documentation* link. This will take you to their documentation page, showing the following options:

- Introduction to Scilab

- Communication Toolbox Documentation

- Signal Processing

- Lmitool: Linear Matrix Inequalities Optimization Toolbox Documentation

- Metanet User's Guide and Tutorial

- Scicos

- Scilab's Internals Documentation

- HOWTO's Scilab

- Scilab's demonstrations

The documentation is available in *html* format, which you can access by clicking in any of the options shown above. If you want to have your own copy of the documentation in *PostScript* or *PDF* format, press the appropriate option in the web page.

Things that SCILAB lets you do

SCILAB can be used for simple arithmetic operations as well as for some algebraic operations, to generate graphics, to program functions, and to solve linear algebra problems and ordinary differential equations, among other things. You can also program SCILAB and produce simple or fancy graphic user interface components for your program.

Getting started with SCILAB

To *get started*, launch the SCILAB application.

To *load a value into a variable* use an assignment statement (one that includes the equal sign), e.g., a = 3.2. Try the following exercises for simple arithmetic operations:

a = 3.2 <return>
b = 6.4 <return>
a+b <return>
a-b <return>
a*b <return>
a/b <return>
a^b <return>
who <return>

The last command will return a list of the active variables in your worksheet. Many of them are system variables that cannot be modified by the user.

SCILAB has a number of *special constants*, such as *%i, %pi,* and *%e*, corresponding to the unit imaginary number, π = ratio of circumference to diameter, and e = the base of the natural logarithms, respectively. The value *%eps* is another special constant corresponding to the maximum value for which *1 + %eps = 1*. Other important constants are *%inf* = infinity, and *%nan* = not-a-number. Boolean (i.e., logical) constants are *%t* = true and *%f* = false.

Try the following exercises to see what values are returned by SCILAB:

%e <return>
%i <return>
%pi <return>
%eps <return>
%inf <return>
%nan <return>
%t <return>
%f <return>

Comments in SCILAB are represented by the double forward slash (//). Anything in front of // is taken as a comment. For example, enter:

a = 4.5 // redefining a <return>

SCILAB will return the value of 4.5 for a and ignore the comment after the //

■ *Scalars*: real, logical, string, polynomial, rational, and basically any object that is not between brackets in SCILAB is referred to as a scalar. Examples are:

```
a = 1 // real constant <return>
2>1 // Boolean constant (i.e., logical) <return>
'my name' // character string or constant <return>
r = poly(1.,'x')  // polynomial with variable 'x' and root at 1.0 <return>
q = y/r  // rational expression <return>
```

■ *Using the "Introduction to SCILAB" demo*: SCILAB is provided with a number of demos to show the software abilities. To access the demos click on the *Demos* menu. This will provide a dialogue window (*SCILAB Choose*) with a list of subjects, such as *Introduction to SCILAB*, *Graphics*, etc. Select the *Introduction to SCILAB*, and press OK. A file with SCILAB commands is loaded in memory. To see each line press <return>. The lines are shown and executed (if executable). Comments have been added to each line to explain the operation shown. (The entire demo session is listed in pages 118 to 137 of the documentation file *Introduction to SCILAB*.) Keep pressing the <return> key to observe a good summary of SCILAB abilities. In particular, pay special attention to item 2 (MATRICES) since SCILAB is a matrix-based system. Also of interest are the items labeled OPERATIONS and SOME NUMERICAL PRIMITIVES. (The term *primitives* is used to refer to simple numerical operations pre-programmed in SCILAB or any other numerical environment).

■ *Using other demos*: To see any of the other demos provided with SCILAB select the option *Demos* from the *File* menu, then select the Demo subject you want (e.g. , *Graphics: Introduction*), and select a particular demo (e.g., *plot2d3* in *Graphics: Introduction*). Notice that, when using graphics in SCILAB, the program generates a window called *SCILABGraphic0*.

■ *Getting help*: In the *Help menu* there are three options that you can use to get help from SCILAB:
- *Help Dialogue:* provides a list of help items classified by chapters. There will be two windows available using this option, the lower window shows the chapter titles while the upper window shows the subjects in the chapter currently selected. Select the item you want to display and press the Show button located to the left side of the help window.
- *Topic:* enter a topic you want help with in the window shown when using this option and press the OK button. If no help is available you will get a message indicating so in the main SCILAB window. If help is available, SCILAB will show you the corresponding instructions. For example, use this option and request information on the keyword *inv*. It will give you information on the inverse operation for matrices.
- *Apropos:* enter a keyword related to the topic that you are looking for (e.g., *inverse*) and press the OK button. You will see a window with topics including the keyword you selected. Scroll through the list and find that topic closer to your interest (e.g., *inv - matrix inverse*), and press OK to get information on that particular topic.

Other menus in the SCILAB worksheet

■ *The "File" menu*: The following options are available in the *File* menu:
- *Getf:* use it to load a function. (Type *help getf* for additional information).
- *Exec:* to execute a script file. (Type *help exec* for additional information).
- *Save*: to save variables (Type *help save* for additional information).
- *Load*: to load a saved variable (Type *help load* for additional information).
- *Change Directory*: self-explanatory.

- *Get Current Directory*: self-explanatory.

■ *The "Functions" menu*:
- *Copy to Clipboard*: lets you copy highligted text to the *Windows* clipboard
- *Paste*: lets you paste text in the clipboard to current SCILAB prompt location
- *Choose Font …*: lets you choose the font in the main SCILAB window
- *History:* access to a menu of commands for editing SCILAB command lines. The character ^ in this menu stands for the *Ctrl* key.

■ *The "Control" menu*
- *Restart*: clears all user-defined variables and initializes environmental variables.
- *Pause:* switch control to *pause* mode. Used mainly for debugging SCILAB functions.
- *Resume:* resume execution after a pause.
- *Abort*: end execution of current program.
- *Interupt* : interrupt execution of current program.

Pause mode: when entering Pause mode, SCILAB creates a new (numbered) prompt. In this mode you can enter SCILAB commands without affecting the main line of calculation (represented by the main SCILAB prompt →). To get into the pause mode you can type *pause* at the main prompt or click on the *Pause* option in the upper menu bar of the main SCILAB window. (The option *Interupt* [sic] in the upper menu bar also allows you to enter into pause mode, however, it will interrupt any operation currently being executed by SCILAB).. You can have several levels of pause, one for every time you enter a *pause* command. To leave the current pause level and move the upper one type *return*.   To move up two levels use the command *quit*. (Note, when used in the main SCILAB prompt, *quit* will terminate your SCILAB session). For more information on the pause mode use the help feature as described above, or simple type *help pause*.

Simple operations with SCILAB

■ *Simple scalar operations*: the following exercises will get you acquainted with a few of SCILAB's abilities for operating with scalar values.
```
a = 2 <return>
b = 3 <return>
Save a <return>
clear a <return>
a <return>
b <return>
load a <return>
a <return>
exp(a) + exp(b) <return>
sin(a*%pi/b) <return>
```
(***Note***: the clear command is used to eliminate variables, as in *clear a*, as shown above. By itself, clear deletes all variables recently defined. Therefore, be very careful when using this command).

■ *Vectors*:
- To enter vectors use the square brackets and separate the elements with commas or blanks for a row vector, e.g.: v = [-1. , 2., %pi].
- The transpose of a vector (or matrix) is expressed using the apostrophe, for example, type: v'
- To enter a column vector, use any of the following procedures: w = [3; -2; 5] <return> or
  r = [6 <return>
  -2 <return>
  10] <return>

- You can create a row vector by indicating a starting value, an increment (or decrement), and an ending value, all separated by the colon (:) operator as follows:

  vector_name = starting_value : increment : ending value

  for example: x = -10.0 : 0.5 : 10.0 <return>
- If you prefer to store it as a column vector, try the following: `xt = x'` <return>
- Let's apply a function to the vector x, try: `y = sin(x*%pi/10)` <return>
- We can plot the vectors *x,y* using: `plot(x,y,'x','y','first plot')` <return>
- [Type `help plot` <return> for more information]
- Let's restart SCILAB and operate with other vectors: Press the *Restart* option in the upper bar menu.
- Enter the row vectors, u = [-1. 2. 3.] and v = [6. -10. 0.]
- Perform the following operations:
  u + v <return>
  u - v <return>
  u*v <return>
  u*v' <return>
  u'*v <return>

- To extract elements of the vectors, try:
  u(3) <return>
  u(2) + v(1) <return>
- Try the following exercise: `a = 1; b = 2; c = 3; r = [a, b, c]` <return>
- To suppress SCILAB responses use the semi-colon after entering a command. For example, try: `s = [-1., 2.];` <return>
- Vectors can also contain characters as their elements, e.g., `letters = ['a', 'b', 'c', 'd']` <return>

  Note: Expressions such as 'a', 'b', etc., are referred to as strings. Therefore, only those string operations such as *concatenation*, *part*, etc. are allowed for vectors with character elements. SCILAB strings and string operations are presented in a subsequent chapter.

Matrices:
- Here are several ways to enter matrices: (press Restart)
  A = [1. 2. 3.; 4. 5. 6.; 1. -1. 0.] <return>
  B = [1. 1. 1. <return>
  2. 3. -1. <return>
  5. 7. -2.] <return>
  u = [1. 3. -5.]; v = [4. 2. 3.]; w = [-1. 0. 1.]; <return>
  C = [u; v; w] <return>
  r = [u, v, w] <return>
  D = [u' v' w'] <return>
- *Matrix operations*: try the following operations:
  A + B <return>
  C - D <return>
  A*B <return>
  B*A <return>
  C*u <return>
  D*v' <return>
  rank (A) <return>
  inv(A) <return>
  cond(B) <return>
  det(C) <return>

A*inv(A) <return>

inv(B)*B <return>

spec(A) <return> (calculates eigenvalues)

trace(C) <return>

(**Note**: to find out more about any of the commands listed here type *help* followed by the command name, e.g., *help spec*).

■ *Solution of linear systems*: two possibilities are:

(Press *Restart*)

```
A = [1. 3. 2.; 2. 1. -1.; 5. 2. 1.]; b = [2; 3; 4]; <return>
```

xa = inv(A)*b <return>

xb = linsolve(A,b)<return>

(Note: type *help linsolve* to learn more about this command).

Simple SCILAB Input and Output

■ *Output*: To get a list of your current session use the function *diary*. The format is as follows: *diary (output_filename)*, where the filename is written within quotes. To end collecting output in the current diary output file use *diary(0)*. For example, try the following session:

(Press *Restart*)

```
diary ('session1.txt') <return>
A = [1. 2. 3.; 2. 3. 1.; 3. 2. 1.];b=[5; 4; -1.];<return>
A <return>
b <return>
x = linsolve(A,b) <return>
diary(0) <return>
```

Next, use NOTEPAD, or other text editor, to open the file *session1.txt* to see the contents of the file. The default directory for storing a *diary* file is the *bin* subdirectory within the SCILAB directory.

Note: The SCILAB worksheet does not allow cutting and pasting. Therefore, the use of the *diary* function with a filename is the only way to copy SCILAB output into a text file.

■ *Command Input*: you can read SCILAB commands from a script file, which is basically a text file listing all the commands you want to use. As an example, create the following text file in the *bin* subdirectory of the SCILAB directory using NOTEPAD, and call it *session2.txt*:

```
//----------------------------------------------------------------------
clear
A = [1. 2. -3.          // entering
3. 4. 5.                // elements of
7. 8. 9.]               // matrix A
b = [1.; 2.; 3.]        // enter vector b
xa = inv(A)*b           // calculate x using matrix inverse
xb = linsolve(A,b)      // calculate x using SCILAB's own linsolve function
//----------------------------------------------------------------------
```

Then, press *Restart* in SCILAB, and type: *exec('session2.txt')*

You will see SCILAB execute all the commands in your file, stopping at the end of the file.

SCILAB command history

All commands entered in a given SCILAB session get stored into a SCILAB command history buffer. The commands are thus accessible for re-use or edition.   All the command history functions are available through the option *History* under the *File* menu in the SCILAB worksheet. The most useful commands are *cntl-P* and *cntl-N*, which lets you access the previous command or the next command, respectively, in the command history buffer. Once a command is recalled from the command history buffer it can be edited by using the backspace or delete keys, or by inserting new characters by simply typing at the proper location.

For example, try the following SCILAB session:

1 - Press the *Restart* option in the menu bar to clear the SCILAB environment.
2 - Enter the following commands (you don't need to enter the comments):

```
-->x = [0:%pi/20:2*%pi];
```

```
-->y = sin(x) + sin(2*x);
```

3 - Use *cntl-P* and edit the previous command (`y = sin(x) + sin(2*x);`) to read:

```
-->z = sin(x) + cos(x);
```

4 - Use *cntl-P* once more to edit the previous command (`z = sin(x) + cos(x);`) to read:

```
-->p = cos(x) + cos(2*x);
```

5 - So far you have created vectors *x*, *y*, *z*, and *p*. Next, we use the functions *xset* with the option *'window'*, and the function *plot* to produce a plot of *y*-vs.-*x*:

```
-->xset('window',0); plot(x,y)
```

6 - Use *cntl-P* to edit the previous command to read:

```
-->xset('window',1); plot(x,z)
```

7 - Continue using *cntl-P* to edit the last commands and produce the following plots:

```
-->xset('window',2); plot(x,p)
```

```
-->xset('window',3); plot(y,z)
```

```
-->xset('window',4); plot(y,p)
```

```
-->xset('window',5); plot(z,p)
```

Selective worksheet output

Suppose you have been working on a lengthy SCILAB session whose command history buffer contains some results that produced errors as well as some intermediary results that are not of interest to you for output. For your output, you can select only those commands relevant to your final results

by using *Cntl-P* and *Cntl-N*.   Try the following SCILAB session that explores some basic vector operations using vectors x and y:

1 - Press the *Restart* option in the menu bar.
2 - Enter the following SCILAB commands:

```
-->x = [1, 2, 5, -4]
 x  =

!   1.     2.     5.   - 4. !

-->y = [0, 2, 3,-5]
 y  =

!   0.     2.     3.   - 5. !

-->x*y
     !--error    10
inconsistent multiplication


-->x.*y
 ans  =

!   0.     4.     15.     20. !

-->sum(ans)
 ans  =

     39.

-->sum(x.*y)
 ans  =

     39.

-->x*y'
 ans  =

     39.

-->x'*y
 ans  =

!   0.     2.      3.   - 5.  !
!   0.     4.      6.   - 10. !
!   0.    10.     15.   - 25. !
!   0.  - 8.    - 12.     20. !

-->y'*x
 ans  =

!   0.     0.      0.      0.  !
!   2.     4.     10.   - 8.  !
!   3.     6.     15.   - 12. !
! - 5.  - 10.   - 25.     20. !

-->x*y' + y'*x
```

```
ans   =

!   39.     39.     39.     39. !
!   41.     43.     49.     31. !
!   42.     45.     54.     27. !
!   34.     29.     14.     59. !

-->b = x*y' + y'*x
 b   =

!   39.     39.     39.     39. !
!   41.     43.     49.     31. !
!   42.     45.     54.     27. !
!   34.     29.     14.     59. !
```

Note: These commands and their corresponding results represent an exploratory session for vector operations.

3 - Suppose that you are only interested in the commands defining vectors x and y, in the operations that produce the dot product of the vectors (i.e., sum(x.*y) and x*y'), and in the very last command (b = x*y' + y'*x). Using the *diary* command create the file *c:\myVectors.txt* and collect only the commands of interest out of the command history buffer by using *cntl-P* and *cntl-N* as needed.   The resulting SCILAB session should look like this:

```
--> diary('c:\myVectors')

-->x = [1, 2, 5, -4]
 x   =

!   1.     2.     5.   - 4. !

-->y = [0, 2, 3,-5]
 y   =

!   0.     2.     3.   - 5. !

-->sum(x.*y)
 ans   =

     39.

-->x*y'
 ans   =

     39.

-->b = x*y' + y'*x
 b   =

!   39.     39.     39.     39. !
!   41.     43.     49.     31. !
!   42.     45.     54.     27. !
!   34.     29.     14.     59. !

-->diary(0)
```

The session, except for the very first command () is stored in file *c:\myVectors.txt*. This file can be edited or printed from a text editor such as NOTEPAD, or my favorite, PFE (*Programmer's File Editor*) available for free from:

http://www.lancs.ac.uk/people/cpaap/pfe

The PFE software

PFE is a text editor designed for text-based programming in languages such as FORTRAN, JAVA, C++, etc. PFE allows the user direct access to DOS windows for compilation and testing of programs, runs DOS commands from the PFE interface, allows recording and playing of Macro commands, and permits easy navigation of files by allowing line numbering for easy debugging of programs.

PFE can be used in combination with SCILAB to type scripts and function files or to edit *diary* files produced from within SCILAB.

Current directory / creating a work directory

SCILAB uses a current directory where files are saved by default, for example when using the function *diary*. To see the current directory use:

```
-->pwd
```

Under a Windows operating system, the default current directory is typically *c:*

The command *pwd* stands for <u>*p*</u>rint <u>*w*</u>orking <u>*d*</u>irectory.

I recommend that you create a sub-directory, or folder, called *work* and locate it under the SCILAB main directory. For example, under a Windows operating system, the SCILAB main directory will typically be

c:\Program Files\SCILAB2.5

Thus, your *work* directory would correspond to:

c:\Program Files\SCILAB2.5\work

At the beginning of a SCILAB session, you can change the current directory to the *work* directory by using the function *chdir*:

```
--> chdir('c:\Program Files\SCILAB2.5\work')
```

You can use your *work* directory to store scripts and functions that you create. For example, if you create a script called *script1* in the *work* directory, once you change the current directory to the work directory, you can simply use:

```
-->exec('script1')
```

to run your script.

If you type a function called *function1* in the *work* directory, with the work directory selected as the current directory, you can load the function by simply using:

```
-->getf('function1')
```

Scripts and function files can be created using NOTEPAD, PDE, or any other text editor.

A preview of SCILAB functions

Here are some *useful functions* in SCILAB that we will explore in more details in subsequent chapters:

- Elementary functions: `sum, prod, sqrt, diag, cos, max, round, sign, fft`
- Sorting: `sort, sortup, gsort, find`
- Specific matrices: `zeros, eye, ones, matrix, empty`
- Linear algebra: `det, inv, qr, svd, bdiag, spec, schur, trace`
- Polynomials: `poly, roots, coeff, horner, clean, freq`
- Random numbers: `rand`
- Programming: `function, deff, argn, for, if, end, while, select, warning, error, break, return`
- Comparison symbols: `==, >=, >, <=, <, =, &` (and), `|` (or)
- Execution of a file: `exec`
- Debugging: `pause, return, abort`
- Spline functions, interpolation: `splin, interp, interpln`
- Character strings: `string, part, evstr, execstr`
- Graphics: `plot, xset, driver, plot2d, xgrid, locate, plot3d, Graphics`
- Ordinary differential equation solvers: `ode, dassl, dassrt, odedc`

To find out more about these functions use the help command. For example, try:

```
-->help roots

-->help eye

-->help trace
```

Exercises

Determine the result of the following calculations using SCILAB if $a = 2.3$, $b = -2.3$, $c = \pi/2$, $x = 2/\pi$, and $y = \sqrt{3}$:

[1]. $(a^2 + bc + x)$
[2]. $sin(c) + y/c$
[3]. $(a+c)/(x+y)$
[4]. $1/(cos(c) + ln(x))$
[5]. $(a+c)^3/b$

Check if the following Boolean statements are true or false based on the values of *a, b, c, x,* and *y* given above.

[6]. $a > c$
[7]. $a = b$
[8]. $(2a+b)/x^2 < 1$
[9]. $x + 2ab + b^2 \leq 23$
[10]. $2ac = 2cb$

[11]. Use SCILAB's help facility to find out information about function *deff* and use it to define a function $y = f(x) = x^2 + 1$.

Using the vectors $u = [3, 2, -1]$, and $v = [4, -6, 2]$, calculate the following operations:

[12]. $w = u+v$
[13]. $r = u./v$
[14]. $z = v*u$
[15]. $t = v.*u$

Using the matrices **A**, **B**, and **C**, shown below, perform the following operations:

$$\mathbf{A} = \begin{bmatrix} 5 & 5 & -2 \\ 1 & -2 & 0 \\ 4 & -1 & -2 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} -5 & 4 & -\pi \\ -1 & -1 & 10 \\ 0 & 4 & -2 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 2 & 5 \\ 1 & -3 \\ 4 & 2 \end{bmatrix}$$

[16]. **A** + **B**
[17]. **A·B**
[18]. **B·A**
[19]. **B·C**
[20]. **A· B·C**

REFERENCES (for all SCILAB documents at INFOCLEARINGHOUSE.com)

Abramowitz, M. and I.A. Stegun (editors), 1965,"*Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*," Dover Publications, Inc., New York.

Arora, J.S., 1985, "*Introduction to Optimum Design*," Class notes, The University of Iowa, Iowa City, Iowa.

Asian Institute of Technology, 1969, "*Hydraulic Laboratory Manual*," AIT - Bangkok, Thailand.

Berge, P., Y. Pomeau, and C. Vidal, 1984,"*Order within chaos - Towards a deterministic approach to turbulence*," John Wiley & Sons, New York.

Bras, R.L. and I. Rodriguez-Iturbe, 1985,"*Random Functions and Hydrology*," Addison-Wesley Publishing Company, Reading, Massachussetts.

Brogan, W.L., 1974,"*Modern Control Theory*," QPI series, Quantum Publisher Incorporated, New York.

Browne, M., 1999, "*Schaum's Outline of Theory and Problems of Physics for Engineering and Science*," Schaum's outlines, McGraw-Hill, New York.

Farlow, Stanley J., 1982, "*Partial Differential Equations for Scientists and Engineers*," Dover Publications Inc., New York.

Friedman, B., 1956 (reissued 1990), "*Principles and Techniques of Applied Mathematics*," Dover Publications Inc., New York.

Gomez, C. (editor), 1999, "*Engineering and Scientific Computing with Scilab*," Birkhäuser, Boston.

Gullberg, J., 1997, "*Mathematics - From the Birth of Numbers*," W. W. Norton & Company, New York.

Harman, T.L., J. Dabney, and N. Richert, 2000, "*Advanced Engineering Mathematics with MATLAB® - Second edition*," Brooks/Cole - Thompson Learning, Australia.

Harris, J.W., and H. Stocker, 1998, "*Handbook of Mathematics and Computational Science*," Springer, New York.

Hsu, H.P., 1984, "*Applied Fourier Analysis*," Harcourt Brace Jovanovich College Outline Series, Harcourt Brace Jovanovich, Publishers, San Diego.

Journel, A.G., 1989, "*Fundamentals of Geostatistics in Five Lessons*," Short Course Presented at the 28th International Geological Congress, Washington, D.C., American Geophysical Union, Washington, D.C.

Julien, P.Y., 1998,"*Erosion and Sedimentation*," Cambridge University Press, Cambridge CB2 2RU, U.K.

Keener, J.P., 1988, "*Principles of Applied Mathematics - Transformation and Approximation*," Addison-Wesley Publishing Company, Redwood City, California.

Kitanidis, P.K., 1997,"*Introduction to Geostatistics - Applications in Hydogeology*," Cambridge University Press, Cambridge CB2 2RU, U.K.

Koch, G.S., Jr., and R. F. Link, 1971, "*Statistical Analysis of Geological Data - Volumes I and II*," Dover Publications, Inc., New York.

Korn, G.A. and T.M. Korn, 1968, "*Mathematical Handbook for Scientists and Engineers*," Dover Publications, Inc., New York.

Kottegoda, N. T., and R. Rosso, 1997, "*Probability, Statistics, and Reliability for Civil and Environmental Engineers*," The Mc-Graw Hill Companies, Inc., New York.

Kreysig, E., 1983, "*Advanced Engineering Mathematics - Fifth Edition*," John Wiley & Sons, New York.

Lindfield, G. and J. Penny, 2000, "*Numerical Methods Using Matlab®,"* Prentice Hall, Upper Saddle River, New Jersey.

Magrab, E.B., S. Azarm, B. Balachandran, J. Duncan, K. Herold, and G. Walsh, 2000, "*An Engineer's Guide to MATLAB®*", Prentice Hall, Upper Saddle River, N.J., U.S.A.

McCuen, R.H., 1989,"*Hydrologic Analysis and Design - second edition*," Prentice Hall, Upper Saddle River, New Jersey.

Middleton, G.V., 2000, "*Data Analysis in the Earth Sciences Using Matlab®,"* Prentice Hall, Upper Saddle River, New Jersey.

Montgomery, D.C., G.C. Runger, and N.F. Hubele, 1998, "*Engineering Statistics*," John Wiley & Sons, Inc.

Newland, D.E., 1993, "*An Introduction to Random Vibrations, Spectral & Wavelet Analysis - Third Edition*," Longman Scientific and Technical, New York.

Nicols, G., 1995, "*Introduction to Nonlinear Science*," Cambridge University Press, Cambridge CB2 2RU, U.K.

Parker, T.S. and L.O. Chua, , "*Practical Numerical Algorithms for Chaotic Systems*," 1989, Springer-Verlag, New York.

Peitgen, H-O. and D. Saupe (editors), 1988, "*The Science of Fractal Images*," Springer-Verlag, New York.

Peitgen, H-O., H. Jürgens, and D. Saupe, 1992, "*Chaos and Fractals - New Frontiers of Science*," Springer-Verlag, New York.

Press, W.H., B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling, 1989, "*Numerical Recipes - The Art of Scientific Computing (FORTRAN version)*," Cambridge University Press, Cambridge CB2 2RU, U.K.

Raghunath, H.M., 1985, "*Hydrology - Principles, Analysis and Design*," Wiley Eastern Limited, New Delhi, India.

Recktenwald, G., 2000, "*Numerical Methods with Matlab - Implementation and Application*," Prentice Hall, Upper Saddle River, N.J., U.S.A.

Rothenberg, R.I., 1991, "*Probability and Statistics*," Harcourt Brace Jovanovich College Outline Series, Harcourt Brace Jovanovich, Publishers, San Diego, CA.

Sagan, H., 1961,"*Boundary and Eigenvalue Problems in Mathematical Physics*," Dover Publications, Inc., New York.

Spanos, A., 1999,"*Probability Theory and Statistical Inference - Econometric Modeling with Observational Data*," Cambridge University Press, Cambridge CB2 2RU, U.K.

Spiegel, M. R., 1971 (second printing, 1999), "*Schaum's Outline of Theory and Problems of Advanced Mathematics for Engineers and Scientists*," Schaum's Outline Series, McGraw-Hill, New York.

Tanis, E.A., 1987, "*Statistics II - Estimation and Tests of Hypotheses*," Harcourt Brace Jovanovich College Outline Series, Harcourt Brace Jovanovich, Publishers, Fort Worth, TX.

Tinker, M. and R. Lambourne, 2000, "*Further Mathematics for the Physical Sciences*," John Wiley & Sons, LTD., Chichester, U.K.

Tolstov, G.P., 1962, "*Fourier Series*," (Translated from the Russian by R. A. Silverman), Dover Publications, New York.

Tveito, A. and R. Winther, 1998, "*Introduction to Partial Differential Equations - A Computational Approach*," Texts in Applied Mathematics 29, Springer, New York.

Urroz, G., 2000, "*Science and Engineering Mathematics with the HP 49 G - Volumes I & II*", www.greatunpublished.com, Charleston, S.C.

Urroz, G., 2001, "*Applied Engineering Mathematics with Maple*", www.greatunpublished.com, Charleston, S.C.

Winnick, J., , "*Chemical Engineering Thermodynamics - An Introduction to Thermodynamics for Undergraduate Engineering Students*," John Wiley & Sons, Inc., New York.