MathWorks®
Accelerating the pace of engineering and science

Symbolic Math Toolbox: Quick Reference Sheet

Algebra

subs	Symbolic substitution:
	`syms a b;`
	`subs(a^3+b,{a,b},{2,sym('e')})`

double	Convert symbolic values to double precision:
	`symN = sym(pi);`
	`doubleN = double(symN)`

solve	Equations and systems solver:
	`syms a b u v;`
	`S=solve(u+v==a, u-v==b)`

simplify	Algebraic simplification:
	`syms x;`
	`simplify(sin(x)^2 + cos(x)^2)`

isolate	Isolate variable or expression in equation: `syms a b c x; isolate(a*x^2+b*x+c==0,x)`
lhs	Left side (LHS) of equation: `syms x y; lhs(x^2 >= y^2)`
rhs	Right side (RHS) of equation: `syms x y; rhs(x^2 >= y^2)`
rewrite	Rewrite expression in terms of another function: `syms x; rewrite(tan(x)/cos(x),'sin')`

Graphics

fplot	Plot symbolic expression or function: `syms x; f(x) = sin(x)/x; fplot(f)`
fplot3	Plot 3-D parametric curve: `syms x; fplot3(sin(x),cos(x),log(x))`

fsurf	Plot 3-D surface, mesh or contour:
	`syms x y;`
	`f(x,y)=x*exp(-x^2-y^2); fsurf(f)`

fmesh	Plot 3-D mesh:
	`syms x y;`
	`f(x,y)=x*exp(-x^2-y^2); fmesh(f)`

fcontour	Plot contours: `f(x,y)=x*exp(-x^2-y^2); fcontour(f)`

fimplicit, fimplicit3	Plot implicit symbolic equation or function:
	`syms x y;`
	`fimplicit(y^2-x^2*(x+1),[-2 2])`
	`syms x y z;`
	`fimplicit3(x^2*y*z+y^3-z^3)`

Calculus

limit	Compute limit of symbolic expression: `limit(1/x,x,0,'left')`
diff	Differentiation: `syms x t; diff(sin(x^2+t),x)`
int	Definite and indefinite integrals: `syms x z; int(x/(1 + z^2), z)`
symsum	Sum of a series: `syms k n; symsum(k,0,n-1)`
series	Puiseux series expansion: `syms x; series(1/sin(x), x)`
taylor	Taylor series: `syms x; taylor(exp(-x))`

gradient	Gradient vector of scalar function:
	`syms x y z;`
	`gradient(x*y + 2*z*x, [x, y, z])`

hessian	Hessian matrix of scalar function:
	`syms x y z;`
	`hessian(x*y + 2*z*x, [x, y, z])`

laplacian	Laplacian of scalar function:
	`syms x y z;`
	`laplacian(1/x + y^2 + z^3, [x y z])`

jacobian	Jacobian matrix:
	`syms x y z u v;`
	`jacobian([x*y*z; y; x+z],[x y z])`

divergence	Divergence of vector field:
	`syms x y z;`
	`divergence([x^2 2*y z], [x y z])`

Functions

functions	Symbolic Functions:
	`syms x y; f(x,y) = x + 2*y;`
	`f(1,2)`

piecewise	Conditionally defined expression or function:
	`syms x;`
	`g(x)=piecewise(x<0,-1,x>=0,2);`
	`g(3)`

matlabFunction, matlabFunctionBlock, simscapeEquation	Convert symbolic expression to a MATLAB callable function, Simulink block or Simscape equation. `g = matlabFunction(f)`

MathWorks
Accelerating the pace of engineering and science

MATLAB: Quick Reference Sheet

Live Editor

Command	Keyboard Shortcut
Move Up One Section	Ctrl+upArrow
Move Down One Section	Ctrl+downArrow
Run Section	Ctrl+Enter
Run Section and Advance	Ctrl+Shift+Enter
Run All Sections	F5
Insert a new line or Code or Text	Alt+Enter
Toggle between Code and Text	Ctrl+E
Title	Ctrl+Alt+L
Heading	Ctrl+Alt+H
Section Break	Ctrl+Alt+Enter
Bulleted List	Ctrl+Alt+U
Numbered List	Ctrl+Alt+O
Italic	Ctrl+I
Bold	Ctrl+B
Monospace	Ctrl+M
Underline	Ctrl+U
LaTeX Equation	Ctrl+Alt+G
Hyperlink	Ctrl+K

{ ... } Cells & Tables

cell	Create a cell array: `cell(3,5)`
curly brackets, {}	Access specific cell in cell array: `c = cell(3,5); c{3,5}`
cell2mat	Transform cell to matrix: `c = cell(3,5); cell2mat(c)`
cellfun	Apply function to each cell: `c = cell(3,5); cellfun(@mean,c)`
table	Create a table from given variables: `table([1:2]',{'one';'two'})`
cell2table	Transform a cell array to a table: `c=cell(3,5); cell2table(c)`
table2array	Transform a table into an array: `t=table([1:3]'); table2array(t)`

Matrix Operations

colon, :	Create a row vector: `1:10; 1:2:10`
linspace	Linearly spaced points between endpoints: `Linspace(1,3,1000)`
square brackets, []	Matrix construction and concatenation: `[1,2;3,4]`
NaN	Create matrix of NaN values: `NaN(3,5)`
ones	Create matrix of 1 values: `ones(3,5)`
zeros	Create matrix of 0 values: `zeros(3,5)`
parentheses, ()	Index into matrix: `x=[1,2;3,4]; x(2,1)`
diag	Diagonal elements of matrix: `x=[1,2;3,4]; diag(x)`
times, .*	Element by element multiplication: `x=[1,2;3,4]; y=[2,3;4,5]; x.*y`
rdivide, ./	Element by element division: `x=[1,2;3,4]; y=[2,3;4,5]; x./y`
plus, +	Element by element addition: `x=[1,2;3,4]; y=[2,3;4,5]; x+y`
minus, -	Element by element subtraction: `x=[1,2;3,4]; y=[2,3;4,5]; x-y`
gt, >	Determine greater than: `x=[1,3;4,2]; x>2`
inv	Invert a matrix: `X=[1,2;3,4]; inv(x)`
size	Size of a matrix: `x=[1,2,3;4,5,6]; size(x)`
eye	Create identity matrix: `eye(3,4)`
isnan	Find elements that are NaN: `x=[1,NaN;NaN,4]; isnan(x)`
repmat	Repeat copies of a matrix: `x=diag([100 200 300]); repmat(x,2)`
bsxfun	Element wise operation on two matrices: `x=[1,2;3,4]; bsxfun(@minus,x,mean(x))`
arrayfun	Apply function to each element: `x=[1;3]; arrayfun(@(x) plus(x,x), x)`

Full MATLAB cheat sheet