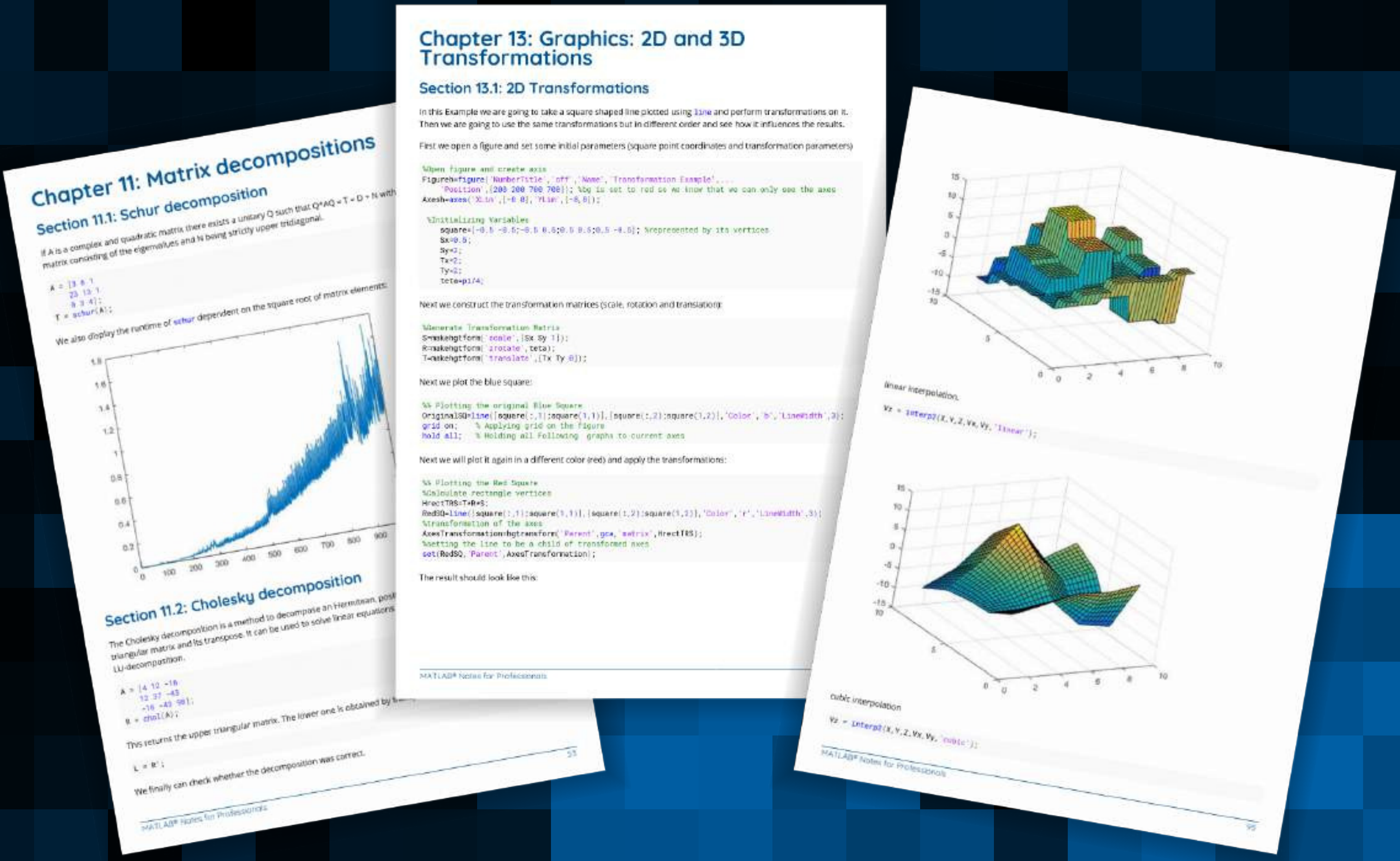
# MATLAB®

## Notes for Professionals



## Chapter 11: Matrix decompositions

### Section 11.1: Schur decomposition

If A is a complex and quadratic matrix there exists a unitary Q such that Q*AQ = T = D + N with
matrix consisting of the eigenvalues and N being strictly upper tridiagonal.

```
A = [3 6 1
     23 13 1
     8 3 4];
T = schur(A);
```

We also display the runtime of schur dependent on the square root of matrix elements:

### Section 11.2: Cholesky decomposition

The Cholesky decomposition is a method to decompose an Hermitean, posi
triangular matrix and its transpose. It can be used to solve linear equations
LU-decomposition.

```
A = [4 12 -16
     12 37 -43
     -16 -43 98];
R = chol(A);
```

This returns the upper triangular matrix. The lower one is obtained by tr

```
L = R';
```

We finally can check whether the decomposition was correct.

## Chapter 13: Graphics: 2D and 3D Transformations

### Section 13.1: 2D Transformations

In this Example we are going to take a square shaped line plotted using line and perform transformations on it. Then we are going to use the same transformations but in different order and see how it influences the results.

First we open a figure and set some initial parameters (square point coordinates and transformation parameters)

```
%Open figure and create axis
Figureh=figure('NumberTitle','off','Name','Transformation Example',...
    'Position',[200 200 700 700]); %bg is set to red so we know that we can only see the axes
Axesh=axes('XLim',[-8 8],'YLim',[-8,8]);

 %Initializing Variables
    square=[-0.5 -0.5;-0.5 0.5;0.5 0.5;0.5 -0.5]; %represented by its vertices
    Sx=0.5;
    Sy=2;
    Tx=2;
    Ty=2;
    teta=pi/4;
```

Next we construct the transformation matrices (scale, rotation and translation):

```
%Generate Transformation Matrix
S=makehgtform('scale',[Sx Sy 1]);
R=makehgtform('zrotate',teta);
T=makehgtform('translate',[Tx Ty 0]);
```

Next we plot the blue square:

```
%% Plotting the original Blue Square
OriginalSQ=line([square(:,1);square(1,1)],[square(:,2);square(1,2)],'Color','b','LineWidth',3);
grid on;    % Applying grid on the figure
hold all;   % Holding all Following  graphs to current axes
```

Next we will plot it again in a different color (red) and apply the transformations:

```
%% Plotting the Red Square
%Calculate rectangle vertices
HrectTRS=T*R*S;
RedSQ=line([square(:,1);square(1,1)],[square(:,2);square(1,2)],'Color','r','LineWidth',3);
%transformation of the axes
AxesTransformation=hgtransform('Parent',gca,'matrix',HrectTRS);
%seting the line to be a child of transformed axes
set(RedSQ,'Parent',AxesTransformation);
```

The result should look like this:

linear interpolation.

```
Vz = interp2(X,Y,Z,Vx,Vy,'linear');
```

cubic interpolation

```
Vz = interp2(X,Y,Z,Vx,Vy,'cubic');
```

**100+ pages**

of professional hints and tricks

**GoalKicker.com**

Free Programming Books

**Disclaimer**

This is an unofficial free book created for educational purposes and is not affiliated with official MATLAB® group(s) or company(s). All trademarks and registered trademarks are the property of their respective owners

# Contents

About ... 1

Chapter 1: Getting started with MATLAB Language ... 2

- Section 1.1: Indexing matrices and arrays ... 3
- Section 1.2: Anonymous functions and function handles ... 8
- Section 1.3: Matrices and Arrays ... 11
- Section 1.4: Cell arrays ... 13
- Section 1.5: Hello World ... 14
- Section 1.6: Scripts and Functions ... 14
- Section 1.7: Helping yourself ... 16
- Section 1.8: Data Types ... 16
- Section 1.9: Reading Input & Writing Output ... 19

Chapter 2: Initializing Matrices or arrays ... 21

- Section 2.1: Creating a matrix of 0s ... 21
- Section 2.2: Creating a matrix of 1s ... 21
- Section 2.3: Creating an identity matrix ... 21

Chapter 3: Conditions ... 22

- Section 3.1: IF condition ... 22
- Section 3.2: IF-ELSE condition ... 22
- Section 3.3: IF-ELSEIF condition ... 23
- Section 3.4: Nested conditions ... 24

Chapter 4: Functions ... 27

- Section 4.1: nargin, nargout ... 27

Chapter 5: Set operations ... 29

- Section 5.1: Elementary set operations ... 29

Chapter 6: Documenting functions ... 30

- Section 6.1: Obtaining a function signature ... 30
- Section 6.2: Simple Function Documentation ... 30
- Section 6.3: Local Function Documentation ... 30
- Section 6.4: Documenting a Function with an Example Script ... 31

Chapter 7: Using functions with logical output ... 34

- Section 7.1: All and Any with empty arrays ... 34

Chapter 8: For loops ... 35

- Section 8.1: Iterate over columns of matrix ... 35
- Section 8.2: Notice: Weird same counter nested loops ... 35
- Section 8.3: Iterate over elements of vector ... 36
- Section 8.4: Nested Loops ... 37
- Section 8.5: Loop 1 to n ... 38
- Section 8.6: Loop over indexes ... 39

Chapter 9: Object-Oriented Programming ... 40

- Section 9.1: Value vs Handle classes ... 40
- Section 9.2: Constructors ... 40
- Section 9.3: Defining a class ... 42
- Section 9.4: Inheriting from classes and abstract classes ... 43

Chapter 10: Vectorization ... 47

- Section 10.1: Use of bsxfun ... 47
- Section 10.2: Implicit array expansion (broadcasting) [R2016b] ... 48

Section 10.3: Element-wise operations ... 49
Section 10.4: Logical Masking ... 50
Section 10.5: Sum, mean, prod & co ... 51
Section 10.6: Get the value of a function of two or more arguments ... 52
Chapter 11: Matrix decompositions ... 53
Section 11.1: Schur decomposition ... 53
Section 11.2: Cholesky decomposition ... 53
Section 11.3: QR decomposition ... 54
Section 11.4: LU decomposition ... 54
Section 11.5: Singular value decomposition ... 55
Chapter 12: Graphics: 2D Line Plots ... 56
Section 12.1: Split line with NaNs ... 56
Section 12.2: Multiple lines in a single plot ... 56
Section 12.3: Custom colour and line style orders ... 57
Chapter 13: Graphics: 2D and 3D Transformations ... 61
Section 13.1: 2D Transformations ... 61
Chapter 14: Controlling Subplot coloring in MATLAB ... 64
Section 14.1: How it's done ... 64
Chapter 15: Image processing ... 65
Section 15.1: Basic image I/O ... 65
Section 15.2: Retrieve Images from the Internet ... 65
Section 15.3: Filtering Using a 2D FFT ... 65
Section 15.4: Image Filtering ... 66
Section 15.5: Measuring Properties of Connected Regions ... 67
Chapter 16: Drawing ... 70
Section 16.1: Circles ... 70
Section 16.2: Arrows ... 71
Section 16.3: Ellipse ... 74
Section 16.4: Pseudo 4D plot ... 75
Section 16.5: Fast drawing ... 79
Section 16.6: Polygon(s) ... 80
Chapter 17: Financial Applications ... 82
Section 17.1: Random Walk ... 82
Section 17.2: Univariate Geometric Brownian Motion ... 82
Chapter 18: Fourier Transforms and Inverse Fourier Transforms ... 84
Section 18.1: Implement a simple Fourier Transform in MATLAB ... 84
Section 18.2: Images and multidimensional FTs ... 85
Section 18.3: Inverse Fourier Transforms ... 90
Chapter 19: Ordinary Differential Equations (ODE) Solvers ... 92
Section 19.1: Example for odeset ... 92
Chapter 20: Interpolation with MATLAB ... 94
Section 20.1: Piecewise interpolation 2 dimensional ... 94
Section 20.2: Piecewise interpolation 1 dimensional ... 96
Section 20.3: Polynomial interpolation ... 101
Chapter 21: Integration ... 105
Section 21.1: Integral, integral2, integral3 ... 105
Chapter 22: Reading large files ... 107
Section 22.1: textscan ... 107

Section 22.2: Date and time strings to numeric array fast ..... 107
Chapter 23: Usage of `accumarray()` Function ..... 109
Section 23.1: Apply Filter to Image Patches and Set Each Pixel as the Mean of the Result of Each Patch ..... 109
Section 23.2: Finding the maximum value among elements grouped by another vector ..... 110
Chapter 24: Introduction to MEX API ..... 111
Section 24.1: Check number of inputs/outputs in a C++ MEX-file ..... 111
Section 24.2: Input a string, modify it in C, and output it ..... 112
Section 24.3: Passing a struct by field names ..... 113
Section 24.4: Pass a 3D matrix from MATLAB to C ..... 113
Chapter 25: Debugging ..... 116
Section 25.1: Working with Breakpoints ..... 116
Section 25.2: Debugging Java code invoked by MATLAB ..... 118
Chapter 26: Performance and Benchmarking ..... 121
Section 26.1: Identifying performance bottlenecks using the Profiler ..... 121
Section 26.2: Comparing execution time of multiple functions ..... 124
Section 26.3: The importance of preallocation ..... 125
Section 26.4: It's ok to be `single`! ..... 127
Chapter 27: Multithreading ..... 130
Section 27.1: Using parfor to parallelize a loop ..... 130
Section 27.2: Executing commands in parallel using a "Single Program, Multiple Data" (SPMD) statement ..... 130
Section 27.3: Using the batch command to do various computations in parallel ..... 131
Section 27.4: When to use parfor ..... 131
Chapter 28: Using serial ports ..... 133
Section 28.1: Creating a serial port on Mac/Linux/Windows ..... 133
Section 28.2: Choosing your communication mode ..... 133
Section 28.3: Automatically processing data received from a serial port ..... 136
Section 28.4: Reading from the serial port ..... 137
Section 28.5: Closing a serial port even if lost, deleted or overwritten ..... 137
Section 28.6: Writing to the serial port ..... 137
Chapter 29: Undocumented Features ..... 138
Section 29.1: Color-coded 2D line plots with color data in third dimension ..... 138
Section 29.2: Semi-transparent markers in line and scatter plots ..... 138
Section 29.3: C++ compatible helper functions ..... 140
Section 29.4: Scatter plot jitter ..... 141
Section 29.5: Contour Plots - Customise the Text Labels ..... 141
Section 29.6: Appending / adding entries to an existing legend ..... 143
Chapter 30: MATLAB Best Practices ..... 145
Section 30.1: Indent code properly ..... 145
Section 30.2: Avoid loops ..... 146
Section 30.3: Keep lines short ..... 146
Section 30.4: Use assert ..... 147
Section 30.5: Block Comment Operator ..... 147
Section 30.6: Create Unique Name for Temporary File ..... 148
Chapter 31: MATLAB User Interfaces ..... 150
Section 31.1: Passing Data Around User Interface ..... 150
Section 31.2: Making a button in your UI that pauses callback execution ..... 152
Section 31.3: Passing data around using the "handles" structure ..... 153

Section 31.4: Performance Issues when Passing Data Around User Interface ..... 154

Chapter 32: Useful tricks ..... 157

Section 32.1: Extract figure data ..... 157

Section 32.2: Code Folding Preferences ..... 158

Section 32.3: Functional Programming using Anonymous Functions ..... 160

Section 32.4: Save multiple figures to the same .fig file ..... 160

Section 32.5: Comment blocks ..... 161

Section 32.6: Useful functions that operate on cells and arrays ..... 162

Chapter 33: Common mistakes and errors ..... 165

Section 33.1: The transpose operators ..... 165

Section 33.2: Do not name a variable with an existing function name ..... 165

Section 33.3: Be aware of floating point inaccuracy ..... 166

Section 33.4: What you see is NOT what you get: char vs cellstring in the command window ..... 167

Section 33.5: Undefined Function or Method X for Input Arguments of Type Y ..... 168

Section 33.6: The use of "i" or "j" as imaginary unit, loop indices or common variable ..... 169

Section 33.7: Not enough input arguments ..... 172

Section 33.8: Using `length` for multidimensional arrays ..... 173

Section 33.9: Watch out for array size changes ..... 173

Credits ..... 175

You may also like ..... 177

Click here to download full PDF material