

# A P P E N D I X

## Guide to Selected MATLAB Commands and Functions

This appendix is a guide to those MATLAB commands and functions that are particularly useful for the system dynamics methods covered in this text. For more information, in the Command window type `help topic`, where `topic` is the name of the command or function. ■

### Operators and special characters

Item	Description
+	Plus. Addition operator.
-	Minus. Subtraction operator.
*	Scalar and matrix multiplication operator.
.*	Array multiplication operator.
^	Scalar and matrix exponentiation operator.
.^	Array exponentiation operator.
\	Left division operator.
/	Right division operator.
.\	Array left division operator.
./	Array right division operator.
:	Colon. Generates regularly spaced elements and represents an entire row or column.
()	Parentheses. Encloses function arguments and array indices; overrides precedence.
[ ]	Brackets. Encloses array elements.
{ }	Braces. Encloses cell elements.
.	Decimal point.
...	Ellipsis. Line continuation operator.
,	Comma. Separates statements and elements in a row of an array.
;	Semicolon. Separates columns in an array and suppresses line feeds.
%	Percent sign. Designates a comment and specifies formatting.
'	Quote sign and transpose operator.
.'	Nonconjugated transpose operator.
=	Assignment (replacement) operator.

## Logical and relational operators.

Item	Description
==	Relational operator: equal to.
~=	Relational operator: not equal to.
<	Relational operator: less than.
<=	Relational operator: less than or equal to.
>	Relational operator: greater than.
>=	Relational operator: greater than or equal to.
&	Logical operator: AND
	Logical operator: OR
~	Logical operator: NOT

## Special variables and constants.

Item	Description
ans	Most recent answer.
eps	Accuracy of floating point precision.
i,j	The imaginary unit $\sqrt{-1}$ .
Inf	Infinity.
NaN	Undefined numerical result (Not a Number).
pi	The number $\pi$ .

## Commands for managing a session.

Item	Description
clc	Clears Command window.
clear	Removes variables from memory.
doc	Displays documentation.
exist	Checks for existence of file or variable.
global	Declares variables to be global.
help	Searches for a help topic.
helpwin	Displays help text in the Help Browser.
lookfor	Searches help entries for a keyword.
quit	Stops MATLAB.
who	Lists current variables.
whos	Lists detailed information about current variables.

## System and file commands.

Item	Description
cd	Changes current directory.
dir	Lists all files in current directory.
load	Loads workspace variables from a file.
path	Displays search path.
pwd	Displays current directory.
save	Saves workspace variables in a file.
type	Displays contents of a file.
what	Lists all MATLAB files.

## Input/output commands.

Item	Description
disp	Displays contents of an array or string.
format	Controls screen display format.
input	Displays prompts and waits for input.
menu	Displays a menu of choices.
;	Suppresses screen printing.

## Some numeric display formats.

Item	Description
format long	Sixteen decimal digits.
format long e	Scientific notation with sixteen digits plus exponent.
format short	Four decimal digits (the default).
format short e	Scientific notation with five digits plus exponent.

## Array functions.

Item	Description
det	Computes determinant of an array.
eig	Computes the eigenvalues of a matrix.
eye	Creates the identity matrix.
find	Finds indices of nonzero elements.
length	Computes number of elements in an array.
linspace	Creates a regularly spaced array.
logspace	Creates a logarithmically spaced array.
max	Returns largest element in an array.
min	Returns smallest element in an array.
ones	Creates an array of ones.
size	Computes array size.
sort	Sorts each array column.
sum	Sums each array column.
zeros	Creates an array of zeros.

## Exponential and logarithmic functions.

Item	Description
exp(x)	Exponential, $e^x$ .
log(x)	Natural logarithm, $\ln x$ .
log10(x)	Common (base 10) logarithm, $\log x = \log_{10} x$ .
sqrt(x)	Square root, $\sqrt{x}$ .

## Complex functions.

Item	Description
abs(x)	Absolute value, $ x $ .
angle(x)	Angle of a complex number $x$ .
conj(x)	Complex conjugate of $x$ .
imag(x)	Imaginary part of a complex number $x$ .
real(x)	Real part of a complex number $x$ .

## Numeric functions.

Item	Description
ceil	Rounds to the nearest integer toward $\infty$ .
fix	Rounds to the nearest integer toward zero.
floor	Rounds to the nearest integer toward $-\infty$ .
round	Rounds toward the nearest integer.
sign	Signum function.

## Trigonometric functions.

Item	Description
acos(x)	Inverse cosine, $\arccos x = \cos^{-1} x$ .
acot(x)	Inverse cotangent, $\operatorname{arccot} x = \cot^{-1} x$ .
acsc(x)	Inverse cosecant, $\operatorname{arccsc} x = \csc^{-1} x$ .
asec(x)	Inverse secant, $\operatorname{arcsec} x = \sec^{-1} x$ .
asin(x)	Inverse sine, $\arcsin x = \sin^{-1} x$ .
atan(x)	Inverse tangent, $\operatorname{arctan} x = \tan^{-1} x$ .
atan2(y, x)	Four quadrant inverse tangent.
cos(x)	Cosine, $\cos x$ .
cot(x)	Cotangent, $\cot x$ .
csc(x)	Cosecant, $\csc x$ .
sec(x)	Secant, $\sec x$ .
sin(x)	Sine, $\sin x$ .
tan(x)	Tangent, $\tan x$ .

## Hyperbolic functions.

Item	Description
acosh(x)	Inverse hyperbolic cosine, $\cosh^{-1} x$ .
acoth(x)	Inverse hyperbolic cotangent, $\coth^{-1} x$ .
acsch(x)	Inverse hyperbolic cosecant, $\operatorname{csch}^{-1} x$ .
asech(x)	Inverse hyperbolic secant, $\operatorname{sech}^{-1} x$ .
asinh(x)	Inverse hyperbolic sine, $\sinh^{-1} x$ .
atanh(x)	Inverse hyperbolic tangent, $\tanh^{-1} x$ .
cosh(x)	Hyperbolic cosine, $\cosh x$ .
coth(x)	Hyperbolic cotangent, $\cosh x / \sinh x$ .
csch(x)	Hyperbolic cosecant, $1 / \sinh x$ .
sech(x)	Hyperbolic secant, $1 / \cosh x$ .
sinh(x)	Hyperbolic sine, $\sinh x$ .
tanh(x)	Hyperbolic tangent, $\sinh x / \cosh x$ .

## Polynomial functions.

Item	Description
conv	Computes product of two polynomials.
deconv	Computes ratio of polynomials.
poly(r)	Computes coefficients of polynomial whose roots are given in the vector $r$ .
poly(A)	Computes coefficients of the characteristic polynomial corresponding to the matrix $A$ .
polyfit	Fits a polynomial to data.
polyval	Evaluates a polynomial at specified values of its independent variable.
residue	Computes residues, poles, and direct term of a partial fraction expansion.
roots	Computes polynomial roots.

## Logical functions.

Item	Description
any	True if any elements are nonzero.
all	True if all elements are nonzero.
find	Finds indices of nonzero elements.
finite	True if elements are finite.
isnan	True if elements are undefined.
isinf	True if elements are infinite.
isempty	True if array is empty.
isreal	True if all elements are real.
logical	Converts numeric values to logical values.
xor	Exclusive or.

## Miscellaneous mathematical functions.

Item	Description
cross	Cross product.
dot	Dot product.
fminbnd	Finds minimum of single-variable function.
fminsearch	Finds minimum of multivariable function.
function	Creates a user-defined function.
fzero	Finds zero of single-variable function.
mean	Calculates the mean value.
std	Calculates the standard deviation.
trapz	Numerical integration with the trapezoidal rule.

## Two-dimensional plotting commands.

Item	Description
axes	Creates axes objects.
axis	Sets axis limits.
fplot	Intelligent plotting of functions.
ginput	Reads coordinates of cursor position.
grid	Displays gridlines.
gtext	Enables label placement with mouse.
hold off	Releases a prior <code>hold on</code> command.
hold on	Holds current graph to enable subsequent plotting.
legend	Enables legend placement with mouse.
loglog	Creates log-log plot.
plot	Generates $xy$ plot.
polar	Creates polar plot.
print	Prints plot or saves plot to a file.
refresh	Redraws current figure window.
semilogx	Creates semilog plot (logarithmic abscissa).
semilogy	Creates semilog plot (logarithmic ordinate).
set	Specifies properties of objects, such as axes.
subplot	Creates plots in subwindows.
text	Places a string in a figure.
title	Puts text at top of plot.
xlabel	Adds text label to abscissa (the $x$ axis).
ylabel	Adds text label to ordinate (the $y$ axis).

## Program flow control.

Item	Description
<code>break</code>	Terminates execution of a loop.
<code>case</code>	Provides alternate execution paths within <code>switch</code> structure.
<code>continue</code>	Passes control to the next iteration of a <code>for</code> or <code>while</code> loop.
<code>else</code>	Delineates alternate block of statements.
<code>elseif</code>	Conditionally executes statements.
<code>end</code>	Terminates <code>for</code> , <code>while</code> , and <code>if</code> statements.
<code>for</code>	Repeat statements a specific number of times.
<code>if</code>	Execute statements conditionally.
<code>otherwise</code>	Provides optional control within a <code>switch</code> structure.
<code>pause</code>	Causes the program to stop and wait for a key press before continuing.
<code>switch</code>	Directs program execution by comparing input with <code>case</code> expressions.
<code>while</code>	Repeats statements an indefinite number of times.

## LTI model functions.

Item	Description
<code>damp</code>	Computes the characteristic roots, damping ratio, and damped oscillation frequency of complex roots.
<code>ltimodels</code>	Gives help about LTI models.
<code>ltiprops</code>	Gives help about LTI model properties.
<code>ltiview</code>	Interface for analyzing time and frequency response.
<code>ord2</code>	Creates a state-space or transfer function representation of a second-order system from its natural frequency and damping ratio.
<code>ss</code>	Creates an LTI model in state-space form.
<code>ss2tf</code>	Converts from state-space to transfer function form.
<code>ss2zp</code>	Converts from state-space to zero-pole form.
<code>ssdata</code>	Extracts state-space matrices from an LTI model.
<code>tf</code>	Creates an LTI model in transfer function form.
<code>tf2ss</code>	Converts from transfer function form to state-space form.
<code>tf2zp</code>	Converts from transfer function form to zero-pole form.
<code>tfdata</code>	Extracts equation coefficients from an LTI model.
<code>zp2tf</code>	Converts from zero-pole form to transfer function form.
<code>zpk</code>	Creates an LTI model from its poles, zeros, and gain.
<code>zpkdata</code>	Returns the poles, zeros, and gain of an LTI model.

## Equation solvers.

Command	Description
<code>impulse</code>	Computes and plots the impulse response of the LTI model <code>sys</code> .
<code>initial</code>	Computes and plots the free response of an LTI model given in state-model form.
<code>lsim</code>	Computes and plots the response of an LTI object to a defined input function of time.
<code>ode23</code>	Solves linear and nonlinear differential equations.
<code>ode45</code>	Solves linear and nonlinear differential equations.
<code>step</code>	Computes and plots the step response of an LTI object.

## Predefined input functions.

Command	Description
gensig	Generates a periodic sine, square, or pulse input having a specified period.
stepfun	Generates a step function input.

## Frequency response functions.

Command	Description
bode	Computes the magnitude ratio and phase angle of an LTI model and displays the Bode plots.
bodemag	Computes the magnitude ratio of an LTI model and displays the magnitude plot.
evalfr	Evaluates a transfer function model at a specified value of $s$ .
freqresp	Computes the frequency response of an LTI model at multiple specified frequencies.
margin	Computes phase and gain margins of an LTI model and displays the Bode plots.

## Root locus functions.

Command	Description
pole	Computes the poles of an LTI model.
pzmap	Computes the poles and zeros of an LTI model.
rlocfind	Enables use of the cursor to select the gain from a specified point on a root locus plot.
rlocus	Computes and displays the root locus plot.
rltool	Starts the root locus GUI interface.
sgrid	Superimposes a grid of constant $\zeta$ and constant $\omega_n$ lines on the root locus plot.
zero	Computes the zeros of an LTI model.

## Control system functions.

Command	Description
acker	Uses Ackermann's method to compute the feedback gain matrix for a single-input system to place the closed-loop poles at specified locations.
c2d	Converts a continuous-time model into a discrete-time model using a zero-order hold on the inputs, with a specified sampling time.
feedback	Creates an LTI model from two subsystems connected with a feedback loop.
ltiview	Starts the LTI viewer.
pade	Pade approximation to the transfer function of the dead time element.
parallel	Creates an LTI model from two subsystems connected in parallel.
series	Creates an LTI model from two subsystems connected in series.
sisotool	Graphical user interface for designing single-input/single-output compensators.