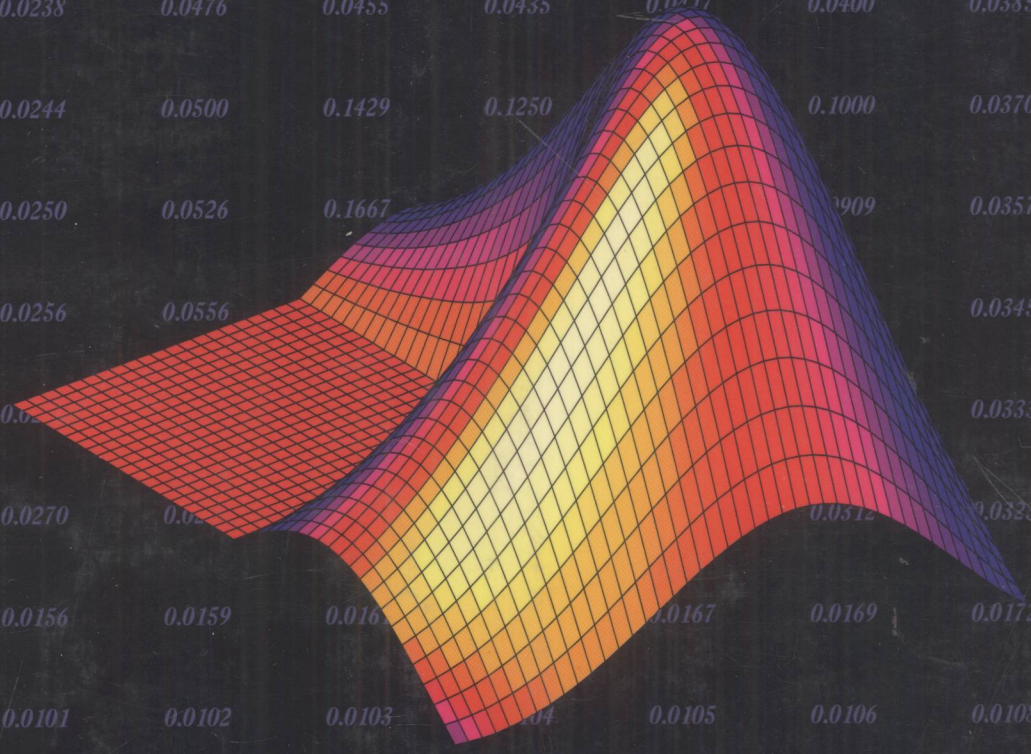
MATLAB®

High-Performance Numeric Computation and Visualization Software

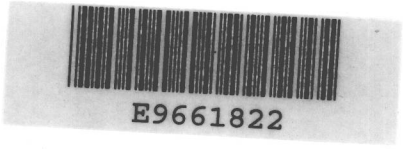


Reference Guide

The MATH WORKS Inc.

MATLAB®

High-Performance Numeric Computation and Visualization Software



Reference Guide

The MATH WORKS Inc.

MATLAB Reference Guide (October 1992)

Printing History

October 1992 First Printing

Reprints - July 1993, September 1994

The MathWorks, Inc.
24 Prime Park Way, Natick, Mass. 01760
Phone: (508) 653-1415 FAX: (508) 653-2997

Mosaic Home Page: `http://www.mathworks.com`

Anonymous FTP server: `ftp.mathworks.com`

Email addresses:

`support@mathworks.com`	Technical support.
`suggest@mathworks.com`	Product enhancement suggestions.
`bugs@mathworks.com`	Bug reports.
`doc@mathworks.com`	Documentation error reports.
`subscribe@mathworks.com`	Subscribing user registration.
`service@mathworks.com`	Order status, license renewals, passcodes.
`info@mathworks.com`	Sales, pricing, and general information.

What Is MATLAB?

MATLAB is a technical computing environment for high-performance numeric computation and visualization. MATLAB integrates numerical analysis, matrix computation, signal processing, and graphics in an easy-to-use environment where problems and solutions are expressed just as they are written mathematically – without traditional programming.

The name MATLAB stands for *matrix laboratory*. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects, which together represent the state of the art in software for matrix computation.

MATLAB is an interactive system whose basic data element is a matrix that does not require dimensioning. This allows you to solve many numerical problems in a fraction of the time it would take to write a program in a language such as Fortran, Basic, or C.

MATLAB has evolved over a period of years with input from many users. In university environments, it has become the standard instructional tool for introductory courses in applied linear algebra, as well as advanced courses in other areas. In industrial settings, MATLAB is used for research and to solve practical engineering and mathematical problems. Typical uses include general purpose numeric computation, algorithm prototyping, and special purpose problem solving with matrix formulations that arise in disciplines such as automatic control theory, statistics, and digital signal processing (time-series analysis).

MATLAB also features a family of application-specific solutions that we call *toolboxes*. Very important to most users of MATLAB, toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment in order to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems design, dynamic systems simulation, systems identification, neural networks, and others.

Probably the most important feature of MATLAB, and one that we took care to perfect, is its easy extensibility. This allows you to become a contributing author too, creating your own applications. In the years that MATLAB has been available, we have enjoyed watching many scientists, mathematicians, and engineers develop new and interesting applications, all without writing a single line of Fortran or other low-level code.

Who Wrote MATLAB?

The original MATLAB was written in Fortran by Cleve Moler, in an evolutionary process over several years. The underlying matrix algorithms are from the many people who worked on the LINPACK and EISPACK projects.

The current MATLAB program was written in C by The MathWorks. The first release was written by Steve Bangert, who wrote the parser/interpreter, Steve Kleiman, who implemented the graphics, and John Little and Cleve Moler, who wrote the analytical routines, the user's guide, and most of the M-files. Since the first release, many other people have joined the MATLAB development team and have made substantial contributions.

MATLAB Documentation

The MATLAB documentation set includes online help and printed manuals.

- *MATLAB User's Guide* contains platform-specific aspects of using MATLAB and provides a tutorial that introduces basic MATLAB functionality.

  Chapter 1 provides instructions for configuring and invoking MATLAB on your particular computer. It also explains system-dependencies, such as how to edit the previous line, print, use the mouse, and interact with the windowing system.

  Chapter 2 is an introduction to MATLAB. It describes and gives examples of MATLAB's basic features, including matrix manipulation, graphics, language features, and M-files.

- *MATLAB Reference Guide* is an alphabetical compendium of all MATLAB commands.

- *External Interface Guide* describes the external interfaces to MATLAB, including importing and exporting data, C and Fortran libraries for dynamic linking, reading and writing data files, and calling MATLAB as a computational engine.

- *Installation Guide* describes how to install MATLAB on your computer.

- Online help and demos provide online reference information about MATLAB commands and demonstrate some of MATLAB's features.

How to Use the Documentation Set

If you need to install MATLAB, you should read the *Installation Guide*. Once you install MATLAB, you can decide which document you prefer to use to learn the MATLAB commands.

If you are a new MATLAB user, you should start by reading the *Tutorial*. The most important things to learn are how to enter matrices, how to use the : (colon) operator, and how to invoke functions. After you master the basics, you can proceed through the rest of the *Tutorial*, or you can use the *Reference Guide*, the online help facility, or the demos to learn other commands.

If you used earlier versions of MATLAB, you should read the Graphics, Sparse Matrices, Debugging, and File I/O sections of the *Tutorial* to learn about these new features. You should also skim the rest of the *Tutorial* for other changes and additions. You may prefer to run some demos or the online Help facility or look through the *Reference Guide* to discover new features.

If you are familiar with MATLAB on other computers and need to learn about a different computer, you should read Chapter 1 for system-dependent features.

Formats and Conventions

This manual uses the following format in the *Reference* section:

Purpose	Provides short concise descriptions.
Synopsis	Shows format of the command or function.
Description	Describes what the command/function does and any rules/restrictions that apply.
Examples	Provides examples of how the command/function can be used.
Algorithm	Associated algorithms and routines.
See Also	Refers you to other related commands/functions.
References	Additional resources.

and the following conventions:

`Monospace`	Commands, function names, and screen displays; for example, `conv`.
Italics	Book titles, names of sections in this book, MATLAB toolbox names, mathematical notation,

and for introduction of new terms; for example, *Tutorial*.

Bold Initial Caps Key names, menu names, and items that are selected from menus; for example, the **Enter** key.

About the Cover

The cover of this guide depicts a solution to a problem which has played a small, but interesting, role in the history of numerical methods during the last 30 years. The problem involves finding the modes of vibration of a membrane supported by an L-shaped domain consisting of three unit squares. The nonconvex corner in the domain generates singularities in the solutions, thereby providing challenges for both the underlying mathematical theory and the computational algorithms. There are important applications, including wave guides, structures, and semiconductors.

Two of the founders of modern numerical analysis, George Forsythe and J. H. Wilkinson, worked on the problem in the 1950s. (See G. E. Forsythe and W. R. Wasow, Finite-Difference Methods for Partial Differential Equations, Wiley, 1960.) One of the authors of this guide (Moler) used finite difference techniques to compute solutions in 1965. Typical computer runs took up to half an hour of dedicated computer time on what were then Stanford University's primary computers, an IBM 7090 and a Burroughs B5000.

The first version of the approach we now use was published in 1967 by L. Fox, P. Henrici, and C. Moler (SIAM J. Numer. Anal. 4, 1967, pp. 89-102.) It replaced finite differences by combinations of distinguished fundamental solutions to the underlying differential equation formed from Bessel and trigonometric functions. The idea is a generalization of the fact that the real and imaginary parts of complex analytic functions are solutions to Laplace's equation. In the early 1970s, new matrix algorithms, particularly Gene Golub's orthogonalization techniques for least squares problems, provided further algorithmic improvements.

Today, MATLAB allows us to express the entire algorithm in a few dozen lines, to compute the solution with great accuracy in a few minutes on a computer at home, and to readily manipulate color three-dimensional displays of the results. We have included our MATLAB program, `membrane.m`, with the M-files supplied along with MATLAB.

Table of Contents

Quick Reference Tables

Commands and Functions

Quick Reference Tables

MATLAB provides 20 main categories of functions. Some of MATLAB's functions are built into the interpreter, while others take the form of M-files. The M-file functions, and in the case of the built-in functions, M-files containing only help text, are organized into 20 directories, each containing the files associated with a category. The MATLAB command `help` displays an online table of these main categories.

MATLAB's Main Categories of Functions	
`color`	Color control and lighting model functions.
`datafun`	Data analysis and Fourier transform functions.
`demos`	Demonstrations and samples.
`elfun`	Elementary math functions.
`elmat`	Elementary matrices and matrix manipulation.
`funfun`	Function functions – nonlinear numerical methods.
`general`	General purpose commands.
`graphics`	General purpose graphics functions.
`iofun`	Low-level file I/O functions.
`lang`	Language constructs and debugging.
`matfun`	Matrix functions – numerical linear algebra.
`ops`	Operators and special characters.
`plotxy`	Two dimensional graphics.
`plotxyz`	Three dimensional graphics.
`polyfun`	Polynomial and interpolation functions.
`sparfun`	Sparse matrix functions.
`specfun`	Specialized math functions.
`specmat`	Specialized matrices.
`sounds`	Sound processing functions.
`strfun`	Character string functions.

The following pages contain tables of functions within each of these specific areas. If you execute `help` on one of the directory names listed on the left side of this table, MATLAB displays an online version of the tables within that area.