

## **GPP Version 6 User's Guide**

### **A GENERAL-PURPOSE POSTPROCESSOR FOR WIND-TURBINE DATA ANALYSIS**

Marshall L. Buhl, Jr.  
*National Renewable Energy Laboratory*

Norman L. Weaver  
*InterWeaver Consulting*



**NREL**

**National Renewable Energy Laboratory**

1617 Cole Boulevard  
Golden, Colorado 80401-3393

NREL is a U.S. Department of Energy Laboratory  
Operated by Midwest Research Institute • Battelle • Bechtel

Contract No. DE-AC36-98-GO10337

# **GPP Version 6 User's Guide**

## **A GENERAL-PURPOSE POSTPROCESSOR FOR WIND-TURBINE DATA ANALYSIS**

Marshall L. Buhl, Jr.  
*National Renewable Energy Laboratory*

Norman L. Weaver  
*InterWeaver Consulting*

Prepared under Task No. WE901210



**NREL**

**National Renewable Energy Laboratory**

1617 Cole Boulevard  
Golden, Colorado 80401-3393

NREL is a U.S. Department of Energy Laboratory  
Operated by Midwest Research Institute • Battelle • Bechtel

Contract No. DE-AC36-98-GO10337

## NOTICE

This report was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or any agency thereof.

Available to DOE and DOE contractors from:  
Office of Scientific and Technical Information (OSTI)  
P.O. Box 62  
Oak Ridge, TN 37831  
Prices available by calling 423-576-8401

Available to the public from:  
National Technical Information Service (NTIS)  
U.S. Department of Commerce  
5285 Port Royal Road  
Springfield, VA 22161  
703-605-6000 or 800-553-6847  
or  
DOE Information Bridge  
<http://www.doe.gov/bridge/home.html>



Printed on paper containing at least 50% wastepaper, including 20% postconsumer waste

## Notice

We distribute GPP with its source code. Feel free to change the code if you like, but please document your changes in the `CHANGE.LOG` file as you make them. If you distribute a modified version of GPP, please modify the startup screen to warn the user that (s)he is using a modified version of the program. NREL seeks neither credit nor blame for your changes.

## Disclaimer

The software described in this user's guide is distributed on an "as is" basis. It is for evaluation purposes only. We encourage you to notify us of any problems you have with GPP. We also welcome any ideas and suggestions you may have for improving the product. Bug fixes and improvements are made on a time-available basis only and we cannot promise speedy resolution of problems you may encounter.

## Trademark Acknowledgments

All terms mentioned in this report that we know are trademarks or service marks have been appropriately capitalized. We do not attest to the accuracy of this information. Use of a term in this report should not be regarded as affecting the validity of any trademark or service mark.

ADAMS is a registered trademark of Mechanical Dynamics, Inc.

*BLADED* is a trademark of Garrad Hassan and Partners Limited.

Microsoft, MS, MS-DOS, Windows, Windows 95, Windows 98, and Windows NT are trademarks of Microsoft Corporation.

Silicon Graphics is a registered trademark of Silicon Graphics, Inc.

Sun is a trademark of Sun Microsystems, Inc.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Ltd.

## Acknowledgments

In updating GPP to this version, thanks are in order to several individuals. Special acknowledgment goes to Dave Simms whose early suggestions for GPP significantly defined its final form as a broadly useful software tool rather than a simple ADAMS postprocessor. Thanks to Neil Kelley for general review and guidance in the development of early versions of GPP and particularly for providing the original approach for rainflow-counting and power spectral density (PSD) calculations.

On this topic, we must acknowledge the contribution by Robert Lackman of NCAR for use of his sophisticated SPECFT routine for PSD calculations. It definitely makes the PSD Tool much more versatile than the alternative routines available, such as those in *Numerical Recipes* (Press, et al. 1996).

Alan Wright came up with the requirements for the Azimuth-Average Tool and helped us debug the Fourier coefficient part of the code. Alan also found other bugs while using the code.

Additional credits go to Dave Laino and George Scott for helping test, debug, and suggest improvements to the code.

Acknowledgment is also due to James Van Buskirk for allowing us to use a modified version of his arithmetic expression parser to implement calculated data channels. GPP also uses public domain code by John H. Mathews (Mathews, 1992) to implement cubic spline and least-squares curve fitting.

This user's guide was prepared to document work done at the National Renewable Energy Laboratory in support of the U. S. Department of Energy under contract number DE-AC36-98-GO10337.

MLB, NLW

# Table of Contents

<b>How to Use This Manual .....</b>	<b>vii</b>
GENERAL CONVENTIONS .....	vii
KEYBOARD CONVENTIONS .....	vii
<b>1 Introduction.....</b>	<b>1-1</b>
ABSTRACT .....	1-1
BACKGROUND .....	1-1
WHAT'S NEW IN GPP VERSION 6? .....	1-2
REQUIREMENTS FOR RUNNING GPP .....	1-3
<b>2 Installing GPP .....</b>	<b>2-1</b>
OBTAINING A COPY OF GPP .....	2-1
Windows-Based Personal Computers.....	2-1
UNIX-Based Workstations.....	2-1
Confirming Correct Operation of GPP .....	2-1
Porting to an Unsupported Computer:.....	2-2
RUNNING GPP .....	2-4
Personal Computers.....	2-4
UNIX-Based Workstations.....	2-4
<b>3 Using GPP .....</b>	<b>3-1</b>
BACKGROUND .....	3-1
BATCH PROCESSING.....	3-2
THE MENUS .....	3-2
The Startup Screen .....	3-2
The Tool Menu .....	3-3
The Merge-Tool Menu .....	3-3
ANALYSIS TOOLS.....	3-4
The Azimuth-Average Tool.....	3-5
The Bin Tool .....	3-6
The Histogram Tool.....	3-6
The Interpolation Tool.....	3-7
The Least-Squares-Fit Tool.....	3-8
The PSD Tool.....	3-8
The Rainflow-Cycle-Count Tool.....	3-10
The Statistics Tool .....	3-12
FILTER TOOLS.....	3-13
The Filter Tool .....	3-13
The Limit Tool.....	3-13
The Merge Tool .....	3-14
The Trim Tool .....	3-15
MISCELLANEOUS TOOLS .....	3-16
The Calculated-Channel Tool.....	3-16
The GenTime Tool.....	3-16
The Trim Tool .....	3-17

The Units-Conversion Tool .....	3-17
SYSTEM TOOLS .....	3-18
The Choose Tool .....	3-18
The Directory Tool .....	3-18
The Divide Tool.....	3-18
The Escape Tool.....	3-19
The Get-Columns Tool.....	3-19
The Help Tool .....	3-19
The More Tool.....	3-19
The Options Tool .....	3-20
The Write Tool .....	3-20
QUITTING GPP .....	3-20

## References

## Glossary

## Index

# How to Use This Manual

## GENERAL CONVENTIONS

We use different typefaces throughout this manual to help distinguish different concepts. The majority of the body text (including this paragraph) is set in 11 point Times. Headings use various sizes of the Helvetica typeface. Within body text, we also use Helvetica to denote the names of files and other things. When a file name applies to both DOS and UNIX, we will put the name in lower case so that it will work for both operating systems. This manual uses bold Letter Gothic for commands that you enter exactly as they appear. These lines are indented one-quarter inch and are set off from the body text by blank lines. We show screen examples and file excerpts in bold roman Letter Gothic type and usually border the screen examples with a rounded box to denote a screen.

## KEYBOARD CONVENTIONS

When entering commands at the keyboard, enter text exactly as printed. If you are using a DOS- or Windows-based personal computer, you may use either upper- or lower-case letters. If you are using a UNIX-based workstation, you must be sure to duplicate the case exactly.

“Naked return,” a term used occasionally in this manual, means that you press the Enter key (↵) without typing any previous text. Use naked returns to select default answers or to display the long versions of high-level menus when GPP prompts you with short (one-line) menus.

If at any time you find that you need to back out of a menu or subtool, typing the “~” character followed by ↵ will cancel the current menu or tool step and return you to the previous menu.

In addition, when GPP is reading its commands from a file, any line with the “#” character in the first column is treated as a comment line. GPP skips to the next line to continue processing.



# 1 Introduction

## ABSTRACT

GPP (pronounced “jeep”) is a general-purpose postprocessor for wind turbine data analysis. The original author, Marshall Buhl, a member of the National Wind Technology Center (NWTC) of the National Renewable Energy Laboratory (NREL), developed GPP to postprocess test data and simulation predictions. Norman Weaver of InterWeaver Consulting is responsible for coding most of the recent additions to Version 6. GPP reads data into large arrays and allows the user to run many types of analyses on the data stored in memory. It runs on inexpensive computers common in the wind industry. One can even use it on a laptop in the field.

We wrote the program in such a way as to make it easy to add new types of analyses and to port it to many types of computers. Although GPP is very powerful and feature rich, it is still very easy to learn and easy to use. Exhaustive error trapping prevents one from losing valuable work caused by input errors. We believe that GPP can make a significant impact on engineering productivity in the wind industry.

## BACKGROUND

GPP started out as a loose collection of utility programs. As the number of utilities grew, duplication among the programs became excessive. The solution seemed to be to write one program that would input the data and then manipulate that data in a variety of ways. The first version of GPP included only the original tools and ran only on Silicon Graphics, Inc., (SGI) computers. Since then, others have made requests for many additional features, including the ability to run on a variety of platforms.

To make the code platform independent, we put all compiler-specific routines into a separate file so that the majority of the code would require no modifications to run on other types of platforms. The compiler-specific routines include all operating system calls and all screen output.

As of this writing, GPP Version 6 has been run on Sun Microsystems UNIX computers and on personal computers (PCs). We developed recent additions to GPP as a "console" application using DIGITAL Visual Fortran for Microsoft Windows. The current version continues to segregate system dependent code so we believe that porting to a new system supporting Fortran 90 should still be less than a day's effort.

GPP offers engineers a user-friendly tool that makes it easy to look at data in many different ways. Experience has shown that researchers need to analyze data with varied techniques. For example, a power spectral density (PSD) plot that compares test data to simulation predictions may show excellent agreement, while rainflow cycle-count spectra tell an entirely different story.

GPP is a menu-driven, question-and-answer type program instead of a batch-type program. However, one can use GPP in a batch mode - see **Batch Processing** on page 3-2. It is a character-based program instead of one with a graphical user interface (GUI) for several reasons. Although GUI programs are

pretty, they are more difficult to code, run slower than character-based programs, and are often non-portable. Platform-independent GUI development tools are becoming practical, so we may someday create a visual interface for GPP. The portability requirement has also made it too difficult, in addition to costly, to add charting capabilities. However, GPP's output files import easily into standard graphics packages such as Microsoft Excel.

Because GPP reads entire data sets into arrays, it is *very* fast. Once the data is in memory, a modern PC can perform many analyses in just seconds. Older analysis tools had severe memory restrictions because of the limitations of the technology that was available when they were written. Because these older programs had to store information on disk, they were very slow.

In GPP, many prompts have intelligent defaults and the less-obvious questions have context-sensitive help. In some cases, there are on-line examples to help users. We gave usability a high priority and we hope that we have made it "fool resistant." For example, GPP traps input errors and requests the input again.

The targeted users for GPP are engineers in wind energy research and the wind industry. The program assumes that users are familiar with the concepts and techniques that GPP implements.

## WHAT'S NEW IN GPP VERSION 6?

- This version employs Fortran 90 dynamic array sizing.
- We fixed a Y2K-type problem in the handling of date strings.
- We developed a testing protocol script ("GPPTTEST") for both DOS and UNIX.
- The path separator is now OS independent.
- The Options Tool has a number of updated options.
- Version 6 will automatically recognize FAST\_AD, ADAS, and *BLADED* data file types.
- Version 6 attempts to automatically interpret header records for "generic" files.
- The Choose Tool now has additional flexibility for column selection.
- A synthesized time column option has been added to the Merge Tool.
- We have added new bin statistics to the Azimuth-Average Tool.
- GPP Version 6 uses rainflow-cycle counting algorithms from Sandia National Laboratories' (SNL) LIFE2 program.
- This version introduces a general menu escape character.
- We have added a Calculated-Channel Tool to this version.
- The Rainflow-Cycle-Count Tool now includes independent, two-dimension binning and bin-area normalization.
- This version significantly increases the speed of data file input.
- *Numerical Recipe* routines have been replaced with equivalent public domain algorithms.
- The Filter Tool now includes an infinite impulse response (IIR) high/low-pass filter algorithm.

**REQUIREMENTS FOR RUNNING GPP**

GPP runs on all Windows 95, Windows 98, or Windows NT systems. It will also run unmodified on Sun workstations, and on other UNIX-based workstations with minor modifications to the Fortran 90 source code. GPP needs enough virtual memory to hold your data (4 bytes per data value) and only a few megabytes of disk storage. GPP runs extremely fast on any modern computer.

# 2 Installing GPP

## OBTAINING A COPY OF GPP

You may download the latest versions of GPP and this user's manual from the NWTC Design Codes web page. We provide download instructions on the web server. If you do not know the current location of this web site, send email to [marshall\\_buhl@nrel.gov](mailto:marshall_buhl@nrel.gov). See the README file for installation instructions.

You may want to make some changes to GPP to tailor the program to your needs. GPP Version 6 dynamically allocates space for working arrays. Currently, we distribute GPP with upper limits on file size set to 1000 columns and 500,000 rows of data. To change the parameters controlling these limits, modify `MAX_COLS` and `MAX_ROWS` as needed in `gpp_mod.for` and recompile the program.

## Windows-Based Personal Computers

By this point, you should have downloaded the appropriate archive from the NWTC Design Codes web site and unarchived it. Under most circumstances, you will use GPP by running the `GPP.EXE` file. If this is the case, skip ahead to the next section, "Running GPP."

If you desire to make source-code changes, and you have the Digital Visual Fortran (DVF) or other Fortran 90 compiler; you can recompile GPP with the supplied files. With the DVF compiler, you will need to place the source-code files in a directory and create a DVF project that includes all \*.FOR files including `SYS_PCD.FOR`. For other compilers, you may need to change the routines in `SYS_PCD.FOR` to port the code to your new compiler. See [Porting to an Unsupported Computer](#) on page 2-2 for advice on modifying a `SYS_???.FOR` file. If GPP does not function properly without changes to the other files, then GPP is not quite as portable as we would like. Please let us know of any required changes to other source files. Our contact information is given on our web site.

## UNIX-Based Workstations

This section applies to owners of supported workstations. If you are not using a Sun workstation, you should skip to [Porting to an Unsupported Computer](#) on page 2-2 for advice. If you want to make changes to GPP, you will need a Fortran 90 compiler installed on your workstation. If your workstation does not have a Fortran 90 compiler, you can install and compile GPP on another workstation with the same operating system (OS) that does have a compiler. Then copy the executable (`gpp`) and help file (`helpfile.txt`) to your machine.

For Sun workstations, you should not need to make significant changes to the `makefile`, but you may want to adjust the compiler flags. The `makefile` contains lines for both debugging and optimization. Just comment out the unwanted `FFLAGS` line with a pound sign (#).

## Confirming Correct Operation of GPP

If you anticipate making changes to GPP, or would like to confirm correct operation of the program, an acceptance test protocol is available in subdirectory `GPPTTEST`. For Windows-based systems, the DOS

batch file GPPTTEST.BAT causes GPP to execute a series of command input files which exercise various program options. The testing sequence compares GPP outputs to reference files (residing in subdirectory GPPTTEST\TSTFILES) for each test. The script places results from a DOS "file compare" into the file GPPTTEST.OUT. At the end of the test, this file is displayed on the screen by the Windows NotePad editor. With normal completion of this test, the only differences that should appear in the GPPTTEST.OUT file are lines containing a different version number or time stamp. (Also, see the README file in the GPPTTEST directory.)

For UNIX systems, we provide a shell script to perform the same sequence of tests. The contents of `gpptest.out` are displayed using UNIX vi editor.

### Porting to an Unsupported Computer:

If you want to compile GPP with an unsupported computer or compiler, you will probably have to modify the routines in one of the `sys_*` source files. You should copy that file and give the new one a similar name with the letters after the “\_” denoting the platform and/or compiler. To use this file, you should modify the makefile, batch, or script files to account for the name change. If you have to make changes to other source files to port GPP, please let us know so that we can make the code even more portable.

Within the `sys_*` file, you will likely need to make changes to many routines. These routines are described below.

DIR	Unless you are using an operating system other than DOS or UNIX, you should not need to change DIR. Just use the appropriate one.
FILESIZE	This routine calls the routine FSTAT for both DOS and UNIX to report the size of the requested file.
FLUSHOUT	This routine flushes the screen buffer so informational messages do not need to wait for the buffer to fill before the computer displays them on the screen. Some compilers use the FLUSH function and others write an end of file to the appropriate unit with ENDFILE. FLUSHOUT may need to be changed, or, in the case of HP-UX, you may need to call a system routine from OPEN_CON to set the buffer size of standard output to zero.
GET_DIR	GET_DIR calls routine GETCWD to obtain the path to the current working directory.
GET_ENV	This routine examines the OS's environment to get the value of one of its variables. Some compilers supply a GETENV (without the “_”) routine for this purpose. Otherwise, you will need to issue some OS commands with SYS_CALL to put a copy of the environment variables into a file, and then read the file to find the requested variable. See the supplied file SYS_PCL.FOR for an example of this method. If your OS does not use environment variables, you may have to make some major kludges.
GET_SYS	This routine sets some system-specific information. It must supply GPP with the name of the user's command shell (for example, <code>csh</code> , <code>CMD.EXE</code> , or <code>COMMAND.COM</code> ). It also sets the maximum length of a file name (without an extension) and the path separator such as “\” for DOS, or “/” for UNIX. It should set a three-letter mnemonic to designate the name of the platform and/or compiler (probably the same as the second part of the <code>sys_*</code> file name).

---

GETONAME	GPP uses this routine to get the name of the user's options file. It must begin with the path separator. Because so many old Windows programs used INI files, we chose \GPP.INI for PCs. The C shell under UNIX usually uses rc (run command) files for this sort of thing, so we use /.gpprc for UNIX systems. Use whatever file name is appropriate for your system.
IS_NAN	Call to "is-not-a-number" routine to check for a valid result from numerical calculations. For Digital Visual Fortran and the SUN Fortran 90 compiler, this routine is IS_NAN and IR_ISNAN, respectively.
OPEN_CON	For some compilers, GPP re-opens standard output (the display), so that it uses the proper options for displaying formatted text to the screen. You must use options that allow you to generate prompts not followed by new lines. Selected options must also allow new text to be appended to the end of an existing line. This routine is not necessary for some compilers. If you do not need it, just eliminate all executable statements but the RETURN. This is also where you should put the call to the routine that sets the display's buffer size to zero, if necessary. See FLUSHOUT for more information.
READ_UAE	This is the binary file read routine for "Unsteady Aerodynamics Experiment" data files. Under Digital Visual Fortran, the file is opened as a "BINARY" type file with data read directly into an array. Under UNIX, the file is opened as an unformatted direct access file. Four-character bytes are read and equivalenced into the output data array.
SYS_CALL	GPP executes OS commands with this routine. Most compilers supply a routine called SYSTEM for this function.
TELLHOME	This routine gives you advice on how to set the HOME directory. You do not need it for UNIX, because the HOME directory is set by default. PC users (and perhaps users of other OSes) may need to set it themselves.
USRALARM	You probably do not need to change this routine, but if your system has advanced sound features, you may want to take advantage of them. GPP currently uses the ASCII "bell" character (^G) to sound the alarm.
WR_BL	This routine writes out the beginning of a multi-part line to the screen. If you are using standard Fortran carriage control for the screen (where the first character means something special like form feed), be sure to prepend a space to the string.
WR_EL	To terminate a multi-part string and generate a new line, GPP calls WR_EL. If you modify this routine, be sure to tell the compiler to append the new string to the end of the previous line. Fortran carriage control uses a prepended ampersand ("&") to append text.
WR_ML	This routine is similar to WR_EL, but it does not follow up with a new line.
WR_NR	GPP uses this routine to write a string to the screen without generating a new line (no return) afterward.
WR_SCR	This routine writes one line of text to the screen. Long lines are broken into multiple lines with indents.

## **RUNNING GPP**

### **Personal Computers**

DOS:

To make GPP easy to run on a DOS system (or in a Command-Prompt window under Windows), add the GPP directory to the search path in your AUTOEXEC.BAT file. Alternatively, you may move the executable and HELPFILE.TXT to a directory that is already in the path. (Under Windows NT, instead of using AUTOEXEC.BAT, update the path environment variable in the environment tab of the system control panel.) Alternatively, you can create a batch file (call it GPP.BAT) to start GPP and put this batch file in one of your searched directories.

After doing the above, just type the following to run GPP:

```
GPP ↵
```

Windows START Menu:

Under Windows, you may also choose to add GPP as an item in your Start/Programs menu. To do so you open your START button folder (right click on the START button and click on OPEN) and add a shortcut to GPP.EXE in the PROGRAMS folder. With GPP appearing in the Start/Programs listing, you can simply click this item to launch GPP.

### **UNIX-Based Workstations**

If you correctly set up the symbolic links in the setup step, you should be able to start GPP from any directory with:

```
gpp ↵
```

# 3 Using GPP

## BACKGROUND

Once you understand the working philosophy of GPP, you should be able to use it with little effort. We have tried to make GPP as user-friendly as possible; it contains much on-line, context-sensitive help and it's good at trapping input errors. To aid your efforts to learn GPP, we have included a glossary of the terms used in the manual and in the program. You can find it on Page xi near the back of this manual.

GPP uses a combination of menus and a question-and-answer coding style. At higher levels, you choose tools from various menus to tell GPP what tasks to perform. Once GPP finishes processing the data with the specified tool, it returns to an appropriate menu. GPP removes tools that are not suitable in a given situation from the menus to avoid confusion. For example, it is meaningless to use the **Azimuth-Average Tool** option before GPP has read in some data.

GPP reads ASCII data files with white space, tab, or comma-separated columns that represent different channels of data. It also recognizes several specific types of data files and can modify its behavior to streamline their use. The code currently recognizes ADAMS, YawDyn, FAST\_AD, Zond's Advanced Data Acquisition System (ADAS), Unsteady Aerodynamics Experiment (UAE), Garrad Hassan's *BLADED*, and GPP Merge Tool output files. When GPP reads one of these special files, it does not need to inquire about headers and columns. It also parses out column headings so the various tools can display information based on column name rather than column number. GPP classifies files that it does not recognize as "generic."

You would rarely change some tool options, so GPP saves them in a permanent file. You can change these options from any of the high-level menus. Examples of these options are the identifying strings for recognized file types, such as ADAMS and YawDyn files. See the subsection on the **Options Tool** on page 3-20 for more information on the options.

***It is important that you understand the different purposes for the storage arrays.***

GPP uses two storage areas for data—a primary array and a secondary array. When working with the primary array, you use tools that (generally) convert time-series data to another domain. These tools write their results to external files. An example of a primary array tool is the **PSD** (power spectral density) **Tool**. This tool converts from the time domain to the frequency domain. Another is the **Rainflow-Cycle-Count Tool**, which generates cycle counts versus cycle amplitudes and/or means. The one tool that does not change domains is the **Interpolation Tool**. The **Tool Menu** section on page 3-3 provides a comprehensive list of tools that process the primary array.

The **Merge Tool** builds up the secondary array by merging columns of data from one or more primary arrays or by adding calculated data channels using the **Calculated Channel Tool**. All the tools available from the **Merge-Tool Menu** leave the data in the time domain, although some leave a series of blocks of time instead of a steadily increasing time. This is where GPP does its filtering, including decimation, low-pass filtering, and cycle limiting. The **Merge-Tool Menu** section on page 3-3 provides a comprehensive list of tools that process the secondary array. When you finish building and filtering the



secondary array, you can move it to the primary array and/or write it to an external file. You can read these external files into the primary array later for additional processing.

When GPP presents you with a list of choices (enclosed in parentheses) for a menu, the one prefixed with an equal sign is the default response. Entering a naked return (↵ with no other text) will select that option. Sometimes GPP will ask you to enter a number (or series of numbers). In this case, GPP will enclose the default response in square brackets [default]. Except for file names on UNIX systems, all input to GPP is case insensitive.

## BATCH PROCESSING

The following sections present the features and use of various tool menus, as one would use them in an interactive manner. GPP also has the ability to process commands from an input file containing a valid sequence of GPP menu commands. This approach can provide a very powerful capability to set up batch processing of a particular data set.

For example, we have implemented the complete GPP acceptance test protocol as a batch script and a set of GPP command files. NOTE: any line in a command file that begins with the "#" character in column one is treated as a comment line. As a sample, Figure 3-1 displays the contents of file `gpptst01.inp`, which is the first command file used in the acceptance test script. The first line is a comment, and the subsequent three command lines open file "a" for processing, generate summary statistics that are placed in file `a.st`, and then GPP is closed.

```
### Test 01: reads "a" and generates "a.st"  
c  
a  
q
```

**Figure 3-1. File contents for `gpptst01.inp`.**

## THE MENUS

### The Startup Screen

When you start GPP, it displays a startup screen (see Figure 3-2 below) with some program information and a menu. The first line tells you that you are running GPP. It is version 6.00 for the PC using the DVF compiler. The effective date of this version is March 1, 1999. GPP then displays the long version of the Startup Menu. After that, it says you have not activated a primary file, and that you can get help on these (and on all high-level tools) by entering a question mark. See the **Help Tool** subsection on page 3-19 for more information on obtaining help. The last line is the short form of the Startup Menu. To save time and screen space, GPP displays long menus only when first starting up or when you enter a naked return at one of the high-level menu prompts.

Probably, the first thing you will want to do is get data into GPP. If you want to analyze an ADAMS out file, you should call up the **Divide Tool** by entering a **D**. This will break out the different ADAMS requests from the file and create individual files that you can read into GPP. If you have done so, or if you want to look at other data, select the **Choose Tool** to choose a file and read it. For more information on the **Divide Tool** and **Choose Tool**, please see the **SYSTEM TOOLS** section on page 3-18.

```
Running the General-Purpose Postprocessor, GPP, v6.00pcd, 01-Jul-1999.

Startup Menu:
  C - Choose a primary file to activate.
  D - Divide an ADAMS .out file.
  E - Escape to a new command shell.
  Q - Quit GPP.
  O - Set options.
  ? - Help.

[No primary file has been activated] Enter "?" for help.

Enter Startup Menu choice (C,D,E,O,Q,?) >
```

**Figure 3-2. The Startup Screen.**

### The Tool Menu

After choosing a file, you will see the short form of the **Tool Menu** on the screen. In the example in below, GPP tells you that you have read in a primary data file, that its name is `adams.rq`, and that it is an ADAMS output file. If you enter a naked return, you will see the long version of the menu, shown in.

As you can see, the menu is quite a bit longer than the one in Figure 3-2. The tools seen on the **Startup Menu** are still available, but GPP added some new tools that process the chosen data in the primary array and produce some sort of output file. Please see the **ANALYSIS TOOLS** section on page 3-5 for more detail on these tools.

A useful tool here is the **Statistics Tool**. It is slightly different from the other tools that manipulate the primary array. It does not produce an output file (the **Choose Tool** produces an `st` file with full statistics when you read in the primary data), and it works on both the primary and secondary arrays. Page 3-12 contains a detailed description of the **Statistics Tool**.

### The Merge-Tool Menu

Once you have read a primary data file, you can choose the **M** option to go to the **Merge-Tool Menu**. The short form of the **Merge-Tool Menu** repeats the information about the primary file and tells how many primary files you have already merged into the secondary array. Again, if you do a naked return, you can see the long form of this menu, as seen in Figure 3-3.

The tools from the **Startup Menu** are still available, some new tools appear, and most of the tools that manipulate the primary data are gone at this point. Only the **Statistics Tool** remains. Once you merge some data into the secondary array, this tool will let you see the statistics of that array. Otherwise, you will get statistics for the primary array.

```
Tool Menu:
```

```
A - Azimuth average the primary data.
B - Bin the primary data.
C - Choose a primary file to activate.
D - Divide an ADAMS .out file.
E - Escape to a new command shell.
H - Histogram of the primary data.
I - Interpolate the primary data.
L - Least squares fit the primary data.
M - Merge files.
O - Set options.
P - Generate PSDs of the primary data.
Q - Quit GPP.
R - Rainflow count the primary data.
S - Show statistics of active data.
? - Help.
```

```
[The ADAMS file "adams.rq" is now the primary file]
```

```
Enter Tool Menu choice (A,B,C,D,E,H,I,L,M,O,P,Q,R,S,?) >
```

**Figure 3-3. The Tool Menu**

The Q option will let you leave the Merge-Tool Menu, but not GPP. If you have merged data into the secondary array, GPP will ask you if you want to move the merged data to the primary array. GPP will also ask if you want to write these results to a Merge Tool file. You can read Merge Tool files into the primary array just like any other data. Because GPP is familiar with the format of Merge Tool files, it knows how to extract the heading and column titles to make GPP easier to work with.

Figure 3-5 shows what the full Merge-Tool Menu looks like after you have merged some data into the secondary array. You now see tools for manipulating the secondary array. These tools enable you to “filter” the data in various ways, trim rows from the array, convert units, add a time or calculated channel, and write the merged results to a file. You can look in the FILTER TOOLS section on page 3-13 for information on the new tools.

```
Merge Tool Menu:
```

```
C - Choose a primary file to activate.
D - Divide an ADAMS .out file.
E - Escape to a new command shell.
M - Merge columns from primary array.
O - Set options.
Q - Quit merging files.
S - Show statistics of active data.
X - Create a calculated channel.
? - Help.
```

```
[The file "adams.rq" is now the primary file]
```

```
Enter Merge Tool Menu choice (C,D,E,M,O,Q,S,X,?) >
```

**Figure 3-4. The Merge-Tool Menu when the secondary array is empty.**

```
Merge Tool Menu:

C - Choose a primary file to activate.
D - Divide an ADAMS .out file.
E - Escape to a new command shell.
F - Filter the primary data.
G - Generate time column.
M - Merge columns from primary array.
O - Set options.
Q - Quit merging files.
S - Show statistics of active data.
T - Trim rows from the secondary data.
U - Convert units in the merge data.
W - Write out the merge file.
X - Create a calculated channel.
? - Help.

[The file "adams.rq" is now the primary file]
[ Secondary array contains 3 columns. ]

Enter Merge Tool Menu choice (C,D,E,F,G,M,O,Q,S,T,U,X,W,?) >
```

**Figure 3-5. The Merge-Tool Menu when there is data in the secondary array.**

## ANALYSIS TOOLS

GPP's analysis tools perform the mathematical and engineering analyses for which we wrote GPP. The other tools filter the data or are related to the OS.

### The Azimuth-Average Tool

You can bin your data against azimuth with the **Azimuth-Average Tool**. When GPP asks for the column number containing the azimuth data, the program will have already provided a default value. If this value is correct, you simply enter a naked return and continue setting up the remaining tool parameters. (GPP determines the default azimuth-column location as follows: first, GPP scans the column headings for the strings **AZ**, **Az**, or **az**. If it does not find this string, GPP uses the default column number set under the **Options Tool**. Normally, this will be the second column unless the user has altered this value. If your azimuthal data are in radians, use the **Units-Conversion Tool** to convert them to degrees.

This tool also produces the Fourier coefficients (converted to magnitude and phase) of the averaged data. You can compute these coefficients for as many as ten frequencies (the default number). This feature makes the **Azimuth-Average Tool** superior to the **Bin Tool** for azimuth averaging.

For long time series with varying wind conditions, it may not be meaningful to azimuth average the entire data set. Consider using the **Limit Tool**, (a subtool of the **Filter Tool**), to limit the data to cycles with similar conditions. Figure 3-6 is an example of a plot generated using data from the **Azimuth-Average Tool**.

The results file has an **aa** extension.

*See also:* **Bin Tool**, **Limit Tool**, **Units-Conversion Tool**, **Options Tool**.

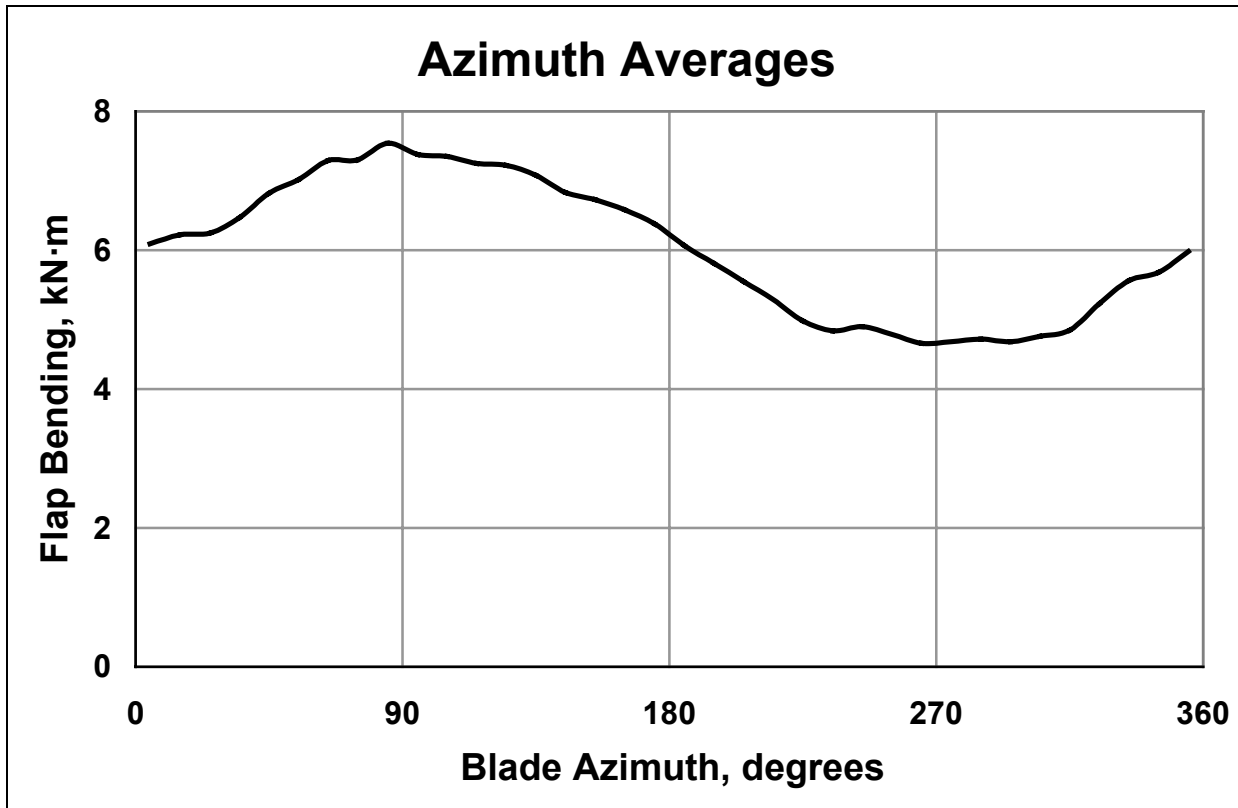


Figure 3-6. Sample plot using data generated by the Azimuth-Average Tool.

### The Bin Tool

The Bin Tool allows you to bin one or more columns of the primary data against one or two independent columns. It is useful for tasks such as plotting power versus wind speed or lift coefficient versus angle of attack. You could even use the Bin Tool to compute azimuth averages, but the Azimuth-Average Tool has additional features.

Like the Azimuth-Average Tool, it may be more meaningful to limit long time series to blocks or cycles that have similar conditions with the Limit Tool. Figure 3-7 is an example of a typical power curve using data generated by the Bin Tool.

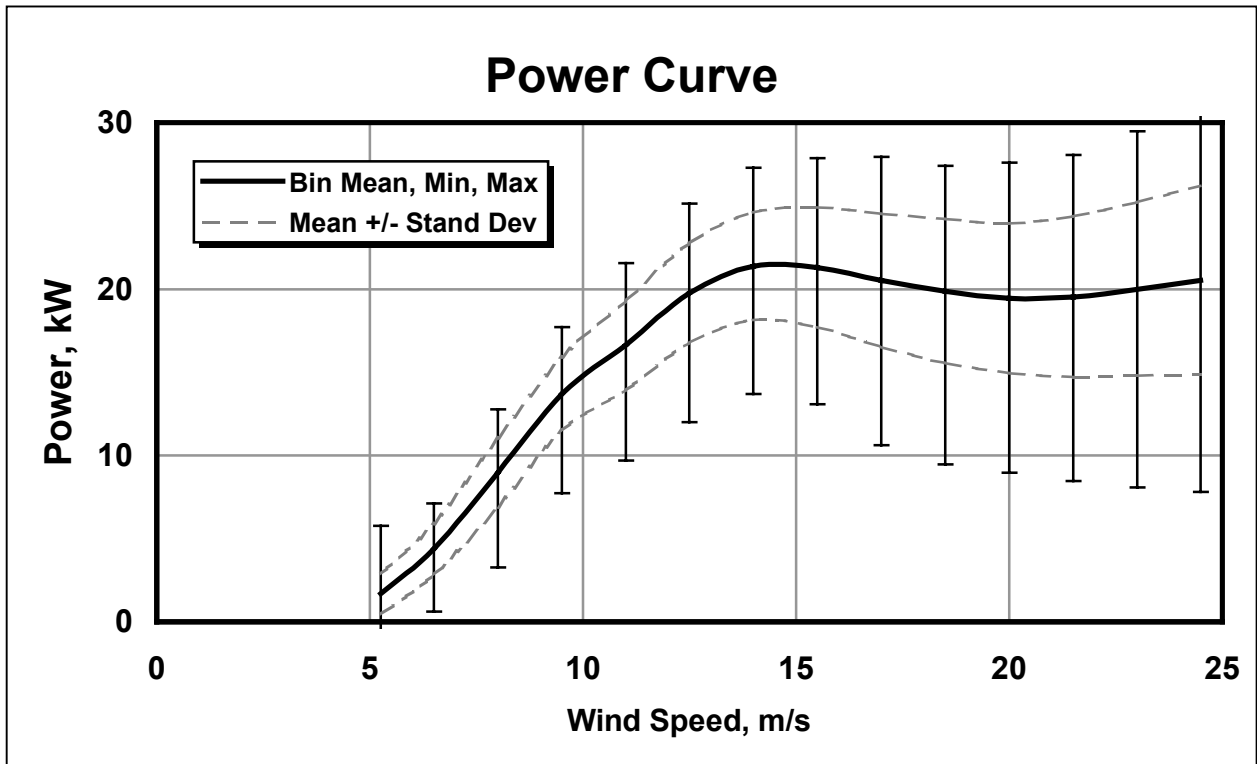
The results file has a `bin` extension.

*See also:* Azimuth-Average Tool, Limit Tool.

### The Histogram Tool

You can generate histograms (or probability density functions) of your data with the Histogram Tool. This tool writes a two-column table for each requested primary-data column to the results file. The table represents the `x`- and `y`-values for the analyzed column. For each requested column, the tool takes the domain between the minimum and maximum and divides it into bins of equal size. You tell the tool how many of these bins you want, and there will be a row in the output table for each of them. The `x`-value for each bin is the mean of all the `x` values in that bin. The `y`-value for each bin is the fraction of the total

data points for the column that fall within the bin. The area under the curve defined by the table is unity. Like some of the other analysis tools, it may be a good idea to limit your data to cycles with similar conditions.



**Figure 3-7. Sample power curve using data generated by the Bin Tool.**

Figure 3-8 is an example of a plot generated using data from the Histogram Tool. In addition to the bin-mean values, GPP version 6 also reports bin maxima, minima, and standard deviations.

The results file has a hi extension.

*See also:* Limit Tool.

### The Interpolation Tool

The Interpolation Tool allows you to interpolate dependent columns of the primary array using a new set of independent values (a new column 1). You can read the new independent values from a single-column file or specify a minimum, maximum, and step size. You may use either linear or cubic-spline interpolation. The cubic-spline routine is based on a routine CUBICSPL from *NUMERICAL METHODS* (Mathews, 1992). As in earlier versions of GPP, cubic-spline interpolation uses natural splines (zero second derivatives at the data endpoints.)

When the interpolation is finished, the tool tells you how many points it had to extrapolate. You probably want this number to be small or even zero.

Example:

Suppose you have two files with different time steps and want to merge them using the time history of the first file. You can use the Merge Tool to select only the time column and save the result to a

file with the Write Tool. Then, read in the second data file with the Choose Tool. Start the Interpolation Tool, and tell it to get the new independent series from a file, and give it the name of the file you generated with the Write Tool. The file created by the Interpolation Tool will be compatible with the first file, so you can merge them together.

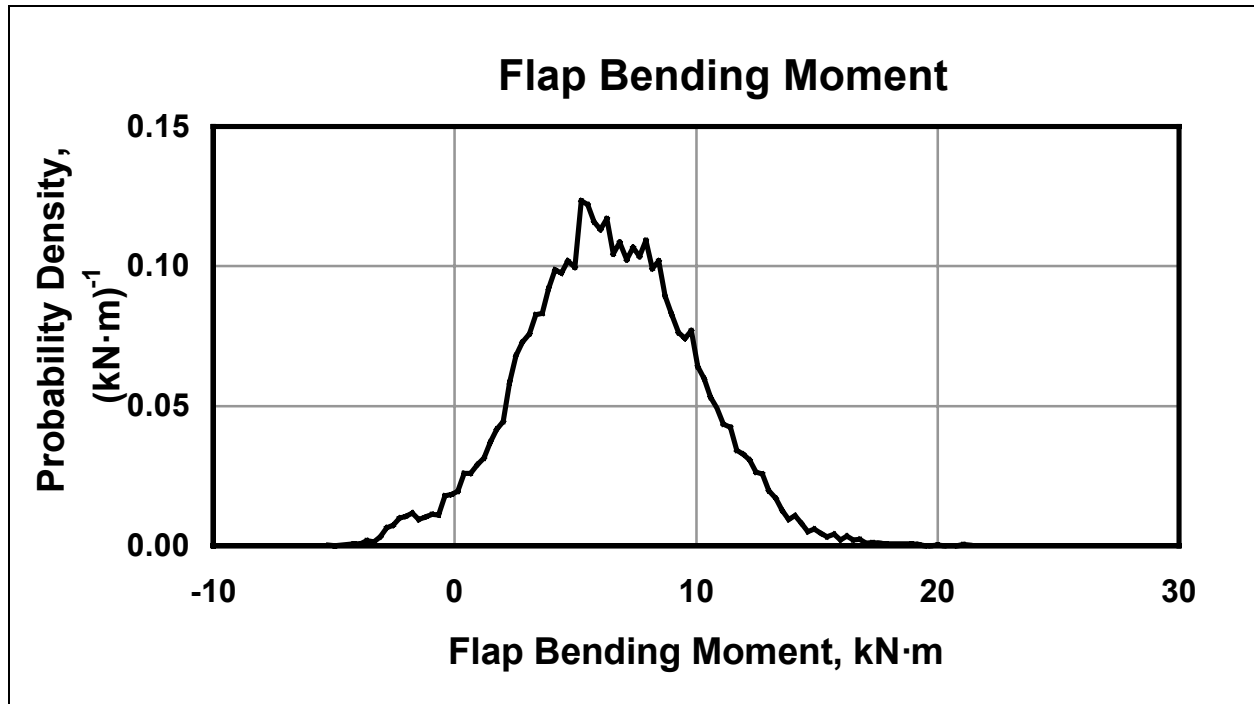


Figure 3-8. Example of a plot of Histogram Tool results.

The results file has an in extension.

See also: Merge Tool, Choose Tool, Trim Tool, Write Tool.

### The Least-Squares-Fit Tool

The Least-Squares-Fit Tool allows you to fit the primary data with polynomials of up to the tenth order. The tool uses a modified version of the LPOLYNOM routine from *NUMERICAL METHODS* (Mathews, 1992).

The tool generates a table of polynomial coefficients for each selected column. It also computes the  $\chi^2$  goodness of fit. It writes these tables to the results file.

The results file has an ls extension.

### The PSD Tool

You can generate PSDs of your data with the PSD Tool. It uses routines derived from GenPro, an old, batch-oriented, generalized post-processor developed at the National Center for Atmospheric Research (NCAR). Robert Lackman of NCAR developed much of GenPro, including the SPECFT routines it uses to generate PSDs. The SPECFT routines call another package, REALFT, which NCAR acquired elsewhere. The Fast Fourier Transform (FFT) algorithm in REALFT is desirable in that it does not limit

the number of input points to powers of two<sup>1</sup>. We heavily edited the SPECFT routines to eliminate unused features.

The PSD Tool removes the means from each time series before generating the PSDs. It also tapers the ends of the data with a cosine rolloff and zero-fills the rest of the array that holds the data. You have the option to detrend the data with a straight line, which *Numerical Recipes* (Press et al. 1990) recommends. You can change this option with the Options Tool.

The default PSD technique uses band smoothing. Other PSD types are cosine (Hamming), triangular (Bartlett), and rectangular windows. Rather than window averaging, you can also choose logarithmic spacing. Initial tests indicate there is little difference in the results for different PSD types for the data we usually analyze. An advantage of band smoothing is that you can specify the number of output frequencies. The Options Tool lets you set the type of PSD technique. Table 3-1 explains the benefits of each technique. Figure 3-9 shows a sample PSD plot using data generated by the PSD Tool with band smoothing.

The results file has a `ps` extension.

*See also:* Tool Menu, Options Tool.

**Table 3-1. Types of PSDs Available for GPP.**

Band Smoothing	Band smoothing allows you to specify any number of output frequencies (up to <code>#points/6</code> ). This allows you to reduce the volume of output and is a good choice if you have a large data set.
Cosine Windows	Using Hamming-style cosine windows is the best choice for random signals. Use this type of PSD if you are unsure. The technique produces half as many output frequencies as there were data points.
Logarithmic Spacing	Logarithmic spacing of output frequencies will give you less resolution at high frequencies.
Rectangular Windows	Rectangular windows are a good choice if your data have discontinuities or singularities. The technique produces half as many output frequencies as there were data points.
Triangular Windows	Triangular (Bartlett) windows give good results for deterministic signals, like sine waves. The technique produces half as many output frequencies as there were data points.

---

<sup>1</sup> In porting the REALFT routines to GPP, we discovered a bug in the kernel for the radix prime transform. The only workaround we found for this bug is to make sure that REALFT does not call this transform. REALFT will call it only if the number of data points factors down to at least one prime number greater than five. We modified SPECFT so that it will pad the time series with zeros to fill the array that holds it.



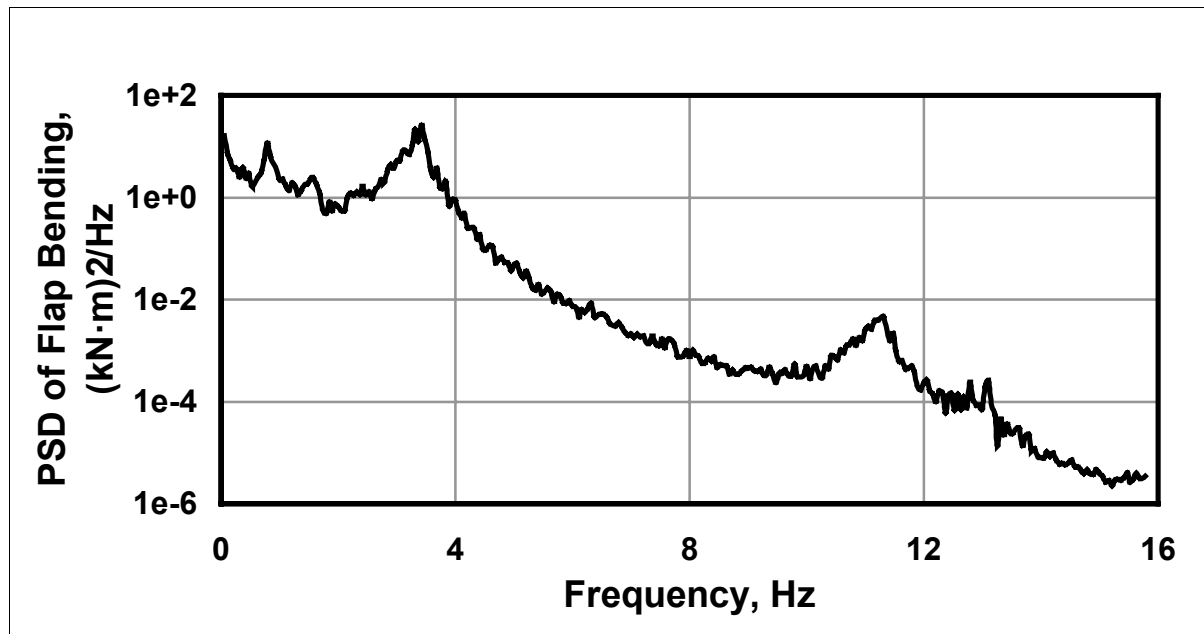


Figure 3-9. Example PSD plot using data generated by the PSD Tool.

### The Rainflow-Cycle-Count Tool

You can count rainflow cycles with the Rainflow-Cycle-Count Tool. GPP uses a vector-based rainflow-counting algorithm derived from Downing and Socie (1982) as implemented in the LIFE2 program (Schluter and Sutherland 1991). The tool cycle counts a time history as it occurs and identifies the same cycles as the two-pass algorithm that requires that the history be rearranged. After the tool identifies the cycles, it bins them to generate curves of cycles per the number of seconds specified for the count period versus the peak-to-peak cycle amplitudes (and, optionally, cycle means for two-dimensional (2D) cycle counting). You can specify the count period with the Options Tool. You can also optionally apply cycle-peak extrapolation and smoothing before rainflow counting by setting these flags in the Options Tool.

Figure 3-10 shows a sample plot of data generated by the Rainflow-Cycle-Count Tool. If you look at the two right-most data points, you will see that they have the same y-value. These cycles either occurred only once or were incomplete cycles. This algorithm closes all the incomplete cycles. We believe these single cycles and artificially closed cycles can be very misleading, so we often delete them from our plots. If your time series were longer, these points would be lower on the plot. If you are using the data to estimate fatigue life, you may want to keep them, as they give you a conservative estimate. These cycles will have a rate that is the inverse of the elapsed time. The Rainflow-Cycle-Count Tool does not output any values for bins that have no points. This eliminates zeros from the series and makes log-linear plots possible.

GPP will also perform mean and range cycle counting (2D). Figure 3-11 shows a sample 3D bar plot of 2D mean-range data generated by the Rainflow-Cycle-Count Tool.

GPP does not give you the option of processing the time data column (column one). For Merge Tool files, you must ensure that time is in the first column to use this tool. The results file has an *rf* extension.

*See also:* Tool Menu, Options Tool.

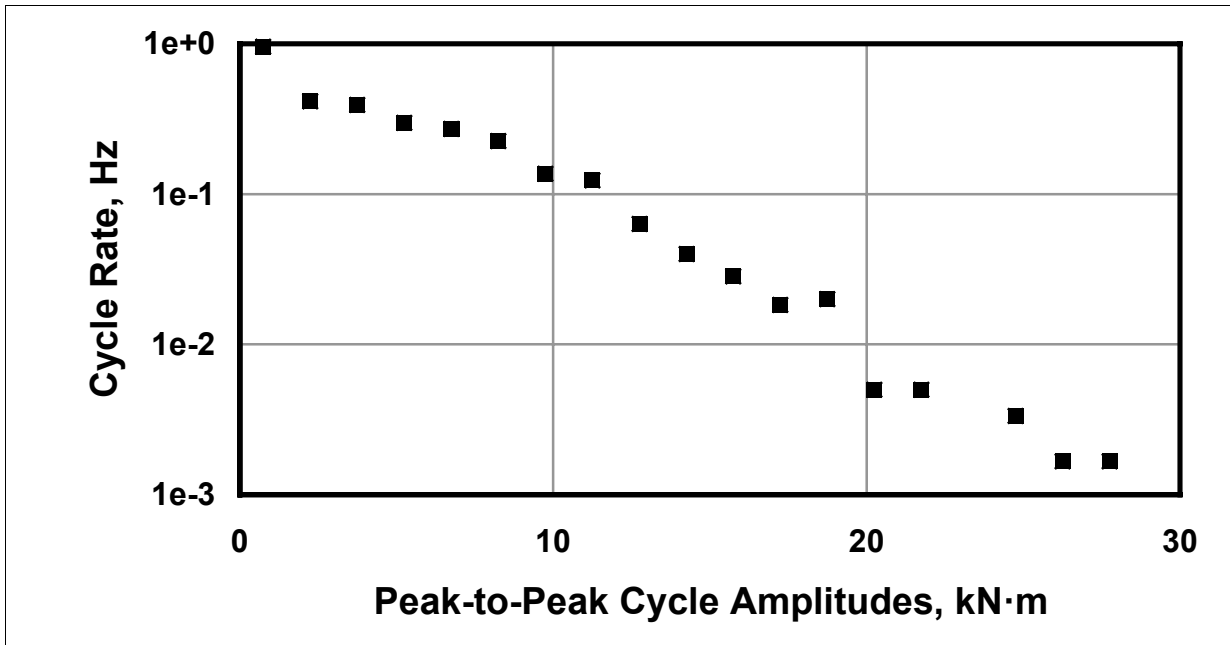


Figure 3-10. Example plot using data generated by the Rainflow-Cycle-Count Tool.

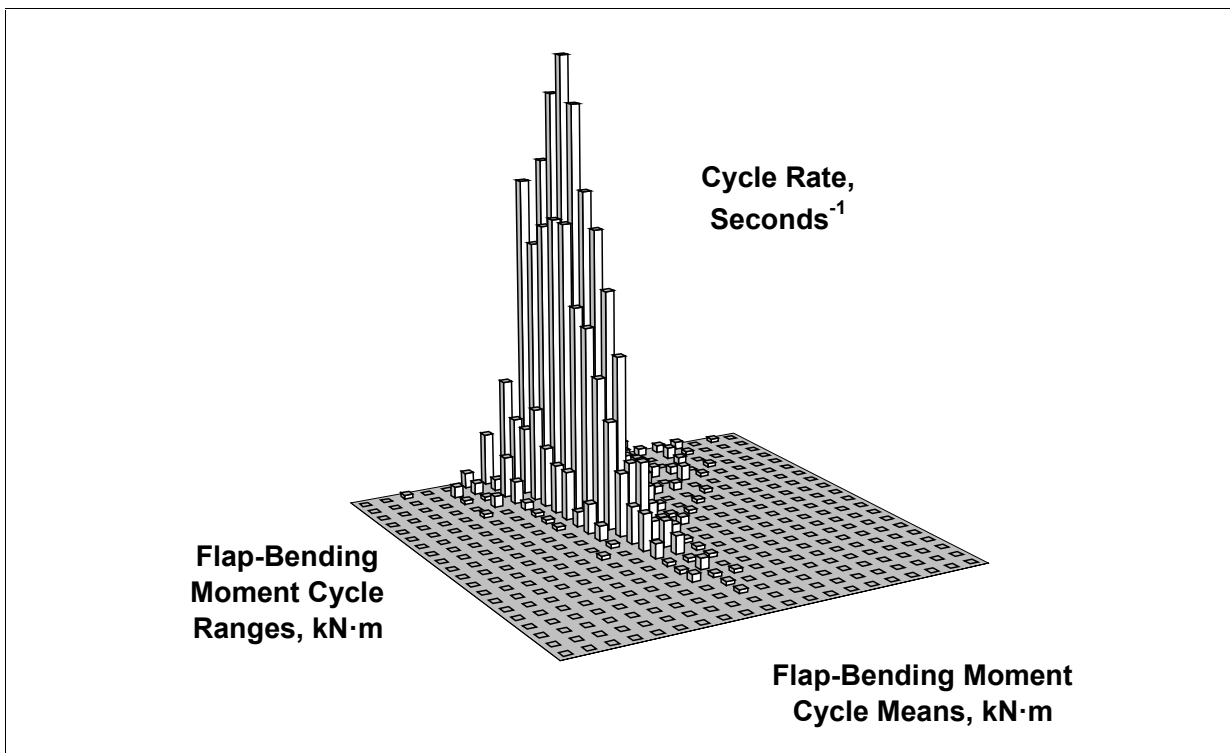


Figure 3-11. Example of a 3D bar chart using 2D rainflow-cycle-count data.

## The Statistics Tool

The **Statistics Tool** displays statistics for the data arrays. These statistics include minimum, mean, maximum, standard deviation, and skewness values. They also include the data record numbers on which the minima and maxima<sup>2</sup> occurred. Figure 3-12 shows a sample statistics screen.

In this sample, because the data came from a recognized file type, the **Choose Tool** knew how to determine the column headings. Thus, the tool includes column descriptions in the output. If this had been a generic file, the first column will contain headings or column numbers depending in whether GPP successfully extracted column headings from the file. This is another example of how GPP changes its behavior for familiar file types.

If there are data loaded into the secondary array, the **Statistics Tool** will ask you whether you want to look at the statistics for the primary or secondary data. The default is the secondary data, as this is most likely what you would be working with at the time. If there are no secondary data, this tool will automatically show the primary-array statistics. The **Choose Tool** calculates the statistics for the primary array after it reads in the data. After each merge action, the **Merge Tool** calculates the statistics for the secondary array. Accessing the **Statistics Tool** only displays statistics—it does not compute them.

When the **Choose Tool** reads a primary file, it also includes a kurtosis calculation and writes all of the statistics to a file with the extension **st**. Figure 3-13 shows an example of this file. Here, you can see that GPP tells you where it pulled the data and when it generated the statistics. It includes the original heading, and sets it off with dashed lines. It then lists the statistics for each column of data. As you can see, GPP writes the statistics file with one more significant digit than you saw in the screen display of the same data.

*See also:* Tool Menu, Choose Tool, Merge Tool.

Statistics for data from "adams.rq":						
Parameter	Minimum (Index)	Average	Maximum (Index)	Std. Dev.	Skewness	
Time	5.0 ( 1)	35.	65. ( 1921)	17.	-5.63E-07	
AZIMUTH	-1.80E+02( 1311)	0.50	1.80E+02( 782)	1.04E+02	-5.89E-03	
BLD_1_RF	2.6 ( 1)	6.7	11. ( 146)	2.7	0.13	
BLD_1_RE	-4.3 ( 19)	1.9	8.1 ( 243)	4.2	2.12E-02	
LSS_TOR	5.5 ( 19)	6.3	6.9 ( 25)	0.36	-0.18	
POWER	-34. ( 25)	-31.	-27. ( 19)	1.8	0.18	
YAW_MOM	-6.54E+03( 17)	-5.89E+03	-5.46E+03( 5)	1.39E+02	0.22	
FA_THRST	-1.58E+02( 796)	-7.10E-02	1.6 ( 8)	3.6	-43.	
SS_THRST	-3.9 ( 796)	-8.06E-02	2.4 ( 609)	0.43	-1.6	

**Figure 3-12. An example display from the Statistics Tool.**

<sup>2</sup> It may be useful to note that the index for the time maximum is the number of records in the file.

```

Statistics for data from "adams.rq":
[Generated by GPP (v6.00, 1-Jul-1999) on 1-Jul-1999 at 09:12:30]

Original heading:
-----
ADAMS/View model name: micon65_v3.21
Request Number      1

Units are in seconds. degrees. kN-m. and kW.
-----

```

Parameter	Minimum	(Index)	Average	Maximum	(Index)	Std. Dev.	Skewness	Kurtosis
Time	5.00	( 1)	35.0	65.0	( 1921)	17.3	-5.629E-07	-1.20
AZIMUTH	-180.	( 1311)	0.495	180.	( 782)	104.	-5.887E-03	-1.21
BLD_1_RF	2.63	( 1)	6.72	10.8	( 146)	2.71	0.129	-1.48
BLD_1_RE	-4.33	( 19)	1.92	8.10	( 243)	4.22	2.118E-02	-1.45
LSS_TOR	5.54	( 19)	6.34	6.92	( 25)	0.357	-0.177	-1.41
POWER	-34.2	( 25)	-31.3	-27.3	( 19)	1.77	0.177	-1.41
YAW_MOM	-6.544E+03	( 17)	-5.893E+03	-5.456E+03	( 5)	139.	0.223	-0.385
FA_THRST	-158.	( 796)	-7.100E-02	1.60	( 8)	3.60	-43.4	1.894E+03
SS_THRST	-3.93	( 796)	-8.060E-02	2.36	( 609)	0.432	-1.60	17.2

**Figure 3-13. An example statistics file.**

## FILTER TOOLS

We use the term “filtering” to mean more than noise filtering. We use the term to mean something that does not let everything get past it. A low-pass filter reduces noise in your data and a decimator reduces the volume of your data by eliminating all but every  $n^{\text{th}}$  record.

### The Filter Tool

The Filter Tool is available from the Merge-Tool Menu after you have used the Merge Tool to load the secondary array. This tool allows you to select subtools that let you filter noise, average, decimate, or limit data in the secondary array. One of these subtools is the Limit Tool, which lets you set up criteria for blocks or cycles of data that the tool will keep or delete. A similar tool is the Trim Tool, which allows you to throw away blocks of data you do not want to process.

The noise filter is a general, band-pass, discrete Infinite-Impulse-Response (IIR) filter. It allows you to apply a low-pass, high-pass, or band-pass filter to your data. The formulation of the band-pass filter follows the derivation of the IIR filter described in *Numerical Recipes* (Press et. al. 1996). This recursive filter exhibits a somewhat rounded and asymmetric filter response below and above the user-selected cut-off frequencies. The data is filtered twice (forward and backward) to remove the phase shift. The formulation of the filter equations assumes evenly spaced data with time in the first column.

*See also:* Limit Tool, Merge Tool, Trim Tool.

### The Limit Tool

The Limit Tool is very powerful subtool available from the Filter-Tool menu. It allows you to specify which blocks of data you want to keep for later processing. Blocks can be of any length (even a single record), and you can specify them by number of records or amount of time.

You can also specify the block length as being a single rotor cycle. The **Limit Tool** does not assume a constant RPM machine. It defines a cycle boundary as being when the azimuth goes from greater than 300 degrees to less than 60 degrees. If your data goes from -180 to 180 degrees, the tool will map the negative values as 180 to 360 degrees. The **Limit Tool** uses this mapping only to determine cycle boundaries. It will not change the azimuth values in your data. If your azimuth data are in radians, use the **Units-Conversion Tool** to convert them to degrees.

The **Limit Tool** asks you for a list of criteria that can be in a stored file or entered through the keyboard. Each criterion consists of a column, a minimum, a maximum, and a type. The type can be the minimum value, the mean, the maximum value, the standard deviation, or every value of a candidate block. Entering the type as a negative value tells the **Limit Tool** to pass only those blocks for which the statistic lies outside the specified range. Table 3-2 shows the values you enter for different types of limiting factors.

Examples:

Suppose you want to exclude all rotor cycles where the wind speed (channel 5) exceeds 15 m/s even once during the cycle. Using 500 as an arbitrarily large number, you would limit by cycles and specify the limiting criterion as:

5, 15, 500, -3

Suppose you want to find all 1-minute time blocks where the wind direction (channel 6) changed little. In this case, you should limit by 60-second blocks and may want to specify that all values in the block lie between -10 and +10 degrees:

6, -10, 10, 5

Suppose the azimuth (channel 2) goes from -180 to 180 degrees. If you want to include all records that lie outside the tower shadow, you would limit by individual records and specify the limiting criterion as:

2, -170, 170, 5

*See also:* Filter Tool, Interpolation Tool, Trim Tool, Units-Conversion Tool.

**Table 3-2. Types of Limiting Factors.**

1	The minimum value in each block
2	The mean of each block
3	The maximum value in each block
4	The standard deviation of each block
5	All values in each block

## The Merge Tool

This tool allows you to select columns from the primary array and put them into the secondary array. After transferring columns from the primary array, you can read in another primary file and merge columns from it into the existing secondary array. You can thus merge multiple files into one file. If you

already have data in the secondary array, the Merge Tool will ask you if you want to erase the old data and start anew.

When you are through using the Merge Tool, you can move the secondary array to the primary array and/or write it to a GPP Merge Tool file. After you move the secondary array to the primary array, you can use the analysis tools on the merged data. You can also read merged files using the Choose Tool and analyze them like other files. GPP recognizes Merge Tool files and takes advantage of their structure as it does for other recognized file types.

When you merge data into the secondary array, the Merge Tool creates a header that gives you a history of merging operations. This header tells you which files you merged and when you merged them. The column labels indicate the file and column of origin. The format for these labels is F<file#>-<col#> (for example, F2-3 for the third column of the second file). If GPP has been able to retain column names (as from ADAMS, YawDyn, or other recognized file types), the Merge Tool will replace the column number with the column name (for example, F2-azimuth for the azimuth column of file 2). When you manipulate the data, the tools you use update the header to record your changes. Figure 3-14 shows an example header.

If the number of data rows in the current primary and secondary arrays differ, GPP will warn you. If the primary array is longer, GPP will truncate the new data to match the size of the secondary data array. If the secondary array is longer, you will be given a choice to cancel the merge operation or trim the length of the secondary array.

The section on the Interpolation Tool on page 3-7 explains how to merge files with different time steps and/or start and end times.

Some high-level tools manipulate data in the secondary array. See appropriate sections for help on the Filter Tool, Limit Tool, Trim Tool, and Units-Conversion Tool. Quite involved channel combinations and transformations are possible using the Calculated-Channel Tool.

*See also:* Filter Tool, Interpolation Tool, Limit Tool, Trim Tool, Units-Conversion Tool, Calculated-Channel Tool.

```
GPP Merge Tool file from "adams.out"; "yawdyn.plt".  
Generated by GPP (v6.00, 01-Jul-1999) on 01-Jul-1999 at 14:47:05; 01-Jul-1999 at 14:48:21.  
1-Time 1-FlapDef1      2-FlapAng1
```

**Figure 3-14. Example of a Merge Tool file header.**

### The Trim Tool

The Trim Tool eliminates rows from the secondary array. You can specify a block of rows by row number. If the first column has monotonically increasing values (like time), you can also eliminate rows by value. This tool is available only from the Merge-Tool Menu after you have merged some data into the secondary array.

*See also:* Filter Tool, Interpolation Tool, Limit Tool, Merge Tool.

## MISCELLANEOUS TOOLS

### The Calculated-Channel Tool

The Calculated-Channel Tool, which is available from the Merge-Tool Menu, uses an algebraic expression parser. This parser allows you to generate a new data channel, which is the result of a calculation based on other channels in the primary- or secondary-data arrays. Primary-data columns are referenced by **p#** and secondary-data columns are referenced by **s#**, where "#" must be a valid, existing column number. GPP supports normal, algebraic expressions, nested parentheses, and most Fortran intrinsic functions. An example of a valid expression follows:

$$\text{New Channel} = ( \sin(p1) + 1.5e+02*( \cos(p3)/2.0 ) )$$

where p1 and p3 refer to data in columns 1 and 3 of the primary data array.

The expression parser recognizes the functions listed in Table 3-3. Trigonometry functions ending in "D" are versions of the functions that use degrees instead of radians. Further information on each of these functions is provided in the program help text accessed by the "?" option in the Calculated-Channel Tool.

**Table 3-3. Functions Available for the Calculated-Channels Tool.**

ACOS	ATAN	COS	EXP	LOG10	NINT	SIGN	TAN
ACOSD	ATAND	COSD	FLOOR	MAX	NUMS	SIN	TAND
ABS	ATAN2	COSH	GAMMA	MIN	OPS	SIND	TANH
ASIN	ATAN2D	DIM	INT	MOD	PI	SINH	
ASIND	CEILING	E	LOG	MODULO	ROOT	SQRT	

The parser recognizes the standard operators (+, -, \*, and /) used in algebraic operations. Either "^" or "\*\*" are recognized for exponentiation. The tool allows one to nest parentheses up to the limit of the size of the operation stack (currently set at 256 for the total number of pending operations). The order of operator precedence from highest to lowest is

(, )  
 \*\*, ^  
 \*, /  
 +, -

The tool evaluates expressions within the most deeply nested parentheses first. GPP ignores any spaces.

Constants may be entered in the expression as normal decimal numbers or with scientific notation (for example, 0.0123, 1.23E-02 or 1.23D-02).

*See also:* Merge Tool.

### The GenTime Tool

The GenTime Tool is available from the Merge-Tool Menu. This tool allows you to add a time channel to an existing secondary data file. The tool asks you for the starting time value and either the maximum time value or the desired time step. GPP makes the new time channel the first column, and shifts

all other channels to the right. The **GenTime Tool** is available only if you have already created a secondary data set. The reason for this requirement is that GPP must know how many times steps it should create.

*See also:* Merge Tool.

### The Trim Tool

The **Trim Tool** eliminates rows from the secondary array. You can specify a block of rows by row number. If the first column has monotonically increasing values (like time), you can also eliminate rows by value. This tool is available only from the **Merge-Tool Menu** after you have merged some data into the secondary array.

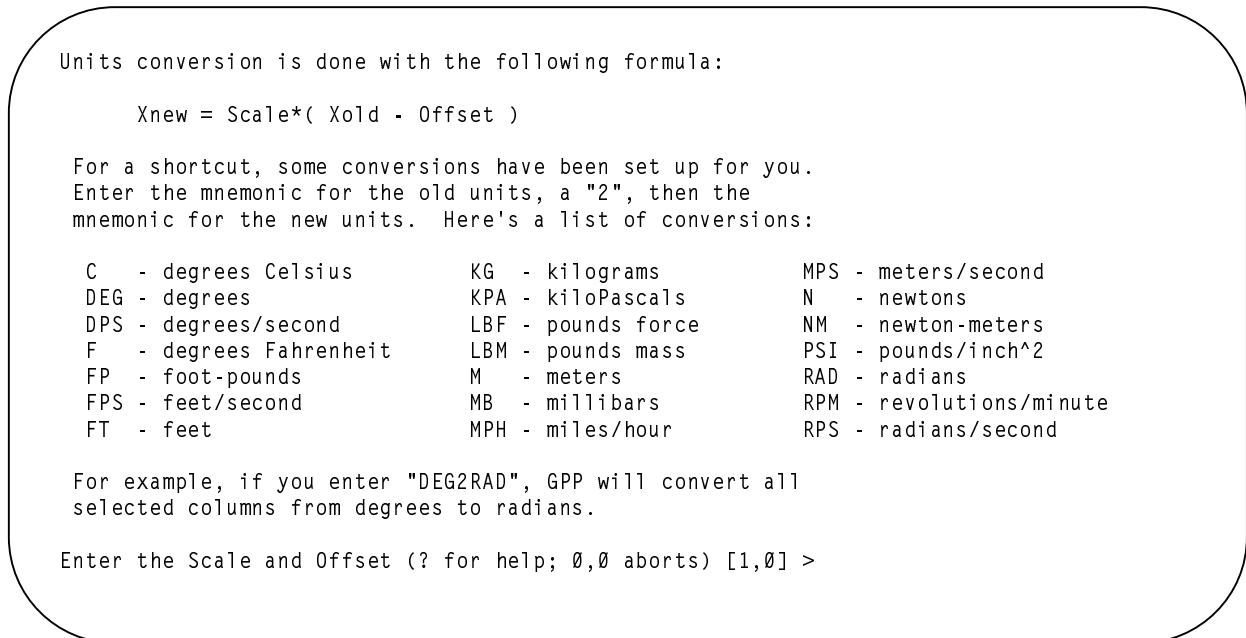
*See also:* Filter Tool, Interpolation Tool, Limit Tool, Merge Tool.

### The Units-Conversion Tool

The **Units-Conversion Tool** is available from the **Merge-Tool Menu**. This tool allows you to rescale the data in selected columns in the secondary array with the same scales and offsets. The **Units-Conversion Tool** does the scaling using the formula:

$$X_{\text{new}} = \text{Scale} * ( X_{\text{old}} - \text{Offset} ) .$$

We have built many automatic conversions into the **Units-Conversion Tool**. When asked for the scale and offset, enter a ? to see the list of those available. For example, instead of entering 0.55555,32 to convert from Fahrenheit to Celsius, you can enter F2C. This way, you do not need to look up conversion factors. Figure 3-15 shows the help screen for these conversions.



**Figure 3-15. The help screen for the Units-Conversion Tool.**



## SYSTEM TOOLS

The system tools allow you to interact with the system resources of your computer and to tell GPP how to behave. They perform essentially no analyses.

### The Choose Tool

The Choose Tool allows you to choose a file to read into the primary array. The tool prompts you for the name of the file and will read it into memory. GPP cannot read files with more than 1000 columns or 500,000 rows of data<sup>3</sup>. The current version of the Choose Tool can read only ASCII files. We may add the ability to read binary files in the future. After reading the data, the Choose Tool will compute statistics, create a file of them, and give the file an `st` extension. You can use the Statistics Tool to see the statistics within GPP.

The Choose Tool can automatically detect files created by ADAMS, YawDyn, FAST\_AD, ADAS, UAE, *BLADED* or the Merge Tool and tailor its behavior accordingly. GPP does this by detecting a specific string at the beginning of the first line. It also recognized files with tab-delimited column headings. The Options Tool section describes how to change these strings for the file types mentioned.

When ADAMS generates an out file, it always creates nine columns—even if you do not request any data for all of them. If you label a column as “UNUSED”, the Choose Tool will eliminate that column from the primary array. If columns with real data follow an unused column, the tool will shift them to the left.

For generic files, the Choose Tool will ask you for the number of header lines and number of columns of data. If you enter a ? in response to either of these two questions, GPP will display the file with the More Tool to allow you to scan the top of the file.

GPP’s tools that use the rotor azimuth position will offer a default for the azimuth column. GPP first searches the column headings for the string “AZ” (any case). When it finds one that contains the string, it makes that column the default azimuth column. If column headings are not used or if GPP does not find the string, it uses the default value established in the Options Tool. You can always override the default by entering a specific column when asked for the azimuth column.

GPP will write results from most of its tools to files that have the same root name as the chosen primary file. Those tools will add unique extensions to the root name to identify the tool.

*See also:* Merge Tool, More Tool, Options Tool, Statistics Tool.

### The Directory Tool

When GPP prompts you to enter an input file name, you can enter a ? and the Directory Tool will show you a list of files in the current directory.

### The Divide Tool

You can divide an ADAMS out file with the Divide Tool. It will create a new file for each ADAMS request in the file. The new file names will have sequential numbers added to their root names and `rq` ex-

---

<sup>3</sup> You can change these limits by modifying the `MAX_COLS`, `MAX_ROWS`, and `MAX_HEAD` parameter statements in the `WORKING_DATA` module found in the `GPP_MOD.FOR` file. You must then recompile GPP to put the modifications into effect.

tensions (for example, `adams.out` may generate `adams1.rq` and `adams2.rq`). The sequential numbers denote the ADAMS request numbers. You can read these new files using the **Choose Tool** and manipulate them with other GPP tools.

If the resulting file name is too long for the operating system, the **Divide Tool** will prompt you for a new name. For Windows and UNIX files, GPP limits root names to twenty characters. The **Divide Tool** limits ADAMS out files to nine different requests.

*See also:* **Choose Tool**.

### **The Escape Tool**

You can temporarily escape to the OS with the **Escape Tool**. This allows you to look around, and perform other commands without having to leave GPP. Enter the `exit` command when you want to return to GPP.

If you are a UNIX C shell user, your aliases will not be available to you unless you set them in your `.cshrc` file. If you have many aliases, setting them all in this file can degrade performance for your other processes. You may want to consider putting your aliases into a separate file (for example, `.aliasrc`) and placing an alias in your `.cshrc` file to allow you to set them interactively (example: `alias .a 'source ~/.aliasrc'`). You can then just enter `.a` after you start the new command shell to set all your aliases.

### **The Get-Columns Tool**

GPP calls this low-level tool from many other tools. The tool asks you for a list of columns to process. If you enter a naked return, the tool will select most columns automatically. Some high-level tools tell this tool to not include certain columns.

You can also enter a list of column numbers and separate them with spaces, commas, or tabs. You may also designate a contiguous range of columns by the first and last column separated by a dash (with no additional spaces or tabs in between [for example, 3-7]). The **Get-Columns Tool** will display a table of selected columns and notify you of their validity. You can enter a naked return to accept the list or enter `N` to reject it. The tool will display a list of numbered column headings to help you choose columns for familiar files. It displays the file header for files without column names.

### **The Help Tool**

You can get on-line help for using GPP by entering a `?` at a high-level menu. A new menu of help options will appear on the screen from which you can repeatedly request help on different subjects. If you enter an option letter after the question mark at a main menu, you will get help on that option only and you return to that menu.

Examples:

- ? Sends you to the **Help Menu**.
- ?A Gives help on the **Azimuth-Average Tool**.
- ?? Gives help on the **Help Tool**.

### **The More Tool**

When you read in a generic file with the **Choose Tool**, you can enter a `?` when asked for the number of rows and/or columns in the file. If you do, the **More Tool** will display the first seven lines of the file and

ask if you want to see more. Just press ↵ to see more of the file or enter a Q to quit the **More Tool**. We chose seven lines so that files with long lines could wrap the screen without scrolling important information off the top. The **More Tool** precedes each line with the text **Line #>**, where # is the number of the line. This helps you determine the length of your header.

## The Options Tool

The **Options Tool** allows you to review and/or change program options. If you do not like the defaults, you can change them for just the current run or for all future GPP sessions. GPP saves permanent changes to the options in a user file. For UNIX users, GPP stores options in `.gpprc` in your home directory. For Windows users, GPP stores options in `GPP.INI` in the current directory. Windows users can change the location where `GPP.INI` is stored by setting an environment variable called `HOME` to point to another directory. You may want to put this **SET** command in your `AUTOEXEC.BAT` file if you use Windows 95 or 98. If you use Windows NT, set it on the Environment Tab of the System Control Panel. Here's an example of a **SET** command:

```
SET HOME=C:\USER
```

Options currently available for GPP include strings that identify `ADAMS`, `YawDyn`, `FAST_AD`, `ADAS`, `UAE`, and `BLADED` files. See help on choosing files for more information on these strings. Another option is the "dot display rate," which determines how many operations elapse before GPP displays a dot on the screen to let you know that it is processing data. You will want to make this number smaller for slower systems. Other options let you tell GPP to prompt you when you quit, tell GPP how to generate PSDs, set directory-path default options, adjust merge file column headings, or to set options for the **Rainflow-Cycle-Count Tool**. If you are reading data from a read-only device like a CD-ROM, use the output-path option to tell GPP where to put its output.

*See also:* Choose Tool, PSD Tool, Rainflow-Cycle-Count Tool, Merge Tool.

## The Write Tool

The **Write Tool** is available from the **Merge-Tool Menu**. With it, you can write out the results of a merge to a file. The tool prompts you for a file name. You can name your file anything within the limitations of your system (as long as it is no longer than 100 characters; paths are allowed). If you specify an existing file, GPP will ask if you want to overwrite it. If you enter a naked return, GPP will cancel the **Write Tool** and return you to the **Merge-Tool Menu**.

*See also:* Merge Tool.

## QUITTING GPP

Enter a Q at the menu prompt to quit GPP. If you enter a Q when in the **Merge-Tool Menu**, you will quit the **Merge Tool**, not GPP. If you want GPP to prompt you before quitting, you can set that option with the **Options Tool**.

*See also:* Merge Tool, Options Tool.

Starting the Options Tool.

Current option settings are:

```
1 - ADAMS identifier           = "1ADAMS"
2 - ADAS identifier            = "Data header file"
3 - BLADED identifier          = "FILE"
4 - FAST identifier            = "FAST_AD"
5 - UAE identifier             = "UAE"
6 - YawDyn identifier          = "YawDyn"
---
7 - Rainflow Bin Norm Flag     = No
8 - Rainflow Peak Filter Flag  = No
9 - Rainflow Filter Threshold (%) = 0
10 - Rainflow count period (sec) = 1
---
11 - Type of PSD               = "Band Smoothing"
12 - Linear detrending for PSDs = No
---
13 - Azimuth column            = 2
---
14 - Merge file number in titles = Yes
15 - Output Path                = "same"
16 - Path Separator            = "\"
17 - Dot display rate           = 500
18 - Prompt before quitting     = No
---
Q - Quit changing options
```

Enter option you wish to change (1,...,18,=Q) >

**Figure 3-16. An example of the Options Menu.**

# References

Downing, S.D.; Socie, D.F. (1982). “Simple Rainflow Counting Algorithms.” *International Journal of Fatigue*; Vol. 4, [N.1], pp. 31–40.

Schluter, L.L.; Sutherland, H.J. (1991). *User’s Guide for LIFE2’s Rainflow Counting Algorithm*. SAND90-2259. Albuquerque, NM: Sandia National Laboratories. Laboratory, January 1991.

Mathews, J.H. (1992). *Numerical Methods for Mathematics, Science and Engineering. Second Edition*. New Jersey: Prentice Hall; pp. 278, 297.

Press, W.J.; Flannery, B.P.; Teukolsky, S.A.; Vetterling, W.T. (1996). *Numerical Recipes, The Art of Scientific Computing (FORTRAN Version). Second Edition*. Cambridge: Cambridge University Press; pp. 541–551, 553–555.

# Glossary

Familiar File	We programmed GPP so that it would recognize often-used files. GPP modifies its behavior to take advantage of its knowledge of a file's structure. The current list of familiar files includes those produced by ADAMS, YawDyn, FAST_AD, ADAS, UAE, <i>BLADED</i> and the Merge Tool. It also recognized files with tab-delimited column headings. This is the opposite of a generic file.
File Header	GPP considers the file header to be the text at the beginning of a file. For generic files, GPP will inspect the first 50 lines of the file for alphabetic text indicating likely header lines. If the estimated number of header lines is correct you may accept this number or tell the <b>Choose Tool</b> how many header lines there are in the file. <i>Do not use the header to eliminate unwanted data at the beginning of the file—use the Trim Tool for that.</i> When GPP asks you for information like a list of desired columns, the <b>Get-Columns Tool</b> will display the entire header to help you figure out column numbers.
Generic File	Generic files are those that are unfamiliar to GPP. It does not understand their structure, so you must tell it how they are organized. They are the opposite of familiar files.
Naked Return	Enter a naked return by pressing ↵ (the Enter key) without pressing any other keys first. Use it to choose a default answer or to ask for the long version of a high-level menu.
Parse	To parse is to scan text for meaningful words. The words can be English-like commands or numbers.
Primary Array	The primary array stores data that you process with most of the high-level tools. GPP generally considers the primary array as a time series (time in the first column), but it does not need to be for some tools. You use the <b>Choose Tool</b> to fill the primary array with data. You can also copy the contents of the secondary array into the primary array when you quit the <b>Merge-Tool Menu</b> . The default dimensions of the primary array are 500,000 rows by 1000 columns. You can change these dimensions by modifying the <code>MAX_ROWS</code> and <code>MAX_COLS</code> parameters in the <code>WORKING_DATA</code> module in the <code>gpp_mod.for</code> file and recompiling the program.
Root name	The root name of a file is all the text before the last period in the name. For Windows and UNIX, the limit for root names is 20 characters.
Secondary Array	Create the secondary array by copying columns from the primary array with the <b>Merge Tool</b> . The filtering-type tools manipulate the data in the secondary array and do not change its time-series nature. The <b>Limit Tool</b> and <b>Trim Tool</b> can eliminate blocks of data from the array and make time discontinuous. Be sure that you do not use the <b>PSD Tool</b> on such data after you have loaded it into the primary array.

# Index

## A

ADAMS, 3-1, 3-2, 3-3, 3-15, 3-18, 3-19, 3-20, xi  
Aliases, 3-19  
Analysis Tools, **3-5–3-20**  
ASCII file, 3-18  
AUTOEXEC.BAT file, 3-20  
Azimuth-Average Tool, iv, **3-5–3-6**

## B

Band smoothing, 3-9  
Band-pass filter, 3-13  
Bartlett window, 3-9  
Bin Tool, 3-5, **3-6–3-7**  
Binary file, 3-18  
*BLADED*, iii, 1-2, 3-1, 3-18

## C

C shell, 2-3, 3-19  
Calculated Channel Tool, 3-1, 3-15, 3-16  
Case insensitivity, 3-2  
Choose Tool, 3-3, 3-7, 3-12, 3-15, **3-7–3-18**,  
3-19, 3-20, xi  
Compiling GPP, **2-1–2-3**  
Cosine window, 3-9

## D

Default response, vii, 1-2, **3-2**  
Digital Visual FORTRAN, 1-1, 2-1, 2-3, 3-2  
Directory Tool, **3-18**  
Divide Tool, 3-2, 3-3, **3-18–3-19**  
DOS, vii, **2-1**, 2-2, **2-4**  
Dot display rate, 3-20

## E

Enter  
Naked. *See* Naked return  
Environment variable, 2-2, 3-20  
Escape Tool, **3-19**

## F

Familiar file, 3-19, xi  
Fast Fourier Transform, 3-8  
FFT. *See* Fast Fourier Transform

File header, xi  
File name, 3-2, 3-20  
Filter Tool, 3-1, 3-4, 3-5, **3-13**, 3-14, 3-15, 3-16,  
3-17, xi  
Filter Tools, **3-13–3-15**  
Filtering. *See* Filter Tool  
Fourier coefficient, iv, 3-5

## G

Generic file, 3-1, 3-12, 3-18, 3-19, xi  
GenPro, 3-8  
Gentime Tool, 3-16  
Get-Columns Tool, 3-19, xi

## H

Hamming window, 3-9  
Help Tool, 3-2, **3-19**  
Hewlett-Packard, 1-1, 2-2  
Histogram Tool, **3-7–3-8**  
HOME environment variable, 3-20

## I

Interpolation Tool, 3-1, **3-8**, 3-14, 3-15, 3-16, 3-  
17

## L

Least-Squares-Fit Tool, **3-8**  
Limit Tool, 3-1, 3-5, 3-6, 3-7, 3-13–**3-14**, 3-15,  
3-17, xi  
Logarithmic spacing, 3-9  
Long menu. *See* Menu, Long

## M

Mathews, iv, 3-7, 3-8, ix  
Menu  
Long, **3-2**  
Merge Tool. *See* Merge Tool Menu  
Short, 3-2  
Startup. *See* Startup Menu  
Tool. *See* Tool Menu  
Merge Tool, 3-1, 3-4, 3-7, 3-8, 3-9, 3-10, 3-12, 3-  
13, **3-14–3-15**, 3-17, 3-18, 3-20, xi  
Merge Tool file, 3-4, 3-10, 3-15, xi  
Merge Tool Menu, 3-1, **3-4–3-5**, 3-13, 3-15,  
3-17, 3-20, xi  
More Tool, 3-18, **3-19–3-20**

## N

Naked return, **vii**, 3-2, **xi**  
National Center for Atmospheric Research, 3-8  
National Renewable Energy Laboratory, iv, 1-1  
NCAR. *See* National Center for Atmospheric Research  
*Numerical Recipes*, iv, 3-9

## O

Options Tool, 3-1, 3-9, 3-10, 3-18, **3-20**

## P

Parse, **xi**  
Porting GPP, **2-1-2-3**  
Power spectral density. *See* PSD Tool  
Primary array, 3-1, 3-2, 3-3, 3-4, 3-6, 3-7, 3-8, 3-12, 3-14, 3-15, 3-18, **xi**  
PSD Tool, iv, 1-1, 3-1, **3-8-3-9**, 3-20, xi

## Q

Quitting GPP, 3-20

## R

Rainflow-Cycle-Count Tool, 1-1, 3-1, **3-9-3-10**, 3-11, 3-20  
REALFT, 3-8, 3-9  
Rectangular window, 3-9  
Requirements, **1-2-1-3**  
Return  
    Naked. *See* Naked return  
Root name, 3-18, 3-19, **xi**  
Running GPP, **2-1-2-3**

## S

Scales and offsets. *See* Units-Conversion Tool  
Search path, 2-4  
Secondary array, 3-1, 3-3, 3-4, 3-8, 3-12, 3-13, 3-14, 3-15, 3-17, xi  
Short menu. *See* Menu, Short  
Silicon Graphics, Inc., 1-1  
SPECFT, 3-8, 3-9  
Startup Menu, **3-2-3-3**  
Startup Menu, 3-3  
Startup Screen. *See* Startup Menu  
Statistics Tool, 3-3, **3-10-3-12**, 3-13, 3-18  
Sun Microsystems, 1-1, 2-1  
System Tools, **3-16-3-21**

## T

**Tool Menu**, 3-1, 3-3-3-4  
Triangular window, 3-9  
Trim Tool, 3-4, 3-8, 3-13, 3-14, 3-15, 3-16, **3-15**, xi  
Trimming. *See* Trim Tool

## U

Units. *See* Units-Conversion Tool  
Units-Conversion Tool, 3-4, 3-5, 3-14, 3-15, 3-16, **3-20**, 3-17  
UNIX, vii, 1-1, **2-1**, 2-2, 2-3, **2-4**, 3-2, 3-19, 3-20  
Using GPP, **2-4-3-21**

## W

Write Tool, 3-4, 3-7, 3-8, **3-20**

## Y

YawDyn, 3-1, 3-15, 3-18, 3-20, xi



REPORT DOCUMENTATION PAGE			Form Approved OMB NO. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE July 1999	3. REPORT TYPE AND DATES COVERED Technical Report		
4. TITLE AND SUBTITLE GPP Version 6 User's Guide A General-Purpose Postprocessor for Wind-Turbine Data Analysis			5. FUNDING NUMBERS  WE901210	
6. AUTHOR(S) Marshall L. Buhl Jr., Norman L. Weaver				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Renewable Energy Laboratory 1617 Cole Blvd. Golden, CO 80401-3393			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Renewable Energy Laboratory 1617 Cole Blvd. Golden, CO 80401-3393			10. SPONSORING/MONITORING AGENCY REPORT NUMBER  TP-500-25815	
InterWeaver Consulting PO Box 775444 Steamboat Springs, CO 80477-5444				
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT National Technical Information Service U.S. Department of Commerce 5285 Port Royal Road Springfield, VA 22161			12b. DISTRIBUTION CODE	
13. ABSTRACT ( <i>Maximum 200 words</i> )  GPP (pronounced "jeep") is a general-purpose postprocessor for wind turbine data analysis. The original author, Marshall Buhl, a member of the National Wind Technology Center (NWTC) of the National Renewable Energy Laboratory (NREL), developed GPP to postprocess test data and simulation predictions. Norman Weaver of InterWeaver Consulting is responsible for coding most of the recent additions to Version 6. GPP reads data into large arrays and allows the user to run many types of analyses on the data stored in memory. It runs on inexpensive computers common in the wind industry. One can even use it on a laptop in the field.  We wrote the program in such a way as to make it easy to add new types of analyses and to port it to many types of computers. Although GPP is very powerful and feature rich, it is still very easy to learn and easy to use. Exhaustive error trapping prevents one from losing valuable work caused by input errors. We believe that GPP can make a significant impact on engineering productivity in the wind industry.				
14. SUBJECT TERMS Software, Postprocess, Wind Turbine, Data Analysis			15. NUMBER OF PAGES 47	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT  L	