



Randle, Inc.

CSP v2.2
Users Guide

PO Box 1010 Great Falls, Virginia 22066 (703) 759-5257

TABLE OF CONTENTS

INSTALLATION	1	
UNIX	1	
MACINTOSH	2	
IBM	2	
FAST START: UNIX/IBM	3	
INTRODUCTION	5	
A WORD ABOUT THE MACINTOSH	5	
A WORD ABOUT THE IBM	7	
CSP ORGANIZATION	8	
GENERAL PROCEDURES	9	
THE SEQUENCE OF OPERATION	9	
IMPORTING DATA	13	
ASCII Data	14	
MIDAS	14	
CSP.INIT	15	
THE LOG FILE	16	
CLEANUP	16	
BENIGN & MALIGNANT STOPS AND WARM STARTS	16	16
CHANGE HISTORY	18	
THE MACINTOSH SYSTEM MANAGER	22	22
DISPLAY ATTRACTOR	22	
SIGNAL TRACE	23	
PLOT RESULTS	23	
THE UNIX/DOS SHELL	25	
IBM/UNIX GRAPHICS	25	
THE IN-CORE MODULES	29	
EMBED	29	
MUT	30	
FAMI (Fast Average Mutual Information)	31	31
CFD (Correlation Dimension)	31	
CFDTrim	32	
Lyapunov (now obsolete)	32	
Local Lyapunov exponents	33	

THE TOOLKIT	35	
ADDNOISE	36	
ASC2CSP	36	
CHKDATA	36	
WAV2-	37	
MPGS (Cray specific)	37	37
XGRAPH (Unix only)	37	37
SIGNAL SIMULATORS	37	
MAKEDUF	38	
MAKELOR	38	
MAKEROS	38	
MAKESIGS	38	
MODULATE	38	
CLIPM	38	
PSK	38	
MODULE LISTING	39	
ADDITIONAL INFORMATION	46	
Direct Calls	46	

INSTALLATION

UNIX

CSP/UNIX is distributed as a tar file on either a 160 MB QIC tape or 1.44 MB diskettes. To load the software from the tape, create (mkdir) a directory (such as CSP or chaos). Go to that directory (cd chaos). After inserting the tape into the drive, use the command "tar -xvf /dev/rst0" to transfer the files. On some systems, the device is "nst0" vice "rst0." Then, rewind the tape with mt -f /dev/rst0 rewind. Save the tape as a backup.

For 1.44 MB diskette use the following commands to move the tar file and uncompress it:

```
tar -xvf /dev/rfd0c           { move tar file to the hard drive
uncompress chaos.tar         { uncompresses the tar file
tar -xvf chaos.tar           { extracts file and puts them into the
subdirectory
eject                         { ejects floppy
cd chaos                       { puts you into the chaos directory
```

Add the directory to your path command (IBM: autoexec.bat, UNIX: setpath).

The primary graphing system for CSP on the Sun is a copyrighted freeware graphics system called xvgr. Xvgr was written by Paul J. Turner who is at the Center for Coastal and Land-Margin Research, Oregon Graduate Institute of Science and Technology. Respecting the terms of his copyright, we do not include xvgr on the Randle, Inc. distribution media. It can be acquired via internet anonymous ftp from ftp.ccalmr.ogi.edu.

For the cost of media and time (about \$50), Randle will provide a copy of the files that are available from ftp.ccalmr.ogi.edu.

Xvgr should be placed in the directory usually used for general system utilities -- /usr/bin, for example, or in the csp directory.

Alternatively, you can use "xgraph." Xgraph is an X-Windows utility. Be certain that the X-windows system is present and that xgraph is included. Modify the setpath environment so that xgraph can be called with the single command "xgraph" (no path or modifiers needed). Edit or create a file called csp.init and add the line "cspplot:xgraph" (do not include the "). csp.init should be in the same directory as the executable files.

MACINTOSH

The files are in a self extracting "stuffit" file. Move the icon to the folder you wish to use. Double click on the stuffit icon. You're ready to go!

IBM

Why?

If you must, copy all the files to the appropriate directory. Add the directory to your path. Have fun with toy problems. Buy a Mac.

FAST START: UNIX/IBM

1. Enter cspshell.
 2. Select generate simulated signals. Select Duffing. Enter the suggested parameters. The signal simulator for Duffing's oscillator will take a few minutes to generate three files: duf_a.dat, duf_v.dat, duf_d.dat.
 3. (IBM) After the main menu returns, select plot (p). Then, at the prompt for file name enter duf_a.dat. The next prompt will ask for signal trace or attractor. Enter t. A signal trace for the acceleration in Duffing's oscillator will appear. Pressing "f" will generate the FFT. Press esc to return to them menu.
 4. (IBM) Select p again. Enter duf_a.dat (again), but this time enter "a" at then next prompt. Several parameters will be asked for. Enter 16 or 21 when asked for time lag. And, press n when asked if you want to arm the screen grabber. The reconstructed attractor will appear as an animation on the IBM. Pressing esc will return you to the main menu (IBM).
 5. Precomputed output files for this signal are on the diskette and should have been copied to your hard drive. Select plot (p) from the main menu and enter duf_a.mut. If a minimum time is asked for (UNIX) enter 0 or 1. A plot of the average mutual information will appear.
 6. When the menu returns, select plot again to display duf_a.emb, duf_a.qcr, and duf_a.lyp. When the first correlation dimension screen is up -- the top is the raw counts and the bottom is the local slope of a log-log plot (not shown) -- use the left and right cursor control keys to move the marker to a place on the bottom curves where the slopes seem to converge at small radii (say $r = 3$ or 4). Press enter, and the final correlation dimension estimate graph will appear. If you do not move the cursor keys, a radius of 1/4th of the mean attractor size will be used (a pretty good number).
- Now, let's process the signal (don't really do this on the IBM, the files diskette has these computations already done).
7. Select fast mutual information from the main menu. Use 10,000 samples (you can use more under UNIX, but the IBM will blow up). Use a maximum time of 50. Fast mutual information will run until it finds the first local minima. It then returns to the main menu (IBM). The UNIX fast mut runs in background.
 8. After fast mutual info has run, select embedding dimension from the menu. On the IBM, keep it simple (fast mut uses an array that is huge by IBM standard and the machine may crash). Note that the optimum time lag

computed by fast mut is supplied as a recommended value. If the full mutual information algorithm was used, the optimum time lag computed by that program is shown. If these routines were not run first, you are on your own -- I generally use 16, but any low number will do.

9. Then, use average mutual information out to 100 or 200 time lags.

10. Finally, compute correlation dimension and Lyapunov exponents (UNIX only).

11. Finally, display the graphics by selecting plot from the menu. If you enter the file type (.dat, .emb, .qcr, .lyp, or .mut) the plotter will select the appropriate routine. If you do not enter a file type the system looks for a data file (.dat) and, if present, assumes you want to either show the signal trace or a reconstructed attractor.

INTRODUCTION

CSP (Contemporary Signal Processor) is a software system for processing signals using methods derived from the study of chaotic systems. CSP consists of five "in-core" modules, user shells for the in-core modules, a data conversion and handling toolkit, and platform specific graphics support. The toolkit provides platform independent conversion of graphics data for many popular graphics systems. Simulated signal generators are provided as part of the toolkit.

If you are not familiar with the principles of nonlinear signal processing, a companion booklet, "Contemporary Signal Processing," provides an overview and some details of the procedures. A separate software system is available for tutorials.

The system is command driven in the UNIX and DOS environments. Each module is loaded and run separately. The Apple Macintosh version has a unified shell (CSPmain) that contains all the functions. The parameters are identical to the UNIX system, but the user enters the values in menus, rather than command line prompts. In many cases, additional menu items and information are available to the Apple user (ASCII conversions for example). An Apple like shell for UNIX is contemplated for the next major version (sometime in early 1994).

As a final note, the best native host for CSP is either a Cray or a Sun. While it runs on smaller systems, it is designed for intensive signal processing in a signals laboratory.

A WORD ABOUT THE MACINTOSH

The Apple Macintosh IIci with System 7 and at least 8 MB of memory are required to run CSP. The more memory the better. CSP does not check to see if System 7 is available, so the error message under System 6.x is simply a crash (yes, someday we'll fix this).

Background processing by the CSP kernel is supported on the Mac. But, System 7 is not a true multitasking system. Rather, Apple uses what they call "cooperative" multitasking. The applications arbitrate processor resources among themselves, rather than having the operating system do it. The net effect is that background processing on the Mac is not as robust or efficient as it is under Unix. Windows come and go and resources get hogged or clash occasionally. The system response in the "front" window may degrade.

User control over CSP background processes is handled through a menu bar item called "Control." When a background application is launched, two Apple specific run parameters are set so that the front application (the CSP main menu) continues to run efficiently. The background application is running, but with a low resource burden. The user may go to finder (to hide the CSP menu) and click on the Control bar. It may be necessary to hold the button down until the application responds. Select the only option --

"Response and Time slice." A control window will then pop up.

Response is the number of internal processes that are completed between each check for outside events. A higher number (say 1,000 or 5,000) allows the background application to run without the high overhead burden of frequent checks for outside events. But, response in the front window will degrade. A low number (say 1 to 10) will cause the background application to be better behaved with respect to your word processor, at the expense of processing efficiency. Someday, Apple will wise up and develop a true multitasking system.

Time slice is the amount of time the background process allows for outside processes when requested. A high number, say 500 to 1000, allows heavy resource users time for completion before the background application resumes. A low number (say 1 or 2) only provides enough time to grab an event or two (such as a menu request for access to the Control menu). The lowest number accepted is 1, to insure that control can always be regained (even if after a long wait for response).

Because the kernels are processor and memory intensive, the user should experiment to determine how many processes can efficiently run simultaneously. This is at most one or two, depending on problem complexity. Processing may also slow when multiple kernels are running because they may attempt to share data and output resources. For example, a record of all processing activity for a given data file is kept in a log file. If one process has opened this file and another application needs to write to it, the second process must wait until the log file is closed. The same problem may occur when data files are needed for input. A background program can be speeded up by closing other windows or by bringing the offender to the foreground (double click on the window or use multifinder). The standard partition size for the Kernel programs is 8 - 16 MB. If a module refuses to run and you have a lot of data and are in a high dimension, try increasing the partition size.

Some of the Macintosh utilities belie their Unix heritage and run as console windows. The Macintosh console window is not one of Apple's better efforts and often a mis-entered parameter causes bizarre behavior. Just "Command ." out and start over.

CSPmain is the control center for the Macintosh version. All operations can be performed from this shell, including file conversions and graphics plotting.

A WORD ABOUT THE IBM

Don't.

Randle can supply an unsupported DOS version and graphics systems. This is essentially our Unix code. But, experience has taught us that the IBMs are worthless for these problems. It possible that with a "Pentum" chip and OS/2 or Windows NT the situation will change. If so, we will revisit the issue of support for the IBM.

CSP ORGANIZATION

The CSP system is organized into three categories of modules. The in-core Kernel consists of the processing intensive modules that can run in background mode on machines that support multiprocessing. The modules are accessed through a workstation shell that prompts the user for parameters and then calls the appropriate kernel to execution. These modules perform the computations for embedding dimension (embed), Average Mutual Information (mut), fast mutual information (FAMI), cross mutual information (mutx), fractal dimension (CFD), global Lyapunov exponents (lyp), local lyapunov exponents (loclyp) and noise reduction (PSC -- if your license includes the PSC noise reduction system).

CSP uses its own data format and file structure. The conversion tools convert data from other representations, such as ASCII, to the CSP format. The conversion toolkit also includes tools to generate files to display state space portraits in three dimensions using the Cray proprietary Multi Purpose Graphics System (MPGS) on a Silicon Graphics workstation, or the WAVEFRONT display system.

GENERAL PROCEDURES

Many of the parameters and prompts are common to several CSP modules. These include data file requests, etc.

When a shell or tool asks for a data file name, the data file extension (.dat) need not be given. The system assumes that where appropriate only a valid CSP data file can be used. Even if a file has the CSP data file extension, most modules check internal file information to insure that the file is valid for the process. The two exceptions to this are CHKDATA and ASC2CSP. Because these tools work on any file type, it is necessary to specify the file type (extension) when running the tool.

Many modules ask for a "skip" value. Skip allows the user to bypass data at the beginning for the file. Thus, uninteresting data or previously studied data, may be ignored. Entering a "0" causes the module to begin processing with the first value in the file.

The user shells that initiate the in-core modules prompt for a "VERBOSE" parameters. If the user responds with "y", then the in-core module will display intermediate results and status information in the current window. Responding with "n" will suppress all output to the terminal.

Again, the Apple system accesses all operations from the main menu, CSPmain.

THE SEQUENCE OF OPERATION

On the Macintosh, all CSP functions are accessed from the main menu. Double clicking on the application called CSPmain (I rename mine to just "CSP") starts the system.

Procedurally, the operations are

0 Import data (see next section).

1 *View the signal trace, FFT, or polyspectra.* If these transforms are uninteresting, there is no reason to proceed with the more computation intensive methods of chaos. The issue here is whether the data have a broad-band spectra.

2 *Make a preliminary estimate of the time delay, T .* Experience has shown that a reasonable estimate of time delay in two dimensions is sufficient for determining the minimum embedding dimension, d_E , using the method of false nearest neighbors in any dimension. The optimum value for the time delay is determined from the first local minima of the average mutual information for the signal. Precise estimates of the numerical value of $I(T)$ are not needed because we are interested only in the first local minima of the time delay (T_{opt}). So, a fast but magnitude-inaccurate algorithm (fast average mutual information -- FAMI) can be used to find this point. The full

algorithm may be used with no penalty other than computation time. If the accurate algorithm is used, T_{opt} for two dimensions provides a time delay that can be used to directly view the attractor. Because both the algorithms provide a sample size constrained lower bound of $I(T)$, use the largest sample size possible and the fast algorithm at this point.

2a *Take a peek at the attractor.* Even though little is known about the signal at this point, it is worthwhile to take a look at the attractor in three dimensions. Visualization may provide clues to the nature of the system and/or defects. There are many potential problems in signal collection and some of these are strikingly apparent in a visualization, especially if animated. Clipping and undersampling are an occupational hazard in signal processing and these defects are immediately noticeable. Then again, if the signal has a dimension much over 3.5, features begin to become indistinct as a consequence of the embedding theorem (or perhaps in deference to it as a fundamental law).

3 Using the estimate of time delay, *compute the minimum embedding dimension (d_E) using false nearest neighbors.* One can presuppose that any real signal of interest has an embedding dimension of at least two. However, the computations for dimension one are reasonable fast and provide a baseline estimate, so start with dimension one. Estimating the upper limit is less clear. This is why we are doing the calculation. A practical limit is imposed by the amount of data available and its SNR. In general, one should use as much data as is available. We usually do preliminary studies with 50,000 to 60,000 samples and the final run for high dimension data with 120,000 to 200,000 samples. Depending on the system and its dimension, far fewer samples may suffice. A mere 20,000 samples provide good estimates of the embedding dimension for the classic Lorenz system.

There is always a temptation to compute values for high dimensions, even if there is insufficient data. In any case, dimension 10 usually suffices. Just be careful interpreting the results of there are few data. In several published cases, estimates of system dimension actually seem to be efficient calculations of the space occupied by a insufficiently sampled attractor.

The definition of "close" is arbitrary. The system default for the metric is 15.0. Occasionally, however, the default is too large and the number of false neighbors in the lowest dimension (i.e., 1) is intuitively too small. In this case, use a lower metric -- 10. or even 5. Experience has shown that the final estimate of dimension does not change even though the number of neighbors in a given low dimension may.

3a If insufficient data is a concern, then *the method of Kennel and Isabelle* may shed some light on whether the calculated d_E is the undesirable artifact of a sparse attractor or the desired effect of system dynamics. This technique applies false nearest neighbor testing to known random data with the same distribution and same number of samples. If the false neighbors in any dimension for the random number set is similar to the real data, then the results can be presumed to result from under sampling.

4 The correlation dimension, d_A , comes next as a proxy for the fractal or Hausdorff dimension. Even with the comparative efficiencies of the Grassberger-Procaccia algorithm, these are time consuming calculations. The computation penalty is significant in higher dimensions. In general, one would like to compute the correlation integral out to dimension 10, but $d_E + 1$ may do in the interests of time. Computing the higher dimensions does provide a more satisfying final curve, so dimension 10 is a good target.

These computations cannot be restarted or extended after completion because the computations are carried out as a function of time delay rather than dimension so the intermediate results are dependent on the assumed upper limit for d . Occasionally, one finds that the assumed upper limit is insufficient and, assuming there is sufficient data, must start over.

A more practical issue is how well data populate the attractor and the resolution of the computations. The issue here is representation of numbers taken from a sensor and the number of samples. Sensors and analog-to-digital converters deliver a variety of numeric formats and ranges. Typically, a A/D converter may provide an integer number that represents some scaled and translated range of analog measurements at the sensor. The analog floating point value is now an integer. Although there is no reason that integer valued floating point numbers cannot be processed, most signal processing systems convert the integer numbers back to floating point and scale the data to the another, perhaps original, range. Thus, a sensor output of 5.3333 mV, for example, may become 187 leaving the converter and back to 5.3300 in the signal processor.

For correlation dimension calculations, the histogram of distance between vectors is computed as

$$C[r] += 1 \text{ if } |x_i^2 + x_i^2| \leq r$$

Because r is used as an index it must be an integer. There is an ambiguity associated with each vector that is the resolution of the calculation. For the example number, 5.3300 may be indistinguishable from 5.6000. If the range of floating point values is small, the resolution is poor. The fine structure at small values of r is lost. But, the definition of the correlation integral requires good resolution for small radii.

The first application method is to use a multiplier to improve the resolution of close vectors. Scaling and translation of numeric values only moves the attractor and changes its size. The dynamics and invariant geometric properties are preserved. So, multiplying the floating point number by 10 or 100 increases the resolution of fine structure. A larger histogram is required, but there is no computation penalty because all radii must be computed anyway. The magnitude of the multiplier depends on the characteristics of the numbers. Using a multiplier of 10 on the example of 5.3300 increases the resolution by an order of magnitude. The difference between 5.33 and 5.60 becomes apparent. The first, obvious, consideration is how far can we take this? For this example, a multiplier of 100 provides all the improvement

possible.

But, there is another problem. The increase in resolution comes at the expense of population density at each radius and more samples must be used, if possible. Even so, the fluctuations between histograms can be large. Because the calculation depends on local slopes at small radii, these slopes can vary widely. One radius may have a sufficient count while the next one is half the count because insufficient data were available. It then becomes difficult to determine which radius to use for the final estimate. This effect can be mitigated by *smoothing over a moving average*. Experience shows that for moderately rough curves, a moving average over 10 points is adequate. Again, it's a situational problem and the trick is to use enough smoothing to mitigate uneven histograms, but not so much as to completely corrupt the final estimate.

Fortunately, this compute intensive analysis is of little operational value. After it is performed for research and characterization,

5 *The final relevant calculations are Lyapunov exponents.* Global exponents characterize long term behavior over the entire attractor. Local exponents characterize behavior as a function of length (time) and location. Normally, one looks at an aggregate of time periods over many locations to get the best estimate of the global exponents.

We normally use a length (number of jacobians) of 2,048 (2^{11}) samples for the first try. If the curves are asymptotic, look no further. Note that the number of points used to characterize the attractor may be much higher (50,000 to 100,000 samples). This is contextual, however, and one must often use whatever data are available. Experience also has shown that 300 to 500 neighbors are sufficient to develop an accurate map. The "group factor" is used in the indexing of near points in state space. A value of 25 should suffice.

Because the local exponents converge to the global value as L increases, we are dropping support for the original global Lyapunov exponent module soon. The new local Lyapunov exponent module contains more accurate and robust mapping functions with no decrease in performance.

5a *Calculation of the time reversed local Lyapunov exponents* provides confirmation that the estimate of d_E is correct. This feature is a recommended option in the local Lyapunov exponent module.

IMPORTING DATA

The toolkit provides several facilities for porting data into CSP. Data can be converted directly from ASCII files, floating point (binary) data files, WAV and AIFF files (industry standard digitized sound files), and MIDAS files.

The utilities are accessed on the Macintosh by pressing the convert button on the main menu. Because the ASCII conversion is frequently used, a button for this conversion also is provided on the main menu. The Unix shell option "i" takes you to the import data menu.

CSP does not alter the original file, but creates a new file. The new CSP

file has the same name as the original file, but has an extension of "dat." Thus the only precaution that the user must take is to insure that the extension of the original file is NOT "dat." { version 2.3 may use an extension of ".csp" and will check to see if there is a conflict}

ASCII Data

ASC2CSP is the most general conversion routine. While ASCII file handling is storage intensive and has slow I/O, it does have the virtue of being readable on any machine. Thus, as long as appropriate media can be generated, data can be transferred.

After you select a file, the first three lines in the ASCII file are displayed. You can then enter a title and comment that will be stored in the CSP data file header. The options include skipping a specified number of lines (to get past headers, etc.), skipping either a specified number of characters or strings on each line, or selecting a column from a delimited file that was created by an application such as EXCEL or dBase.

These three options are mutually exclusive and selecting one automatically disables the other two. If you select delimited but do not specify a delimit character and/or column, then the system will use the blank (space) character as the delimiter and the first column.

The final options, also mutually exclusive, are to adjust the data to either a zero average or to "normalize" the data. Normalizing the data is highly recommended. It adjusts the data to a zero mean and adjusts the range to -15 to +15 (nothing magic here, just convenient range to work with). Many of the graphics programs sort of expect data with these characteristics and life will be a whole lot easier. Because this is the consummate linear transform, there is no effect on the analytical results.

{ Known bug -- we don't do tab delimited data, as yet }

MIDAS

On the Sun, MIDAS binary (floating point) files can be imported by selecting the "import other files" (i) option from the main menu. The sub-menu selection is ""MIDAS". At the prompt enter the MIDAS file name. This routine then prompts for the number of bytes in a header skip. For example, one MIDAS file format ("Blue") has a 512 byte header, so enter 512 at this prompt. The data file produced by this routine has a type (extension) of ".da" (vice ".dat") to insure that your original file is not overwritten if it had an extension of ".dat" (which is commonly used). Be sure to rename this file so that it has the required .dat extension!

An alternative import can be done by using a MIDAS tool to convert the MIDAS file to ASCII, then importing the ASCII file.

If you've read this far and are wondering what a MIDAS file is, you don't really want to know. MIDAS is one way the U.S. Government wastes your money. It's a signal processing system that was a marvel in it's time:

the late 1960's. The USG is still using it. Its FORTRAN 66, card bin input heritage is still evident. It stinks. You are paying for it. It employs legions of pedestrian engineers who would otherwise be on welfare.

CSP.INIT

Many of the modules check for a file called csp.init when they start. This file, which can be created with any text editor, contains switch setting for system options. The switch has the form "module name" delimiter "option." For example, to use the Xgraph graphing tool in unix, the init file would contain the line:

```
cspplot:xgraph
```

The file is case insensitive. That is, when the lines are read in, they are converted to lower case before evaluation. You can use upper case, but it is ignored.

The options available are:

global accepted by any module

```
global:nolog          /* disables file specific logging */
global:notime         /* disables capture of cpu processing time */
global:oldps         /* parses older ps for cpu processing time */
```

cspplot (unix)

```
cspplot:xgraph        /* uses X-graph in lieu of xvgr */
cspplot:xvgr          /* not really needed, since xvgr is default */
global:nolog         /* suppresses generation and updating of log
files)
```

CSPmain (Mac)

```
macshell:tutorial    /* uses long introduction */
```

THE LOG FILE

All processing information about a given data file is saved in a file that has the name "datafilename.log." This file is created when the data set enters the CSP system. It can only be erased by user action at the computer system command level (i.e., "erase").

The log file contains a record of the processing, the parameters used,

and some results. It can be suppressed by adding the line "global:nolog" to the csp.init file. Logging can also be disabled for specific modules -- "embed:nolog" for example.

CLEANUP

CSP should be nice to its hosts and clean up all temporary files. But, some temporary files contain run related data for systems development and debugging. The interesting information is in the log file. You can delete any file named "dump" or any file with a type of ".dmp".

BENIGN & MALIGNANT STOPS AND WARM STARTS

All kernels can be hard stopped. On the Macintosh, get the window to the foreground, grab the "file" menu bar and go to quit.

In all versions, a "kill file" can be created that signals individual modules to shut down. Use an editor to create a file called "kill.csp." In this file place the names of the modules you want to stop and, if appropriate, the UNIX PID number (you may have more than one process of the same kernel running in a multiprocessing system such as the Cray). One module and PID per line. Save this file.

It may take a few minutes, but eventually, each kernel looks for kill.csp. If it is present, the kernel opens the file and looks for its own name. On systems that provide PIDs, the kernel then looks to see if the PID in kill.csp matches the system supplied PID. If these matches are made, the process shuts down.

For the moment, only MUT and EMBED shut down benignly. They write output files that reflect the last completed calculation. For MUT, this is the last time lag computed. For EMBED, it is the last dimension computed. These fellows can then be restarted later.

CFD, LYP, and FAMI just hard stop. All results are lost. We're working on this for CFD. There is no point in a benign stop for FAMI. It only runs a few minutes and intermediate results are meaningless.

For information on warm starts, see the section on each kernel and the section titled "Additional information" (if your license includes access to this proprietary information).

CHANGE HISTORY

September 1993

CSP.INIT file added to control options.
Local Lyapunov exponents released for customer testing.
Unique output file names are generated for defaults.

Unix

Solaris 2.2 (Sun OS5.x) supported.
Unix shell is much better behaved, especially under Sun OS4.x.
Primary graphing system is switched to xvgr.

Mac

Graphics system is faster and formatting is improved.

May 1993

Prompts for Time delay: The optimum time delay for 2 dimensions is now shown & is the recommended default.

EMBED

The second test (size/threshold) for false neighbors is now included and plotted. See the embed section.
The shell now permits restarting these calculations. Enter the plot file name (name.emb) when prompted for the file.

MUTUAL INFO. Can be restarted from the shell.

IBM/UNIX

Bug fix in shell: No data loaded when 0 specified for ASCII conversion. Fixed.

More robust file existence checking is now done before a new file is created.

The trimQCR function now works.
Labels on some of the xgraph plots have been changed to increase plot size and clarify axis numerical values.
Error checking in the ASCII conversion routine is improved.
The menu in CSPshell for starting the embed module now (correctly) accepts 0 as a proxy for all points in the file.

UNICOS

NQS now supported via qsub command.

March 1993

Mac:

CSPmain. The length of time the logo is displayed is reduced. Default output file name is now shown for the mutual information startup screen. No checking for such a file is done, yet.

EMBED had the nasty habit of crashing if it needed more memory than was allotted in its size resource. Cause: unknown but probably related to the interaction of the Apple toolkit and the Think C ANSI compliant memory manager. EMBED now calculates the amount of memory needed and gracefully quits. An error message containing the required size is written to CSP_ERROR.

EMBED response to system requests (foreground processing) is improved during the "Index" procedure.

MUTBACK response to system requests is more evenly distributed.

MacCSPPlot eliminates the edit control bar, which was not functional anyway. A new selection is added to the size menu bar. This option toggles the graphics display between color and Black & White. B&W provides better looking images on non-color printers. But, none of the currently supplied graphics use color, so this is a moot item. This feature relates to a new background processor that is still in development.

Working on:

Mac:

CSPedit. The annoying screen flash during initialization is eliminated.

CSPedit. If the window is closed rather than quitting, then the next time CSPedit is called by the shell, it comes to the foreground but has no window and the user must manually request a file from the file menu. Fixed.

Mac:

All modules were reloaded to fix compiler optimization problems (Think C v5.0.4). CSP is now generated on a Quadra 800 under System 7.1, so there should be no '040 cache problems (none had been reported).

CSPedit will accept any file with the type text in addition to CSP log files.

CSP_ERROR file is classed as "text" so it can be viewed from the main menu.

MACCSPPlot and MacState3D now have standard screen sizes.

MACCSPPlot opens with a window that will fill an 8.5x11 page.

MacState3D initiates with a window that fills the screen.

The menu bar "Size" allows selection of a "page" size that fills an 8.5x11 page. An allowance for a border is made. Don't forget, you must rotate the graphic from within your word processor.

The half page window fills 1/2 of a page, so two graphs per page are possible.

Quarter page is a 3.25 inch (1/2 column) graphic. Things get rather tight at this size, so some commentary is not shown and (temporarily?) a few of the MacCSPPlot types refuse to open this window (it reverts to half page).

CSPmain: ASCII 2 CSP menu has been rearranged to all display (and modification) of the preferred data (output) file name.

MacState3D: That incessant beep at the end of file is gone. It politely beeps once.

IBM: No updates.

Cray/Sun No external updates. An option for NQF processing (vice background) can be toggled by the "=" key from the main menu.

December 3, 1992

Restart problem in embed fixed.

Improved support for file transfers between platforms with csp2xfr and xfr2csp.

QCRtrim allows for user supplied bin for cutoff.

MAC:

Files in a folder other than the CSP folder can be correctly accessed by the plotting routines. The files needed by the processors and the utilities must still be in the same folder as the CSP system.

September 17, 1992

Version 2.2.0 (interim) released.

Version 2.1 no longer official release and will not be supported.

As of this date, the version 2.2 file structure is implemented. Existing v2.1 files are not compatible with v2.2. If needed, a conversion utility "v21tov22" is available.

THE MACINTOSH SYSTEM MANAGER

The Macintosh CSP system has a graphical user interface (GUI) that provides push button access to most CSP functions. CSPmain is the application launched by the user. It performs utility functions such as ASCII conversion, controls displays of signals, attractors, and plot files, and, most importantly, puts the in-core modules into background execution. Menus called up by the push buttons are used to set the run parameters. A description of the Macintosh specific functions follows.

DISPLAY ATTRACTOR

This function displays a three dimension attractor in a perspective plot. If the fractal dimension of the attractor is much greater than 3.0, the object still exists, but displaying it is an exercise in futility.

The scale parameter in the menu describes how far from the attractor the viewer is located. This allows sizing of the attractor in the display window. The value shown is the default setting for the midpoint of the slide control in the window. Larger values move the observer away from the attractor, making it appear smaller. If adjusting the lower left slide bar does not provide satisfactory sizing, return to this menu (select Parameters from the menu bar) and increase or decrease the value.

The top slide bar controls the speed of the trajectory.

The bottom right slide bar controls the vertical positioning of the display and the bottom left bar is the distance to the object.

On the menu bar, restart clears the screen and begins the display at the start of the data file. The parameters control either returns you to the main parameter selection menu (which is needed less now than in early versions that had no sliders), or allows you to change the size of the red tail. The "show" item in parameters does nothing in this version.

The file item, following Apple guidelines, allows selection of a new file, opening and closing the window, and quitting the program. The "go away" button on the window also closes the window.

SIGNAL TRACE

Signal trace plots a signal and shows the FFT. On its menu, the FFT size is either set to 16K or, if there are less than 16K points in the file, to the largest FFT that can be done.

The compression factor controls how many points are averaged to show on the screen. This allows either point by point display, or a more panoramic view. It has no effect on the FFT size. The scale is the vertical scale and may be useful when the scale factor is very big.

Selecting the FFT item from the Control menu bar item executes the FFT. The first data point used is the one at the far left of the signal trace

screen. If FFT fails to execute, it means (usually) that there is not enough memory in the partition for the program. Exit CSP and "get info" for the application "MacSignal." Increase the partition size and try again.

When the FFT window pops up, the bottom slide bar moves you through the FFT (which can be much larger than the window) and the vertical slide controls how many points are averaged and displayed together (compressed). When the window first comes up, the entire FFT is displayed and the compression is at the maximum.

Freeze on the Control bar, stops the dynamic display of the signal trace.

PLOT RESULTS

The menu for this function shows CSP plot files for false neighbor embedding, correlation dimension, mutual information, and global Lyapunov exponents. Selecting a file activates the appropriate plot function.

When the first screen for the fractal dimension is displayed, the user may use the mouse to select any horizontal point (i.e. ball radius) for the final computation of log/log slopes -- the fractal dimension. When you click the mouse for the first time, another screen will appear. This screen shows the final estimate of d_A for the point you selected. A good location is the smallest radius (x axis) where the slopes do not exceed the volume (a physical impossibility) and where the slopes do not regress. This second condition manifests itself where the curves fold down over themselves.

Select "next" when you are ready to go to the final screen. We recommend that the trimming tool (in the convert menu push "Trim CFD") be used before plotting fractal dimension results. This eliminates extraneous data for radii greater than the attractor size.

Clicking the mouse at any point on the mutual information plot will display the corresponding time lag and $I(T)$ of the vertical mouse location (not, the plotted location).

A FINAL MAC NOTE: For the moment, all files must be in the same folder as the CSP system when processing. The plot routines will, however, accept files in any folder.

THE UNIX/DOS SHELL

As an interim, a character based shell is used for the UNIX system. It mimics some of the Macintosh commands and provide a single command driven shell for most of the CSP functions. CSPshell incorporates the "INI" modules, file checking, and the interface to Xgraph and mathematica. To activate the shell, enter "cspshell" as a unix command, and the first menu (such as it is) will come up. The CSP prompts for each function will be listed.

Eventually, this scheme will be augmented with a Mac like system for hosts that support X-windows. Support for CSPshell will continue, however, because it is the only truly machine independent interface. It may be user unfriendly and lack a graphic interface, but it will run on any machine.

IBM/UNIX GRAPHICS

The output graphics are accessed from the main menu. The IBM systems use Randle developed indigenous graphics that provide (varying degrees) of interactive analysis. The UNIX shell has an interface to the xvgr public domain (but copyrighted) graphics system.

The X-Windows tool Xgraph can also be supported by adding "global:xgraph" to the csp.init file.

For IBM and UNIX, the shell calls CSPPLOT, that, in turn, calls the appropriate plotting routine. On the IBM, the sub-calls are to plotmut, plotemb, plotlyp, plotqcr, plotsig, or state3d. Under UNIX, a xgraph file is generated and xgraph is called to display the plot. When you close the xgraph window, control returns to CSPPLOT or CSPSHELL (if it is running).

Plots are provided for:

- Average mutual information
- Embedding dimension
- Correlation dimension
- Lyapunov exponents
- Signals traces and FFTs (Cray only)
- State space reconstruction

When using xvgr, it provides a continuous readout of the mouse position in the x-y coordinates. This is especially useful for determining the best location to estimate correlation dimension.

PLOTSIG (IBM)

Using a data (.dat) file, plotsig shows the trace of a signal. The F key controls the FFT plot. The esc key exits the program.

PLOTSIG compresses up to 8,192 samples into the 512 pixel display

screen. The FFT is taken on all the points, so it shows multiperiod broadband behavior. The FFT may take a few seconds, depending on the hardware -- the screen will be blank.

STATE3D (IBM)

STATE3D displays a pseudo state space of three dimensions. That is, the vector $x(n)$, $x(n + T)$, $x(n + 2T)$ is plotted in a three dimension perspective. This program demonstrates the properties of mutual information -- how much information about other datum is contained in each point.

The user can specify the time lag. Because the signal file is a discrete digital representation of the signal, the time lag is treated as the number of samples between the samples that make up the vector. The user is also prompted for number of samples to skip. This allows the system to ignore start up transients -- i.e., the dynamical system needs sometime to fall to the attractor from the startup vector.

The axes are already rotated to $x = -30^\circ$, $y = 30^\circ$, and $z = -20^\circ$. As the program runs, the user can rotate the axis, change the time lag, reset the scaling factors, and go back to the beginning of the data file.

The commands are:

r rotate all viewing coordinates
x change x axis rotation
y change y axis rotation
z change z axis rotation

≠ Increase forcing amplitude in Duffing programs. Moves display in others.

← Decrease forcing amplitude in Duffing programs. Moves display in others.

∅ Decrease damping in Duffing programs. Moves display in others.

◆ Increase damping in Duffing programs. Moves display in others.

Home Resets display parameters to defaults.

m change time lag and go to start of data file

R go back to start of data file

t change time lag

p switch between portrait and Poincare section

2 switch between a geometric perspective (3D) and a projection (2D)

display

s decrease display scale

S increase display scale

a redisplay axes without clearing the screen
space clear screen and continue

ESC exit

As an example, the signal trace file from a sine wave is included on the distribution media. Plotting this file (or any other) with a time lag of 0 simply produces a line at 45° between the axes. This is not surprising because this is a plot of a point against itself. Increasing the time lag, begins to make the mutual information between data apparent. For the sine wave, it is apparent that any given point allows full knowledge of all other points (try plotting this file with a time lag of 32, for example).

Correlation Dimension Plots (IBM)

The only interactive graphics for the IBM is the correlation dimension plots. PLOTQCR may be run as a stand alone program or it may be accessed from CSPshell or CSPplot. PLOTQCR prompts for a smoothing factor. Ten (10) is a good value.

The first screen shows the counts as a function of radius on the top. The bottom curves are the local slopes of a log-log plot (not shown). The smoothing factor uses a moving average to reduce local fluctuations caused by insufficient data (Grassberger-Procaccia never has sufficient data!). If these curves are too ragged, use "esc" to exit and start again with a larger factor.

The correlation dimension is then found by selecting the smallest radius that has good counts (i.e. smooth slopes) for all dimensions, and plotting those slopes against the embedding dimension.

Pressing "return" uses the default radius of 1/4 of the attractor size, which experience suggests is a good value.

The user may select a radius by using the left and right cursor keys (no mouse support is provided in DOS). As soon as either one of these keys is pressed, a marker appears at the bottom of the screen for the corresponding radius. Using the cursor keys, move the marker to a good region. Using "Shift" cursor will move the marker by ten pixels.

Then, hit return to view the final graph.

THE IN-CORE MODULES

EMBED

EMBED computes the minimum embedding dimension using the method of Kennel, Brown, and Abarbanel (see the footnotes and reading list in the companion piece). The user is prompted for:

The data file name,
the number of samples to use,
the number of samples to skip,
the minimum dimension,
the maximum dimension,
the time delay,
a "group factor",
Tolerance 1,
Tolerance 2.

A minimum dimension of 1 is recommended (and is the minimum allowed) and the maximum dimension should be larger than the expected dimension of the system. These computations are reasonably fast and efficient, so the penalty for overestimating the maximum dimension is negligible.

Occasionally, the EMBED will terminate with an error message that says something to the effect of "nFile overflow - try reducing the number of buckets." This condition typically occurs when there are few data and the group factor is relatively high. Try reducing the "group factor" if this happens.

The time delay can be any number, although we recommend that you use a time lag derived from Average Mutual Information in either three or four dimensions.

The "group factor" is a technical parameter. It defaults to 25. If you have trouble with EMBED and your data are sparse (say less than 10,000 points) try a smaller number (20 often works). If all else fails, let us know and we can activate a special (very nasty) modification that will work with sparse data.

The tolerance should be a number about 15. If the results show that the percent of false neighbors is intuitively too low in the lower dimensions, then reduce the tolerance. If the attractor is very compact (the size is less than 5 or 6 after CSP normalization), try 10 or even 5. The important results are insensitive to this parameter.

Tolerance 2 is normally set to 2.

EMBED output is a CSP file with a type of .emb and a Mathematica notebook file called FILENAME-e.mat. On the Macintosh this file is tagged as a Mathematica file and can be viewed by double clicking on the file.

The results are put into a log file and also can be plotted using

CSPPLOT (on the Mac).

Don't forget to enter a time delay if one has not been provided by the system (i.e. run Fast mut first!).

MUT

The shell prompts for:

the input CSP data file name,
an output file name,
samples to skip,
vector length (which must be a power of 2),
the dimension,
minimum time delay,
maximum time delay,
verbose mode (y/n) {Unix only}
 T_{opt} only.

As with all CSP modules, the entry for input file does not need a file type. MUT expects a file type of .DAT. The system will generate its own file types for the output (and some intermediate files).

Responding with "y" to the verbose mode query will cause the kernel to display intermediate results on the window where the shell was run. "n" puts the kernel into a quiet mode where no output is displayed on the workstation.

The T_{opt} only option causes the calculations to halt after the first local minima is found. The appropriate time lag is reported to the log file, stored in the data file header, and the plot file. The user can then recover this value by examining the log file or running the plotter. Halting at T_{opt} allows for efficient determination of time lag without extra processing. This is especially useful early in a study when some preliminary educated guess is needed (usually done in dimension 3).

Upon exit, the shell puts the kernel into background execution on machines that support background processing. Regardless of the verbose mode, the kernel displays a header on the workstation to let the user know it is executing.

The output is in three files. The log file contains a printout of the results, a .MUT file contains CSP binary plot data. A mathematica file with the name FILE-m.mat also is generated. For workstations that support the CSP plotting routines, the .MUT file can be displayed with the PLOTMUT tool or by selecting "plot results" in the CSPmain shell.

FAMI (Fast Average Mutual Information) prompts for input in a manner similar to the full algorithm. The major difference is that there are no restrictions on the length of the input vector. It does not have to be a power of 2. FAMI is fast enough, that it only runs in the foreground (v2.1). {not on

the Mac, yet}

Apple notes: MUT is accessed from the main program via a menu. MUT is run in the background while the user remains in the shell. MUT is compute intensive and machine response will degrade while it is running. Also, you can ignore the request for an output file name in the menu. MUT can supply the standard file name unless you want something else.

Mutual information calculations can be restarted. If the shell detects a plot file of the same name as the data file, the shell prompts to determine if the user wants a restart. If so, then the calculations resume with the original parameters at the next increment of time.

CFD (Correlation Dimension)

CFD calculates the correlation dimension of a scalar time series. The correlation dimension is functionally equivalent to the fractal dimension. CFD prompts for:

- The data file name,
- minimum and maximum embedding dimensions,
- the number of points to use (the more the better),
- the time delay,
- the number of histogram bins,
- a multiplier (scaling factor) for the data,
- and, a VERBOSE request.

The time delay can be any good guess, although computing average mutual information in three dimensions is a good start. If the dimension is much higher, an other iteration may be in order.

The number of histogram bins can be anything your computer can handle. There is no computation penalty for overestimating (see QCRTrim, below).

The multiplier is used to improve the resolution of results when the originating data were integer. This often occurs with PCM encoded A/D boards, for example.

CFDTrim

As noted above, there is no computation penalty for overestimating the number of bins. But, displaying a large number of "saturated" bins is pointless. By saturation, we refer to those bins where the entire attractor is already encompassed at some smaller radius. Running CFDTrim (the Trim CFD button in the conversion menu on the Mac) eliminates the redundant bins. This can significantly reduce the size of the file and improves the

quality of the useful information on the graphics display.

Lyapunov (now obsolete)

Recommended parameters:

Samples: The more the better. this is application dependent and should reflect a reasonable evolution time.

Number of Jacobians: 2,000 is the default. But, experience with specific data sets may indicate fewer (or more). Try 2,000. Then try a lower number. If the results are identical, try even lower numbers. Then again, 2,000 may not be adequate.

Group Factor: 25 is adequate. A lower number will yield better (perhaps) results, a larger number will run faster and use less memory.

Order of fit. Again, experience will tell. Linear signals only need a first order fit, highly complicated chaotic signals may need five. There is no memory penalty for a higher order fit, but the processing time will increase.

Local Lyapunov exponents

Loclyp is the replacement for the global Lyapunov module. Parameters are:

Data file name: the csp format data file (.dat)

Output file name: the system provides the recommended unique name.

Forward and Reverse: Entering 0 calculates the Lyapunov exponents in both the forward and reverse direction.

samples to use: the more the better. Toy problems can use 20K or so.

points to skip: self explanatory

Time delay: system supplies optimum, if known.

Embedding dimension: From global false nearest neighbors

Local dimension: Must be less than embedding dimension.

Number of Jacobians: must be larger than the longest length (see below).

Group factor: 25

Order of fit: 1 to 4. Four provides best results, but are compute intensive.

Number of neighbors: Try 300, but be prepared to go to 1,500 for real data.

Number of lengths: This goes to the heart of the local nature of the dynamics. The calculations for short lengths provide information about short term predictability over the attractor. Longer lengths approach the global values. Entering 0 is the same as entering 14, which calculates the following values:

1 2 4 8 16 32 64 128 256 512 1024 2048 4096 8192

After entering a positive number at the prompt for the number of lengths, you will be prompted for each of the lengths.

As an unfriendly shortcut, if you enter a negative number at the prompt for the number of lengths, the system will generate a powers of two progression (as shown above for the 0 default) for that many lengths. For example, entering -10, will generate the lengths 1, 2, 4, 8, 32, 64, 128, 256, 512, 1024.

Memory factor: stick with 1 unless you get an error message suggesting a higher value.

Verbose: These runs are long and can produce a lot of verbiage. Recommend n for no unless you are a sadist.

PROBABILISTIC SCALED CLEANING

PSC (Probabilistic Scaled Cleaning) is a noise reduction system. It is supplied separately and is not part of the general CSP package.

THE TOOLKIT

3D2BYU (Cray only)

3D2BYU converts three separate .dat files into an MPGS file (BYU format). Typically, the three files will be signal data from different dimensions for the same system. For example, MAKEDUF generates three files, duf_a.dat, duf_v.dat, and duf_d.dat that contain the signals for each of the variables in Duffing's oscillator. A true portrait of the system can be formed by using 3D2BYU. 3D2BYU is used to visualize low dimension attractors when data from all dimensions is available.

3D2CSP

3D2CSP converts a "flat" ASCII file of dimension 3 to CSP format. The points are read in as one from each dimension. So, the input file could have the form:

```
x[1,1] x[2,1] x[3,1]
x[1,2] x[2,2] x[3,2]
```

```
.
.
.
```

3D2CSP is used primarily to convert data files generated by other systems and is of little interest to the general user. Randle, Inc. uses it to work with certain test files provided by our scientific team.

ADDCHAOS { NOT SUPPLIED }

ADDCHAOS superimposes one file on top of another, at some specified SNR. It is typically used to add chaotic contamination to other files, although any signal file can be used. It prompts for the name of the file to be contaminated and the other file. A new file, *nameSNR.DAT* is generated.

ADDNOISE

ADDNOISE adds uniform white noise to a CSP data file and produces a new "noisy" file. It prompts for the name of the file to contaminate and then generates a new noisy data set that it places into *nameSNR.DAT*.

{ Mac Note -- the menu remains up while this is done -- it may take a few moments. The menu drops when the operation is finished. }

ASC2CSP

Unix - From CSPShell.

MAC - Accessed from the convert button in CSPmain.

ASC2CSP is the primary method of importing data into CSP. It accepts an ASCII file and generates a CSP data file (*name.dat*). The file name of the ASCII file is used for the comment field in the CSP header.

The Unix version will only accept a "flat" ASCII file where the numbers are delimited by a white space or return.

The Mac interface (via CSPmain) provides greater flexibility, including selection of delimiting characters and column. Alternatively, the user may skip a specified number of characters or strings. A single line header also may be skipped over. The menu displays the first two lines of the input file so that the appropriate selections can be made.

ATTMOVIE (Cray specific)

Attmovie is used to generate an MPGS "movie" of an attractor unfolding as the time lag is increased.

CHKDATA

Unix - From CSPShell.

Mac - function is pushbutton in CSPmain

CHKDATA checks CSP files to insure that the information in the header matches the data. It checks for maximum value, minimum value, number of points, and other parameters. If an error is found, it fixes the header.

CSP2MPGS (Cray specific)

This module is the interface to Cray's proprietary Multi Purpose Graphics System.

CSP2WAVFM

This module is the interface to the Waveform graphics system, for system that support Waveform (Silicon Graphics).

WAV2-

The WAV format is used by some A/D D/A boards and software vendors to store low dynamic range data. WAV2- will convert a WAV file to CSP format or to an ASCII file.

MPGS (Cray specific)

MPGS (Multipurpose Graphics System) is a Cray Research proprietary graphics system that runs on a Silicon Graphics workstation. A separate license for MPGS must be secured from Cray Research.

XGRAPH (Unix only)

XGRAPH is the X-Windows utility that plots X-Y data. XGRAPH is supplied as part of X-Windows, and is not available from Randle, Inc. Our toolkit provides conversion routines so that, with the appropriate workstation, the results of EMBED (future), MUT, and CFD can be plotted. Another utility allows two dimension maps to be created from .DAT files. Files for XGRAPH have the extension of .XGF.

SIGNAL SIMULATORS

A signal simulation package provides a wide variety of signals for training. Your system may not contain all of the modules.

MAKEDUF

Generates CSP data files of the three independent variables in Duffing's oscillator. Prompts for magnitude of forcing function, linear damping coefficient, and "stiffness." Interesting standard values are shown on initiation.

MAKELOR

Lorenz system.

MAKEROS

Rosler system.

MAKESIGS

Generates a sine wave, random data and a nonlinear (but not chaotic) signal (a bispectral signal)s\.

MODULATE

Modulated AM and FM signals.

CLIPM

Clipped AM & FM signals.

PSK

Generates a spread spectrum signal where chip length and phase rotation can be manipulated.

MODULE LISTING

NAME	ARGUMENTS	FUNCTION	REMARKS
3D2BYU		Converts three separate .DAT files into a MPGS file	Cray specific
3D2CSP		Convert a flat ASCII n-dimension file to .DAT	Not Supplied (NS)
ADDCHAOS		Adds chaotic contamination to a .DAT file	
ADDNOISE		Adds uniform white noise to a .DAT file	In shell
ASC2CSP		Converts a flat ASCII data file to a .DAT file	In shell
ATTMOVIE		Generates an MPGS movie of an attractor unfolding	Cray specific
CFD		Computes fractal dimension	Kernel
CHKDATA		Checks files	In shell
CHKWAV		Checks WAV files	
CLIPM		Generates demo data files of clipped AM & FM signals	
CORR		Autocorrelation	Not supported
CRAYFFT		FFT	Cray only
CSPmain		Macintosh main menu	Apple only
CSPplot	FILENAME(.TYPE) optional	Unix/DOS plot menu (also called by CSPshell)	Unix/DOS
CSPshell		Unix/DOS main menu	Unix/Dos
CSP2MPGS		Converts a .DAT file to MPGS or BYU format.	Cray specific
CSP2PART		Builds a MPGS particle trace file	Cray specific
CSP2VEC		Builds a MPGS vector file	Cray specific
CSP2WAV		Converts a .DAT file to .WAV format	
CSP2XFR	FILENAME(.TYPE)	Transfer CSP files	In shell
DETECT		Special MI file. Not part of general CSP package.	NS
DUFFING		Demonstration of Duffing's Oscillator	IBM/DOS only
DUFTRC		Duffing's oscillator signal trace	IBM/DOS only
DUFTRCA			IBM/DOS only
EMBED		Computes embedding dimension	Kernel
FAMI		Computes AMI using fast method for 2 dimensions (only)	
FFT2BYU		Generates an MPGS BYU file of a signal FFT	Cray specific
HSIG		Displays signal and FFT	IBM/DOS only (see MacSignal)

LORENZ		Demonstrates Lorenz system	IBM/PC only
MACSIGNAL		Displays signal and FFT	Macintosh only
MACCSPPlot		Plot results	Macintosh only
MACSTATE3D		State space reconstruction of a signal	Macintosh only
MAKEDUF		Generates .DAT files for Duffing's oscillator	
MALELOR		Generates .DAT files for Lorenz system	
MAKEROS		Generates .DAT files for Rossler system	
MAKESIGS		Generates .DAT files of a sine wave, a bispectral signal, and random numbers	
MUTBACK		Mutual Information processor	Kernel
PLOTEMB	FILENAME	Plots embedding dimension (.emb)	DOS only
PLOTCFD	FILENAME	Plots correlation dimension (.qcr)	Dos only
PLOTLYP	FILENAME	Plots Lyapunov exponents	Dos only
PLOTMUT	FILENAME	Plots mutual information curves	IBM/DOS only
PLOTSIG	FILENAME	Older version of HSIg	IBM/DOS only
PLOTCFD	FILENAME	Plots fractal dimension curves	IBM/DOS only
PSC		Noise reduction	Kernel
ROSSLER		Demonstrates Rossler system	IBM/DOS only
SIG2MPGS		Converts a .DAT file to MPGS format	Cray specific
STATE3D	FILENAME (.dat supplied)	Displays a reconstructed attractor Called by CSPshell/CSPplot	IBM/DOS only (see MacState3d)
XGRAPH	Xgraph data file name	Plots Xgraph formatted files Called by CSPshell/CSPplot	X-Windows utility
XVGR	XVGR graphing system	Plots xvgr files	

Modules

FUNCTION	MODULE
EMBED	Global minimum embedding dimension
EMBEDL	Local minimum embedding dimension
MUTBACK	Average Mutual Information
MUTX	Mutual Information between time series
CFD	Fractal Dimension
LOCLYP	Lyapunov Exponents (local and global)
FAMI	Fast Mutual Information
PSC	Noise reduction

Data file generators

CLIPM	Clipped AM & FM linear signals	ammodc.dat fmmodc.dat
MAKEDUF	Generates signals for Duffing's oscillator	duf_a.dat duf_v.dat duf_d.dat
MAKELOR	Generates signals for the Lorenz system	lor_1.dat lor_2.dat lor_3.dat
MAKEROS	Generates signals from the Rossler system	ros_1.dat ros_2.dat ros_3.dat
MAKESAR	Generates Synthetic Aperture Radar I&Q pairs for a single range bin. Supplied separately.	sari.dat sarq.dat
MAKESIGS	Generates a sine wave, a bispectral signal and random data.	sine.dat bispec.dat rand.dat
MODULATE	Generates AM & FM modulated sine waves (see also CLIPM).	ammod.dat fmmod.dat
PSK	Generates a PSK spread spectrum signal. Supplied separately.	psk.dat
FSK	Generates a FSK signal	fsk.dat

FILE TYPES

TYPE	USED BY	REMARKS
.DAT		basic data file
.MUT	AMI results	AMI file
.QCR	QCR1 results	
.CFD	CFD results	
.LYP	LYP results	
.EMB	Embed results	
.WAV	WAV2***	.wav files
.MPG	MPGS	Cray only
.BYU	MPGS	Cray only
.PRT	MPGS	Cray only
.XFR	file transfer	
.MAT	MATHEMATICA	(obsolete)
.XGF	XGRAPH	generic plot file
.XGR	xvgr	generic plot file