

MATLAB[®]

The Language of Technical Computing

Computation

Visualization

Programming



Known Software and Documentation Problems

Release 11 (MATLAB 5.3 Product Family)

Introduction	vii
Known Software Problems	vii
Contents	viii

MATLAB 5.3

1

Documentation Updates	1-2
String Comparisons	1-2
Printing	1-2
Notebook Support	1-3
Exploratory MATLAB Java Interface	1-3
UNIX Usage Information	1-3
Known Software Problems or Limitations	1-5
Graphics	1-5
Application Program Interface	1-6
PC-Specific Problems	1-7
Help System	1-7
UNIX-Specific Problems	1-10

MATLAB Compiler and C/C++ Math Libraries

2

MATLAB Compiler	2-2
MATLAB Compiler 2.0	2-2
MATLAB Compiler 1.2.1	2-3
MATLAB C/C++ Math Library	2-4
MATLAB C/C++ Math Library 2.0	2-4
MATLAB C/C++ Math Library 1.2.1	2-5

3

Known Software Problems and Limitations	3-2
Help Button	3-2
S-Functions	3-2
Using an MGA Matrox Board	3-2
Multiport Switch Block	3-3
Configurable Subsystem Block	3-3

Stateflow 2.0

4

Known Software Problems and Limitations	4-2
Cyclic Flow Graphs	4-2
Output Event Triggers	4-2
Broadcasting Events Among Charts	4-2
Documentation Updates	4-3
lcc Compiler	4-3

Real-Time Workshop Products

5

The Real-Time Workshop 3.0	5-2
Rapid Simulation Target	5-2
DSP Processor Support	5-2
Unsupported Features in Version 3.0	5-2
Real-Time Workshop Ada Coder 3.0	5-3
External Mode Not Supported	5-3
The Real-Time Windows Target 1.0	5-3
Documentation Updates	5-3

Control System Toolbox 4.2: Documentation Updates	6-2
Nx Property Has Been Removed	6-2
Initializing an Array of SS Models with Zero Entries	6-2
Corrections to the 747 Jet Transport Example	6-3
Signal Processing Toolbox 4.2	6-4
Pole/Zero Editor (PC)	6-4
detrend Function	6-4
Documentation Updates	6-4
Fuzzy Logic Toolbox 2.0.1	6-6
Limited Real-Time Workshop Support	6-6
DSP Blockset 3.0: Documentation Updates	6-7
Configuring Simulink for DSP	6-7
Matrix From Workspace Block	6-7
Multirate Filter Blocks	6-7
Sine Wave Block	6-7
Compatibility with Real-Time Workshop Embedded-C Target .	6-8
Fixed-Point Blockset 2.0: Documentation Updates	6-9
DSP Processor Support	6-9
Fixed-Point Demos Moved	6-9
fpupdate Modified	6-9
Fixed-Point GUI Modified	6-9
Parameter Saturation Produces Warning	6-10
Signal Errors Produce Warnings or Errors	6-10
FixPt Relay Block	6-10
FixPt Relational Operator Block	6-10
FixPt Switch Block	6-10
FixPt Unit Delay	6-11
FixPt Look-Up Table (1D and 2D)	6-11
Using rsim Target or External Mode	6-11
MATLAB Web Server	6-13
Suppressing the HTTP Header with htmlrep	6-13

Excel Link 1.0.8	6-14
Excel 7 and Excel 97 Versions	6-14

How to Contact The MathWorks:



508-647-7000

Phone



508-647-7001

Fax



The MathWorks, Inc.
24 Prime Park Way
Natick, MA 01760-1500

Mail



<http://www.mathworks.com>
<ftp.mathworks.com>
<comp.soft-sys.matlab>

Web
Anonymous FTP server
Newsgroup



support@mathworks.com
suggest@mathworks.com
bugs@mathworks.com
doc@mathworks.com
subscribe@mathworks.com
service@mathworks.com
info@mathworks.com

Technical support
Product enhancement suggestions
Bug reports
Documentation error reports
Subscribing user registration
Order status, license renewals, passcodes
Sales, pricing, and general information

Known Software and Documentation Problems

© COPYRIGHT 1984 - 1999 by The MathWorks, Inc. All Rights Reserved.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

U.S. GOVERNMENT: If Licensee is acquiring the Programs on behalf of any unit or agency of the U.S. Government, the following shall apply: (a) For units of the Department of Defense: the Government shall have only the rights specified in the license under which the commercial computer software or commercial software documentation was obtained, as set forth in subparagraph (a) of the Rights in Commercial Computer Software or Commercial Software Documentation Clause at DFARS 227.7202-3, therefore the rights set forth herein shall apply; and (b) For any other unit or agency: NOTICE: Notwithstanding any other lease or license agreement that may pertain to, or accompany the delivery of, the computer software and accompanying documentation, the rights of the Government regarding its use, reproduction, and disclosure are as set forth in Clause 52.227-19 (c)(2) of the FAR.

MATLAB, Simulink, Stateflow, Handle Graphics, and Real-Time Workshop are registered trademarks, and Target Language Compiler is a trademark of The MathWorks, Inc.

Other product or brand names are trademarks or registered trademarks of their respective holders.

Printing History: January 1999 New for Release 11

MATLAB Software Acknowledgments

MATLAB and its associated products incorporate the following third-party software:

The Delaunay function is based on code from

Steve J. Fortune (1987) A Sweepline Algorithm for Voronoi Diagrams, *Algorithmica* 2, 153-174.

The HDF capability in the functions `imread`, `imwrite`, `imfinfo`, and `hdf` is based on code, of which portions were developed at the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign.

The JPEG capability in the functions `imread`, `imwrite`, `imfinfo`, `print`, and `saveas`:

This software is based in part on the work of the Independent JPEG Group.

The TIFF capability in the functions `imread`, `imwrite`, `imfinfo`, `print`, and `saveas`:

© Copyright 1988-1999 Sam Leffler

© Copyright 1991-1999 Silicon Graphics, Inc.

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Introduction

This *Known Software and Documentation Problems* document describes Release 11 known software and documentation problems. This document also contains information that was not available at the time the rest of the Release 11 documentation set was released. The information in this document takes precedence over any information in the rest of the printed or online documentation.

Known Software Problems

Unfortunately, all software of the complexity of MATLAB and other Release 11 products ends up having known software problems at the time it is released. Although The MathWorks makes every effort to address as many known software problems as it can before releasing its software, we must manage the balance between:

- Quality
- Features
- Release Date

These three aspects of a software release form a triangle in which efforts to address one aspect affect the other two aspects. For example, addressing all known software problems extends the release date and reduces the number of new features provided. We have extended our release date for Release 11 as part of a major effort to improve the quality of our software.

However, to get the bug fixes and the new features included in Release 11 to you in a timely fashion requires that we ship the release with some known software problems discovered too late in the process to fix without destabilizing the product.

Known Software and Documentation Problems describes the known software problems that may have the most impact to you.

Contents

Known Software and Documentation Problems is organized into the following chapters:

- Chapter 1, MATLAB® 5.3
- Chapter 2, MATLAB Compiler and C/C++ Math Libraries
- Chapter 3, Simulink® 3.0
- Chapter 4, Stateflow® 2.0
- Chapter 5, Real-Time Workshop® Products
- Chapter 6, Other MathWorks Products

MATLAB 5.3

Documentation Updates	1-2
String Comparisons	1-2
Printing	1-2
Notebook Support	1-3
Exploratory MATLAB Java Interface	1-3
UNIX Usage Information	1-3
 Known Software Problems or Limitations	 1-5
Graphics	1-5
Application Program Interface	1-6
PC-Specific Problems	1-7
Help System	1-7
UNIX-Specific Problems	1-10

Documentation Updates

String Comparisons

When comparing a string array to a cell or cell array, the string array is debanked (trailing spaces are removed) before comparison. However, when comparing a scalar string (a string array with only one row) to a cell or cell array, the scalar string is not debanked. Use the `deblank` function to eliminate any trailing blanks in the scalar string before comparison.

This example compares a scalar string to a cell

```
strcmp('hello ', {'hello','world'})
ans =
     0     0
strcmp(deblank('hello '), {'hello','world'})
ans =
     1     0
```

Note The HTML version of the MATLAB Function Reference reflects this information. However, the PDF version has not been updated for Release 11. The PDF version of the documentation only states that “When comparing a string array to a cell or cell array, the string array is debanked (trailing spaces are removed) before comparison.”

Printing

Printing MATLAB graphics in a variety of configurations can be quite complex. Release 11 has enhanced the software to make printing simpler and more reliable.

In addition, the MATLAB documentation about printing has been enhanced to include specific examples of including MATLAB graphics in word processing applications. See the “Printing MATLAB Graphics” section of the HTML version of *Using MATLAB Graphics*.

For assistance with printing issues, you can contact our Technical Support team from The MathWorks Web page at <http://www.mathworks.com>.

Notebook Support

For Office 97 on Windows 95

For Microsoft Windows 95, due to an Office 97 problem, printing a Notebook document that includes an imported graphic may not print correctly. See “OFF97: Imported EMF Files Are Not Printed Correctly” in the online Microsoft Knowledge Base for details.

The MATLAB 5.2 Notebook is fully supported for Microsoft Windows NT with Microsoft Office 97.

Microsoft Word 6.0 No Longer Supported

You must use either Microsoft Word 95 (Word 7.0) or Word 97.

Exploratory MATLAB Java Interface

The description of the exploratory MATLAB Java Interface on page 1-14 of the *Release 11 New Features* document indicates how to start that interface. Note that once enabled, the MATLAB Java Interface is available throughout the current MATLAB session. To close it, restart MATLAB. There is no `java off` command.

UNIX Usage Information

DEC UNIX Recursion Limit

On DEC UNIX systems, the `csh` built-in command `limit` or the current shell equivalent can be used to modify the user stack size limit up to a maximum of 32 MB in the user’s current process context.

The system-wide default value can be changed by someone with root access, using the `dxkerneltuner` command to modify the parameters in the `proc` subsystem. That change will take effect after a reboot.

Help Desk Search on DEC Alpha Running Netscape Navigator

Netscape Navigator 3.0 does not correctly implement Java on the DEC Alpha platform. Therefore, the search utility of the MATLAB Help Desk does not run on DEC Alpha under the Netscape 3.0 browser. You can use the MATLAB `lookfor` function to search for command line help by topic.

Adobe Acrobat Reader Not Supported for Xoftware

The Adobe Acrobat Reader, which is required to access the PDF files available through the Help Desk, is not supported for Xoftware.

Known Software Problems or Limitations

This section describes MATLAB 5.3 known software problems, providing workarounds for most problems.

Graphics

Printing with the HPGL Driver

The HPGL -dhpgl print driver does not perform clipping in MATLAB 5.3. As a result, the text labeling port names in Simulink blocks can extend beyond the rectangle, enclosing the block. Also, zooming in on line plots can produce lines that are not clipped to the axes box, (that is, it looks as if the line object's Clipping properties are set to off).

Microsoft Word users who import HPGL illustrations into documents will see no difference, because Word does not honor clipping.

dragrect and rbbox Functions

The dragrect function assumes that rectangles passed to it are specified in pixel units, not current figure units. To avoid incorrect scaling of rectangles when using dragrect, specify arguments in pixel units.

In function rbbox(initialRect,fixedPoint,stepSize), the stepSize argument is in pixel units. Arguments initialRect and fixedPoint are in the current figure units.

MinorGridLine Style Property Not Supported

The axes MinorGridLineStyle property, which is mentioned in Chapter 10 of the MATLAB 5.0 *Using MATLAB Graphics* manual and returned when you query axes properties, is not supported.

Application Program Interface

Avoid Modifying Input Arguments in MEX-Files

In MATLAB 5.1 to 5.3, MATLAB arrays can share data. There is currently no way for a MEX-file to determine that an array contains shared data. MEX-files that modify their input arguments may corrupt arrays in the MATLAB workspace. This style of programming is strongly discouraged.

Powerstation Fortran No Longer Supported

Powerstation Fortran 4.0 is no longer supported for MEX-files. DEC Visual Fortran 5.0 is supported.

Floating-Point Exceptions and Third-Party Libraries

MATLAB-based applications and MEX-files should never get floating-point exceptions. If you do get a floating-point exception, it may be that a third-party library that you link against has enabled floating-point exception handling. A floating-point exception may also occur if you use the wrong compiler options when building. For additional information, see the floating-point exception sections in the *Application Program Interface Guide*.

mexErrMsgTxt Warnings

If you get warnings when using `mexErrMsgTxt`, you may have a memory management compatibility problem. For more information, see “Memory Management Compatibility Issues” in the *Application Program Interface Guide*.

Microsoft Visual C++ Version 6.0 For Use with mex

You can use the Microsoft Visual C++ Version 6.0 compiler with `mex`. When installing Microsoft Visual C++ 6.0, if you need to change where this compiler is installed, you must change the location of the Common directory (at the appropriate installer dialog). If you change the location of the VC98 directory from its default setting, the `mex` script will not work properly.

MATLAB 4 MEX-Files

Beginning with MATLAB 5.2, MEX-files from MATLAB 4 do not run as is. You must recompile them in MATLAB 5.3 using the `-V4` option of `mex` and a supported compiler.

Application Program Interface Guide

In the “Custom Building MEX-Files” section of the “System Setup” chapter, the MEX Script Switches table includes a `-n` switch. This switch does not exist.

PC-Specific Problems

The following problems apply to Microsoft Windows 95 and 98, as well as Windows NT platforms running MATLAB, unless otherwise indicated.

Help System

The setup program for Microsoft Visual C++ 6.0 or Visual J++ 6.0 installs a defective version of Microsoft’s new HTML Help viewer. Some MATLAB components and toolboxes use the HTML Help viewer by default to display help. As a result, if either Visual C++ 6.0 or Visual J++ 6.0 is installed on your system, the MATLAB components and MathWorks products that use the viewer may display an error message when you invoke online help. The error message typically occurs when the help system attempts to display a single page of help (as opposed to a collection of help pages). The MATLAB components and MathWorks products affected by this problem are the Plot Editor, **Page Layout** dialog box, Report Generator, Simulink, and Stateflow.

Microsoft has an update to the HTML Help viewer that fixes this problem. The update is available as a self-installing program at

<http://www.microsoft.com/workshop/author/htmlhelp/localize.asp>

To install the update, download and run the self-installing program appropriate to your locale (for example, English). The whole process should take less than a minute.

If you experience any other problems with the HTML Help viewer or prefer to use a Web browser to view MATLAB documentation, you can disable use of the HTML Help viewer by resetting the `matlab_use_html_help_viewer` environment variable. Setting this variable to no causes MATLAB to display all help in the default Web browser on your system.

Set the environment variable on a Windows 95 or 98 system by adding the following line

```
set matlab_use_htmlhelp_viewer=no
```

to your system's `autoexec.bat` file and rebooting your system. The `autoexec.bat` file is located in the root directory on your system's boot drive (typically, `c:\`).

On a Windows NT system, use the **Environment** pane of the Windows NT **System Properties** dialog box to set the environment variable. To display the **System Properties** dialog box, double-click the System folder in the Control Panel folder in the My Computer folder on your Windows NT desktop.

Avoid Spaces in Directory Names and Filenames

To ensure predictable results when running MATLAB on the PC, avoid using spaces in directory names and filenames for the locations you use for installing MATLAB and its ancillary products such as the compilers used for building MEX-files.

Following this recommendation shields you from a limitation in the Windows 95 and Windows NT command environment. If you do use spaces in such directory names or filenames, you may encounter problems stemming from the Windows command environment, such as messages about being unable to find files that are actually where you think they are.

Graphics Display Problems

If graphics objects in a MATLAB figure do not display correctly when the figure is updated during animations or when you drag objects, you may need to turn off hardware acceleration. Settings for controlling hardware acceleration are generally accessible from the Microsoft Windows 95 or NT Control Panel, in the Display settings. Consult your graphics hardware manual or contact your hardware vendor directly for information on making adjustments to your graphics hardware.

Tooltips Do Not Work on Windows 95 with comctl32.dll Version 4.00.950

The tooltip feature introduced in MATLAB 5.2 does not work with a certain configuration of Microsoft Windows 95 because of a bug in that operating system.

In particular, tooltips do not work on computers having version 4.00.950 of `comctl32.dll`. To determine which version of `comctl32.dll` you have, run the Microsoft Windows Explorer. In the Win95 directory, display the files in the System directory. Right-click on `comctl32.dll` and select **Properties**. Then, select the **Version** tab to see the version number.

Microsoft Internet Explorer 3 upgrades comctl32.dll to version 4.70; Internet Explorer 4 upgrades the file to version 4.71. Tooltips work with either version. Also, more recent versions of Windows 95 correct this problem. You can access the complete self-extracting archive file Com32upd.exe at

<http://support.microsoft.com/download/support/mslfiles/Com32upd.exe>

Note that even though your computer may have an upgraded version of comctl32.dll, if you are writing an application to be used by other users, they will not be able to take advantage of tooltips if their computers have the older version of the file.

Antivirus Software Can Conflict with mex and mbuild -setup Process

Certain PC antivirus software packages, such as Cheyenne AntiVirus and Dr. Solomon, can conflict with the mex -setup and mbuild -setup process. If during the setup of mex or mbuild, you receive an error message of this form

```
mex.bat: internal error in sub get_compiler_info(): don't  
recognize <string>
```

then you need to disable your antivirus software temporarily and rerun with the -setup option. After you have successfully rerun the setup operation, you can reenable your antivirus software.

Netscape Navigator

If you try to access (e.g., via the helpdesk command or the Simulink **Help** button) the HTML online documentation via the Netscape Navigator browser and the documentation does not appear, open the browser manually and then access the online documentation. In general, MATLAB automatically starts the browser for you, but in some Netscape Navigator installation configurations, the browser does not start automatically.

Note that not all earlier releases of Netscape Navigator Version 4 work well with the Help Desk; Netscape 4.0.4 does work well.

UNIX-Specific Problems

Graphical Installation Script May Not Work in Some Environments

The graphical installation script for Release 11 may not work properly under some UNIX X Window System environments. An example of a known environment in which the graphical installation script does not work is Redhat Linux 5.2 systems with the XFree86 3DLabs driver.

As a workaround, use the text-based installation script instead of the graphical installation script. To do so, type

```
install -t
```

and follow the text-based installation prompts on your screen. Those prompts are very similar to those used in the graphical installation script.

MATLAB Compiler and C/C++ Math Libraries

MATLAB Compiler	2-2
MATLAB Compiler 2.0	2-2
MATLAB Compiler 1.2.1	2-3
 MATLAB C/C++ Math Library	 2-4
MATLAB C/C++ Math Library 2.0	2-4
MATLAB C/C++ Math Library 1.2.1	2-5

MATLAB Compiler

MATLAB Compiler 2.0

Two Versions of the Compiler

Release 11 includes both the MATLAB Compiler 2.0 and the MATLAB Compiler 1.2.1.

The MATLAB Compiler 2.0 is used by default. However, you may want to use the 1.2.1 version of the Compiler to take advantage of some of the optimizations in the MATLAB Compiler 1.2.1. See the Readme file for details.

See the Readme File

Additional information about the MATLAB Compiler is available in the following file:

```
toolbox/compiler/Readme.m
```

Don't Install Your C/C++ Compiler in a Directory Whose Name Includes Spaces

The MATLAB 2.0 Compiler may fail to build your executables if the C/C++ compiler you are using with the MATLAB Compiler is installed in a directory whose name contains spaces.

Watcom 10.6 Compiler

If you are using Watcom 10.6, you cannot install the MATLAB C/C++ Math Library in a directory whose name contains special characters such as dashes or dollar signs. For example, installing the MATLAB C/C++ Library in a directory named `matlab5-3-c` would cause a failure because Watcom 10.6 truncates this directory name, which causes problems when trying to build standalone applications.

Microsoft Visual C++ Version 6.0 For Use with `mex` or `mbuild`

You can use the Microsoft Visual C++ Version 6.0 compiler with `mex` or `mbuild`. When installing MSVC 6.0, if you need to change where this compiler is installed, you must change the location of the Common directory (at the appropriate installer dialog). If you change the location of the VC98 directory from its default setting, the `mex` and `mbuild` scripts will not work properly.

MATLAB Compiler 1.2.1

General Information

The MATLAB Compiler 1.2.1 contains numerous bug fixes for small problems.

Note The Watcom and the Microsoft Visual C++ compiler issues discussed above, in the “MATLAB Compiler 2.0” section, also apply to the MATLAB Compiler 1.2.1.

Linking MFC Applications with the C++ Math Library

In order to allow Microsoft Foundation Class (MFC) applications to link against the C++ Math Library, we build the Microsoft Visual C (MSVC) version of the library against the multithreaded DLL versions of the MSVC runtime libraries. The impact of this change is as follows:

- No changes in the way you use `mbuild`.
- For MFC applications in the Visual C++ IDE, MFC must be used as a DLL (the default).
- For Console applications in the Visual C++ IDE, change the project settings for C/C++.
- For Code Generation / Runtime Library, change the Code Generation/ Runtime Library project setting for C/C++ to Multithreaded DLL or Debug Multithreaded DLL
- If you are building from the command line without `mbuild`, use `-MD` option when compiling

Additionally, the MSVC runtime libraries, `msvcrt.dll` and `msvcirt.dll`, need to be distributed with any applications created with the C++ Math Library on Visual C++.

Runtime Errors Involving Operands

The Compiler may not always detect runtime errors involving incorrectly typed operands. If you suspect that an error might be occurring within your code, check the MATLAB version first. If the MATLAB version produces no errors, then the Compiler-generated code will work correctly.

MATLAB C/C++ Math Library

MATLAB C/C++ Math Library 2.0

Two Versions of the Library

Release 11 includes both the MATLAB C/C++ Math Library 2.0 and the MATLAB C/C++ Math Library 1.2.1.

If you are upgrading from the MATLAB 1.2.1 C Math Library, for 2.0 you are getting an upgrade that includes the MATLAB 2.0 C++ Math Library as well, for no additional charge.

The MATLAB C/C++ Math Library 2.0 is used by default. However, if you are using existing code that depends on the MATLAB 5.2 Application Program Interface (API), then you should use the MATLAB C/C++ Math Library 1.2.1.

See the Release Notes Files

Additional information about the MATLAB C Math Library is available in the following file:

```
extern/examples/cmath/release.txt
```

Additional information about the MATLAB C++ Math Library is available in the following file:

```
extern/examples/cppmath/release.txt
```

Watcom 10.6 Compiler

If you are using Watcom 10.6, you cannot install the MATLAB C/C++ Math Library in a directory whose name contains special characters such as dashes or dollar signs. For example, installing the MATLAB C/C++ Library in a directory named `matlab5-3-c` would cause a failure because Watcom 10.6 truncates this directory name, which causes problems when trying to build standalone applications.

Microsoft Visual C++ Version 6.0 For Use with `mex` or `mbuild`

You can use the Microsoft Visual C++ Version 6.0 compiler with `mex` or `mbuild`. When installing MSVC 6.0, if you need to change where this compiler is installed, you must change the location of the Common directory (at the

appropriate installer dialog). If you change the location of the VC98 directory from its default setting, the `mex` and `mbuild` scripts will not work properly.

MATLAB C/C++ Math Library 1.2.1

General Information

The MATLAB C/C++ Math Library 1.2.1 contains numerous bug fixes for small problems.

Note The Watcom and the Microsoft Visual C++ compiler issues discussed above, in the “MATLAB C/C++ Math Library 2.0” section, also apply to the MATLAB C/C++ Math Library 1.2.1.

Recompiling Your Application

If you are a MATLAB C/C++ Math Library Version 1.2 user, you must recompile your existing applications with the 1.2.1 versions of the header files and libraries.

Improved Usability for mbuild

The `mbuild` tool now automatically detects:

- The location of your C/C++ compiler
- The language of the files you are compiling, based on file extension

The `-setup` option is still available, but you no longer need to run `mbuild -setup` to build your first application or to change languages.

Options files are no longer language-dependent.

Linking with libmmfile

If your application calls MATLAB M-File Math Library functions, you must link with `libmmfile`. In previous versions, linking with `libmmfile` was optional.

Note that this release reduces the size of `libmmfile`, so your application will require less memory.

Linking MFC Applications with the MATLAB C/C++ Math Library

In order to allow MFC applications to link against the MATLAB C++ Math Library, we build the MSVC version of the library against the multithreaded DLL versions of the MSVC runtime libraries. The impact of this change is as follows:

- No changes in the way you use `mbuild`.
- For MFC applications developed in the Visual C++ IDE, MFC must be used as a DLL (the default).
- For console applications developed in the Visual C++ IDE, Change the Code Generation/Runtime Library project setting for C/C++ to Multithreaded DLL or Debug Multithreaded DLL.
- If you are building from the command line without `mbuild`, you need to use the `-MD` option when compiling

Additionally, the MSVC runtime libraries, `msvcrt.dll` and `msvcirt.dll`, need to be distributed with any applications created with the MATLAB C++ Math Library on Visual C++.

Simulink 3.0

Known Software Problems and Limitations	3-2
Help Button	3-2
S-Functions	3-2
Using an MGA Matrox Board	3-2
Multiport Switch	3-3
Configurable Subsystem Block	3-3

Known Software Problems and Limitations

Help Button

Under Microsoft Internet Explorer 4, the Simulink HTML online documentation does not display if invoked via a **Help** button on a Simulink dialog box when the online documentation is installed on a network drive separate from where your temp directory is located. This is due to new cross-frame security restrictions in Internet Explorer 4. You can, however, access the same information through the Help Desk.

S-Functions

You must recompile any S-functions written using a version of Simulink 2.0.

Any S-function that indicates it has no inputs and/or outputs causes Simulink to remove the corresponding port or ports from the S-Function block icon. This behavior is different than in Simulink 2.0.

The ability to use a block diagram as an S-function is not implemented in Simulink 3.0. There is no workaround; work is in progress to reimplement this functionality or an equivalent for a later release.

Note You can use almost all Simulink 2.2 S-functions with Simulink 3.0 without recompiling them. However, in rare cases you may receive an message indicating you need to recompile a Simulink 2.2 S-function.

Using an MGA Matrox Board

If your computer has an MGA Matrox board, you might find that some circles (Simulink block outlines) are drawn in the wrong window when opaque dragging is in effect and you drag a window over a Simulink model window. If this is the case, right-click on the desktop, select the **Properties** item from the pop-up menu, and then the **MGA Settings** tab. Select the **Advanced** option, and then the **Performance** tab. Do *not* select the **Circle and Ellipse Acceleration** option.

Multiport Switch Block

The behavior of the Multiport Switch block has changed.

In Simulink 2.2 and earlier, the Multiport Switch block control port used to be rounded to the nearest integer. If the resulting integer was out of range, the result was bound to the nearest end point in the list of indexes.

In Simulink 3.0, the Multiport Switch block control port converts double precision to an integer index via a cast to an integer that is equivalent to using the `floor` operation. If the control input is out of range during a simulation, an error is generated. If the control input is within range *and* if the `floor` operation causes truncation to occur, a warning is generated.

Configurable Subsystem Block

If a configurable subsystem points to a library that has blocks with identically named parameter variables, the variables may not be set properly. The solution is to rename the parameter variables so that they are unique.

Stateflow 2.0

Known Software Problems and Limitations	4-2
Cyclic Flow Graphs	4-2
Output Event Triggers	4-2
Broadcasting Events Among Charts	4-2
 Documentation Updates	 4-3
lcc Compiler	4-3

Known Software Problems and Limitations

Cyclic Flow Graphs

Executing a cyclic flow graph can put Stateflow into an endless loop. To prevent this from happening, make sure that any cyclic flow graphs in your model contain an unconditional exit transition.

Output Event Triggers

Stateflow 2.0 no longer allows output event triggers to be **Rising Edge** or **Falling Edge**, because of problems in propagating successive single-edge triggers. When loading a model created by previous versions, Stateflow 2.0 issues a warning whenever it detects a single-edge output event trigger and converts the trigger to **Either**. To maintain the model's behavior, use the Stateflow Explorer to convert the trigger inputs that are connected to the outputs to **Either**.

Broadcasting Events Among Charts

Stateflow 2.0 no longer allows direct broadcast of events from one chart to another. Version 2.0 disallows direct broadcasting of interchart events to permit incremental code generation. If you need to broadcast an event from one chart to another, create a function-call output event in the source chart and a function-call input event in the target chart. Connect the corresponding ports on the source and target charts.

Documentation Updates

lcc Compiler

Stateflow 2.0 is shipped with the lcc compiler. The lcc compiler is documented in the *Stateflow User's Guide*, in Chapter 9, "Building Targets."

You can access online documentation for the Microsoft Windows 95 and NT versions of the lcc compiler by going to the `\sys\lcc\bin` in the MATLAB root directory and, in a DOS window, executing `lcc.exe`.

For a general information about the lcc compiler, you may want to refer to a book such as *A Retargetable C Compiler: Design and Implementation* by Christopher Fraser and David Hanson (Addison-Wesley) (ISBN 0-8053-1670-1).

Real-Time Workshop Products

The Real-Time Workshop 3.0	5-2
Rapid Simulation Target	5-2
DSP Processor Support	5-2
Unsupported Features in Version 3.0	5-2
 Real-Time Workshop Ada Coder 3.0	5-3
External Mode Not Supported	5-3
 The Real-Time Windows Target 1.0	5-3
Documentation Updates	5-3

The Real-Time Workshop 3.0

Rapid Simulation Target

The Rapid Simulation Target is used for accelerating simulations and for batch simulations. Its usage is slightly different from what is documented in the *Real-Time Workshop User's Guide*.

- The Rapid Simulation Target supports all built-in data types for Simulink parameters. The *Real-Time Workshop User's Guide* incorrectly states that the Rapid Simulation Target only supports block diagrams with double parameters.
- You change the values of From Workspace blocks the same way that you change parameters (i.e., use the `rsimgetrttp` command, save the parameter vector to a `.mat` file, which will include the From Workspace data, and use the `-p` switch on the generated executable). Ignore the sections of the manual that discuss using the `-w` switch.

DSP Processor Support

Real-Time Workshop 3.0 now supports target processors that have only one register size (e.g., 32-bit). This supports data type emulation of 8 and 16 bits on the TCI_C30/C40 DSP and similar processors.

To do this, add `-DDSP32=1` to your template makefile and add `%assign DSP32=1` to your system target file.

Unsupported Features in Version 3.0

Real-Time Workshop 3.0 does *not* currently support:

- Nonreal-time simulation of variable-step solvers
- The Simulink Accelerator

The Fuzzy Logic Toolbox is compatible with the Real-Time Workshop simulation targets only. It is not compatible with hardware targets. The generated fuzzy code is not optimized and will not run on fixed-point embedded microcontrollers (floating-point only). In addition, the generated code may not run on single precision processors such as the Texas Instruments C30/C40 class of DSPs.

Prior to Release 11, Real-Time Workshop provided some support for the integrated use of the Fuzzy Logic Toolbox. In Release 11, there is no Real-Time Workshop support for the Fuzzy Logic Toolbox.

Real-Time Workshop Ada Coder 3.0

External Mode Not Supported

The Real-Time Workshop Ada Coder does not support external mode.

The Real-Time Windows Target 1.0

Documentation Updates

The printed *Real-Time Windows Target User's Guide* has some minor errors and omissions:

- The Real-Time Windows Target supports Microsoft Windows NT 4.0 (not mentioned in the documentation) and the Fixed-Point Blockset (the documentation said that the Fixed-Point Blockset was not supported).
- The following board names are incorrect on page 1-13:
 - For Computer Boards, the board name for driver cdas1606 should be CIO-DAS 1602/16, not CIO-DAS-1606/16
 - For Keithley-Metrabyte, the board name for driver das1401 should be DAS-1401, not DAS-1310. The board name for the dda06 driver should be DDA-06, not DAS-96.

Other MathWorks Products

Control System Toolbox 4.2: Documentation Updates	6-2
Nx Property Has Been Removed	6-2
Initializing an Array of SS Models with Zero Entries	6-2
Corrections to the 747 Jet Transport Example	6-3
Signal Processing Toolbox 4.2	6-4
Pole/Zero Editor (PC)	6-4
detrend Function	6-4
Documentation Updates	6-4
Fuzzy Logic Toolbox 2.0.1	6-6
Limited Real-Time Workshop Support	6-6
DSP Blockset 3.0: Documentation Updates	6-7
Configuring Simulink for DSP	6-7
Matrix From Workspace Block	6-7
Multirate Filter Blocks	6-7
Sine Wave Block	6-7
Compatibility with Real-Time Workshop Embedded-C Target	6-8
Fixed-Point Blockset 2.0: Documentation Updates	6-9
DSP Processor Support	6-9
Fixed-Point Demos Moved	6-9
fpupdate Modified	6-9
Fixed-Point GUI Modified	6-9
Parameter Saturation Produces Warning	6-10
Signal Errors Produce Warnings or Errors	6-10
FixPt Relay Block	6-10
FixPt Relational Operator Block	6-10
FixPt Switch Block	6-10
FixPt Unit Delay	6-11
FixPt Look-Up Table (1D and 2D)	6-11
Using rsim Target or External Mode	6-11
MATLAB Web Server	6-13
Suppressing the HTTP Header with htmlrep	6-13
Excel Link 1.0.8	6-14
Excel 7 and Excel 97 Versions	6-14

Control System Toolbox 4.2: Documentation Updates

Nx Property Has Been Removed

In the lists of state-space (SS) model properties on pages 2-29 and 11-196 of the Release 11 version of the *Control System Toolbox User's Guide*, a property called Nx is listed. This property is *not* shipping with the Control System Toolbox 4.2 software.

To access the number(s) of states in an SS model or an LTI array of SS models, use

```
size(sys, 'order')
```

In the case that sys is an LTI array of SS models with a different number of states in each model,

```
size(sys, 'order')
```

returns an array whose entries correspond to the number of states in each model in the LTI array.

Initializing an Array of SS Models with Zero Entries

To initialize an array of SS models with zero entries, such that:

- Each model in the array has Ny outputs and Nu inputs.
- The array has size [S1 S2 ... Sn].

use

```
sys = ss(zeros([Ny Nu S1 S2 ... Sn]))
```

Note that the syntax

```
sys = ss(zeros([Ny Nu S1 S2 ... Sn]),Nx)
```

reported on page 4-15 of the Release 11 version of the *Control System Toolbox User's Guide* is *not* correct.

Corrections to the 747 Jet Transport Example

There are two small changes to the 747 jet transport example described on page 9-3 of the “Design Case Studies” chapter of the *Control System Toolbox User’s Guide*. You can also make these changes to the M-file `jetdemo` if you want to use the demo to follow along with the text.

The first change is in the values of the B matrix. The result is that the state-space matrices for the 747 jet model during cruise flight at $\text{MACH} = 0.8$ and $H = 40,000$ feet are

```
A = [ -0.0558   -0.9968    0.0802    0.0415
        0.5980   -0.1150   -0.0318         0
       -3.0500    0.3880   -0.4650         0
         0         0.0805    1.0000         0];
```

```
B = [ 0.0729    0.0000
      -4.7500    0.00775
        1.5300    0.1430
         0         0];
```

```
C = [0    1    0    0
      0    0    0    1];
```

```
D = [0    0
      0    0];
```

This change modifies the command-line outputs for the remainder of the demonstration, but does not otherwise change the results of the example qualitatively.

The other modification to this model is that the units of the control inputs (rudder and aileron deflections) are measured in radians, not in degrees.

Signal Processing Toolbox 4.2

Pole/Zero Editor (PC)

Under Microsoft Windows 95/98, the Pole/Zero Editor may not display zeros as they are being repositioned by dragging with the mouse. The zeros *are* displayed properly once the mouse button is released. This issue, and a workaround, are described more completely in the section “Graphics Display Problems” on page 1-8.

detrend Function

The detrend function is now a part of the standard MATLAB language, and resides in the `toolbox\matlab\datafun` directory. To view the latest documentation for detrend, at the command line type

```
doc detrend
```

Documentation Updates

New Functions

The table below lists Release 11 functions that did not make it into the online reference documentation. To view the command-line help for one of these functions, at the command line type

```
help function
```

Name	Purpose
is2rc	Convert inverse sine parameters to reflection coefficients.
lar2rc	Convert log area ratios to reflection coefficients.
lsf2poly	Convert line spectral frequencies to prediction polynomial.
peig	Estimate the power spectral density using the eigenvector method.
poly2lsf	Convert prediction polynomial to line spectral frequencies.
rc2is	Convert reflection coefficients to inverse sine parameters.

Name	Purpose
rc2lar	Convert reflection coefficients to log area ratios.
seqperiod	Find minimum-length repeating sequence in a vector.
wavplay	Play sound using Windows audio output device.
wavrecord	Record sound using Windows audio input device.

Fuzzy Logic Toolbox 2.0.1

Limited Real-Time Workshop Support

The Fuzzy Logic Toolbox is compatible with the Real-Time Workshop simulation targets only. It is not compatible with hardware targets. The generated fuzzy code is not optimized and will not run on fixed-point embedded microcontrollers (floating-point only). In addition, the generated code may not run on single precision processors such as the Texas Instruments C30/C40 class of DSPs.

DSP Blockset 3.0: Documentation Updates

This section includes additional information about the DSP Blockset, including information about compatibility issues, that is not included in the *Release 11 New Features* document.

Configuring Simulink for DSP

DSP Blockset 3.0 includes a startup script called `dspstartup.m` that configures the Simulink environment for DSP systems. It automates a number of typical tasks that you probably now do manually, such as setting the solver type to **Fixed step** and the **Stop time** to `inf`.

You can run this script from the MATLAB command line at any time by typing

```
dspstartup
```

To automatically configure the Simulink environment at startup, add a call to `dspstartup.m` from your `startup.m` file. For more information, see the reference for `dspstartup` in the *DSP Blockset User's Guide*.

Matrix From Workspace Block

The **Sample time** parameter of the Matrix From Workspace block (in the Sources library) now requires a single *scalar* value to define the block's sample period. In previous releases, the **Sample time** parameter accepted a length-N vector containing the actual sample times corresponding to each of the N consecutive matrix outputs.

Matrix From Workspace blocks from earlier versions of the DSP Blockset will continue to function in Simulink 3.0 models. However, if you replace them with the 3.0 version of the block, you should remember to change the **Sample time** parameter to the appropriate scalar value.

Multirate Filter Blocks

The delays introduced by the blocks in the Multirate Filters library may have changed. Consult the *DSP Blockset User's Guide*.

Sine Wave Block

The *Release 11 New Features* document incorrectly refers to the Sine Wave block as the Vectorized Sine Wave block.

Compatibility with Real-Time Workshop Embedded-C Target

Certain blocks in the DSP Blockset, listed in Table 6-1 below, cannot currently generate code in the Real-Time Workshop Embedded-C Target format. However, these blocks *can* generate code in all other Real-Time Workshop formats.

Table 6-1: DSP Blocks Not Compatible with Embedded-C Target

Block Library	Block Name
DSP Sources	From Wave Device
	From Wave File
	N-Sample Enable
	Sine Wave
DSP Sinks	Matrix To Workspace
	To Wave Device
	To Wave File
Matrix Functions	Matrix To Workspace
Linear Algebra	Backward Substitution
	Cholesky Factorization
	Forward Substitution
	LDL Factorization
	Levinson Solver
	LU Factorization
	QR Factorization
	QR Solver
	Reciprocal Condition
Transforms	DCT
	FFT
	IDCT
	IFFT
Buffers	Queue
	Stack
Switches and Counters	Multiphase Clock
	N-Sample Enable
Parametric Estimation	Burg AR Estimator

Fixed-Point Blockset 2.0: Documentation Updates

The Fixed-Point Blockset updates described below are incorporated into the PDF and HTML online documentation. These updates are not present in the *Fixed-Point Blockset User's Guide*.

DSP Processor Support

When using the Fixed-Point Blockset with Real-Time Workshop 3.0, you can use target processors that have only one register size (e.g., 32-bit). This supports data type emulation of 8 and 16 bits on the TCI_C30/C40 DSP and similar processors.

To do this, add `-DDSP32=1` to your template makefile and add `%assign DSP32=1` to your system target file.

Fixed-Point Demos Moved

The Fixed-Point Blockset demos were moved from `toolbox/fixpoint/demos` to `toolbox/fixpoint/fxpdemos`. Most of the demo names were changed to include `fxp` at the beginning of the name.

fpupdate Modified

The description of `fpupdate` given on page 8-6 of the Release 11 version of the *Fixed-Point Blockset User's Guide* is incomplete. For a complete description of this function, type `help fpupdate` at the MATLAB command line.

Fixed-Point GUI Modified

The Fixed-Point Blockset GUI was modified in these ways:

- Buttons are repositioned on the GUI.
- For each block in the model that logs data, the GUI now displays the block descriptive name, the minimum simulation value, the maximum simulation value, the maximum absolute error as a percentage of range, the data type, and the scaling.
- You can now display a block's dialog box by double-clicking on the appropriate GUI entry.

For a complete description of the interface, invoke the GUI and select the **Help** button.

Parameter Saturation Produces Warning

Whenever a Fixed-Point Blockset parameter saturates, a warning is displayed.

Signal Errors Produce Warnings or Errors

If a block is configured for logging, the maximum absolute error for each block is logged in addition to the minimum and maximum simulation values. If the maximum absolute error exceeds 50% of the output range, then a warning, an error, or nothing is issued depending on the setting of the **Integer Overflow** mode under **Simulation>Parameters>Diagnostics**.

The maximum absolute error represents an error due to online calculations within a block. The maximum absolute error excludes effects due to quantization of the parameters and inputs. Online calculations are those performed by the fixed-point processor.

FixPt Relay Block

The FixPt Relay block has a constant vector scaling mode. The best precision is based on the **Output when on** and **Output when off** parameter values. Additionally, this block no longer has a logging option and will not have its scaling set by the automatic scaling script `autofixexp`.

For information about the constant vector scaling mode, refer to page 3-11 of the Release 11 version of the *Fixed-Point Blockset User's Guide*.

FixPt Relational Operator Block

The FixPt Relational Operator block uses a new data type conversion rule.

The second input of the block is not always converted to the data type of the first input before comparison. Instead, the input with the smallest positive range is converted to the data type of the other input prior to comparison.

FixPt Switch Block

The FixPt Switch block uses a new rule for setting the output data type.

The first input no longer specifies the output data type. Instead, if input 1 has a larger positive range than input 3, then it specifies the output data type. Otherwise input 3 specifies the output data type.

FixPt Unit Delay

The FixPt Unit Delay block accepts continuous sample times. When it has a continuous sample time, the block is equivalent to a Memory block.

FixPt Look-Up Table (1D and 2D)

For the FixPt Look-Up Table and FixPt Look-Up Table (2D) blocks, the automatic scaling script `autofixexp` employs a special rule to avoid parameter saturation errors.

`autofixexp` modifies the scaling by using the output look-up values in addition to the logged minimum and maximum simulation values. This prevents the data from being saturated to different values.

For the FixPt Look-Up Table block, the look-up values are given by the **Vector of output values** parameter (the `YDataPoints` variable). For the FixPt Look-Up Table (2D) block, the look-up values are given by the **Table** parameter (the `TableDataPoints` variable).

The first input no longer specifies the output data type. Instead, if input 1 has a larger positive range than input 3, then input 1 specifies the output data type. Otherwise input 3 specifies the output data type.

Using rsim Target or External Mode

If you are using the Real-Time Workshop external mode or rapid simulation (`rsim`) target, there are situations where you may get unexpected errors when tuning block parameters.

These errors can arise when you use blocks that support constant scaling for best precision and you use the “best precision” scaling option. To avoid these errors, you should use the `Use Specified Scaling` parameter value. Refer to the *Fixed-Point Blockset User's Guide* for blocks that support this feature.

For more information about rapid simulation target or external mode, refer to the *Real-Time Workshop User's Guide*.

External Mode

There are two main issues to consider when using external mode.

External Mode Tuning Gives Different Results Than Simulation. When performing a simulation with fixed-point blocks, you can vary the block parameters over any range. The generated code has the flexibility to change the radix point on-the-fly. In external mode, however, the radix point cannot be changed on-the-fly and the code must be regenerated to accommodate radix point changes.

The Fixed-Point Blockset handles this issue by ignoring changes in the radix point when connected to the target in external mode. This means that unlike simulation, the blockset automatically produces the most accurate representation of the number without moving the radix point.

If a Block Parameter Change is Great Enough, the External Mode Cannot be Used. If a block parameter is changed by a sufficient amount (approximately a factor of two), the radix point changes. If you change a parameter such that the radix point moves during an external mode simulation (or during graphical editing) and you reconnect to the target, a checksum error occurs and the code must be rebuilt.

For example, suppose a fixed-point block has a parameter value of -2. You then build the code and connect in external mode. While connected, you change the parameter to -4. In simulation, this parameter change causes a radix point change. In external mode, the radix point is kept fixed. If you keep the parameter value of -4 and disconnect from the target, then when you reconnect, a checksum error occurs and you must rebuild your code.

Rapid Simulation Target

There is one main issue to consider when using rapid simulation target.

If a Block Parameter Change is Great Enough, the Rapid Simulation Target Cannot be Used. If a block parameter is changed by a sufficient amount (approximately a factor of two), the radix point changes. Any change in the radix point requires the code to be rebuilt (i.e., the model checksum is changed). This means that if the parameters are changed over a great enough range, the rapid simulation target cannot be used and a checksum error message occurs when you initialize the `rsim` executable.

MATLAB Web Server

Suppressing the HTTP Header with `htmlrep`

Use this form of the `htmlrep` function to suppress the output of the HTTP header 'Content-type: text/html\n\n' to outfile and outstring.

```
outstring = htmlrep(instruct,infile,outfile,'noheader')
```

This `htmlrep` option is documented online, in PDF and HTML form, but is not documented in the printed version of the *MATLAB Web Server User's Guide*.

Excel Link 1.0.8

Excel 7 and Excel 97 Versions

Release 11 includes two versions of Excel Link, which are placed in the MATLAB/exlink directory.

- Files that include a 95 at the end of their names (before the suffix), such as `excllink95.xla` and `ExliSamp95.xls`, correspond to Excel 7.
- Files that have no number in them, such as `excllink.xla` and `ExliSamp.xls`, correspond to Excel 97

Excel Link will work only with the appropriate version of Excel. You cannot interchange Excel Link files between versions.