


Department Informations- und Elektrotechnik	Labor für Grundlagen der Elektrotechnik	 Hochschule für Angewandte Wissenschaften Hamburg Hamburg University of Applied Sciences
Studiengruppe:	<h1>ETP2</h1>	Protokollführer (Name, Vorname):
Übungstag:		Weitere Übungsteilnehmer:
Professor:		Testat:
Matlab im PC-Labor Einführung in das grundlegende Arbeiten mit Matlab		

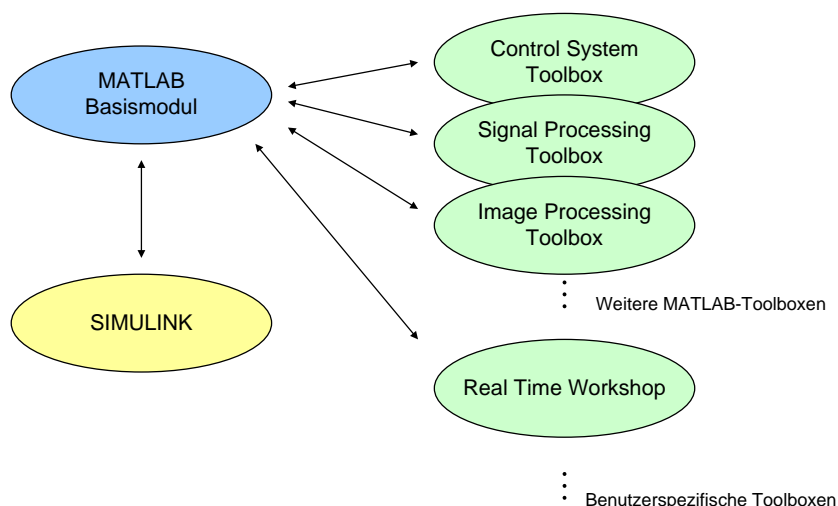
Einleitung

Matlab ist ein numerisches Berechnungs- und Simulationswerkzeug der Firma *The Math Works Inc.* aus den USA, das sich als Standardwerkzeug in der HAW durchgesetzt hat. In diesem Versuch haben Sie die Möglichkeit, sich mit der grundlegenden Bedienung vertraut zu machen.

Die Anfänge von Matlab entstanden zunächst an Universitäten und waren offen für alle. Im Laufe der Zeit ist Matlab jetzt ein kommerzielles Produkt geworden und hat sich zum Industriestandard für Simulationen entwickelt. Anders als andere Simulationsprogramme ist Matlab mathematisch orientiert und damit in seiner Grundform auf keine spezielle Anwendung festgelegt. Erst durch die Verwendung spezieller Funktionsbibliotheken (sogenannter Toolboxes) kann Matlab anwendungsspezifisch erweitert werden.

Der grundsätzliche Datentyp in Matlab ist die Matrix. Daraus abgeleitet ist der Name: MATrix LABoratory.

Eine Besonderheit stellt die Matlab-Erweiterung Simulink dar. Dieses Werkzeug stellt dem Benutzer eine komfortable grafische Benutzeroberfläche für Matlab zur Verfügung. Passend für die bewährte ingenieurmäßige Arbeitsweise ist Simulink blockorientiert aufgebaut. Innerhalb der Blockschaltbilder können Hierarchien gebildet werden, die es erlauben, auf jeder Arbeitsebene übersichtliche Strukturen zu erzeugen. Simulink ist jedoch nicht Gegenstand dieses Labors.



Ausgewählte MATLAB-Kommandos

Allgemein

help	- Hilfefunktion
doc	- HTML-Hilfefunktion
demo, intro	- Demo und Einführung in MATLAB
who(s), what, which	- Info zu Variablen, Dateien und Funktionen
clear (all)	- Löschen von Variablen etc.
size, length	- Größe bzw. Länge von Matrizen
tic, toc	- Laufzeit bestimmen
format (long)	- Wahl des Ausgabeformats
more (on/off)	- seitenweise Ausgabe
save, load	- Variablen abspeichern bzw. laden

Symbole

%	- Hilfe-Text, auskommentieren
[]	- Matrizendefinition und Ausgabeargumente
()	- Prioritäten und Eingabeargumente
=	- zuordnen
:	- Indizierung und Vektordefinition
;	- Ausgabe verhindern
'...'	- definieren eines Textstrings

Operatoren

+ -	- addieren und subtrahieren
* / ^	- matrixweise Operationen
.* ./ .^	- elementweise Operationen

Relationale Operatoren

> >=	- größer, größer oder gleich
< <=	- kleiner, kleiner oder gleich
== ~=	- gleich, ungleich
& ~	- und, oder, nicht

Vordefinierte Variable

pi	- π
inf	- unendlich (z.B. 1/0)
nan	- not a number (z.B. 0/0)
i, j	- imaginäre Einheit
eps	- Gleitkomma-Genauigkeit

Standard-Funktionen (elementweise)

sin, cos, tan	- trigonometrische Funktionen
asin, acos, atan	- inverse trigon. Funktionen
exp, log	- e-Funktion, natürl. Logarithmus
log10	- Logarithmus zur Basis 10
sqrt	- Quadratwurzel
round, fix	- Rundungsbefehle
abs, sign	- Betrag und Vorzeichen
angle	- Phase einer komplexen Zahl

Matrizen

ones, zeros	- Matrix aus 1 oder 0
eye	- Einheitsmatrix
magic	- magisches Quadrat
rand, randn	- Zufallszahlen
diag	- Diagonalmatrix
fliplr, flipud	- vertikal oder horizontal spiegeln
meshgrid	- Gitter erzeugen

Statistische Operatoren (spaltenweise)

sum, cumsum	- Summe und kumulative Summe
diff	- Differenz aufeinanderfolgender Elemente
mean, std	- Mittelwert, Standardabweichung
min, max	- Minimum, Maximum
sort	- sortieren

Lineare Algebra (matrixweise)

transp oder '	- transponieren (' konjugiert zusätzlich komplex!)
inv	- invertieren
eig	- Eigenwerte
rank	- Rang
det, trace	- Determinante, Spur
dot, cross	- Skalarprodukt, Kreuzprodukt

Graphik

plot, plot3	- Liniengraphik
hist, bar	- Balkendarstellung
surf, mesh	- 3D-Oberflächen
shading	- Schattierung
colormap	- Farbtabelle
colorbar	- zeigt Farbskalierung als Balken
contour, clabel	- Höhenlinien mit Beschriftung
quiver	- Vektordarstellung
axis, grid	- Achsenmanipulation, Gitterlinien
xlim, ylim	- manuelle Achsenskalierung
view	- Ansicht von 3D-Graphiken
xlabel, ylabel, zlabel	- Achsenbeschriftung
title	- Graphiküberschrift
text, gtext	- Textpositionierung (auch interaktiv)
legend	- Legende
subplot	- Darstellung mehrerer Graphiken auf einer Seite
hold (on/off)	- mehrere plots in einem Fenster
grid (on/off)	- Gitternetzlinien anzeigen
figure	- Öffnen eines neuen Graphikfensters
close (all)	- Schließen von Graphikfenstern
get, set	- Graphikeigenschaften abfragen bzw. manipulieren
print	- Graphiken drucken

Programmierstrukturen

function [output]	- Kopf einer Funktionsdatei
= Name(input)	(m-File)
nargin	- Zahl der Eingabeargumente
nargout	- Zahl der Ausgabeargumente
if ... elseif ... else ... end	- if-Block
for ... end	- for-Schleife
while ... end	- while-Schleife
break	- Verlassen einer Schleife
input	- Tastatur-Eingabe
disp, sprintf	- Ausgabe am Bildschirm
error	- Programmabbruch mit Fehlermeldung
return	- Verlassen einer Funktion

1. Variable

- a) Berechnen Sie $2.5 \cdot \sqrt{3} \cdot \sin(2\pi \cdot (33.3 + f_0))$ für $f_0 = 20$.
- b) Berechnen Sie Betrag und Winkel (in Grad) von $a = 3 + j$ und $b = -2 - j$.
- c) Erzeugen Sie den Vektor $\mathbf{v} = \begin{pmatrix} 1 \\ 2 \\ -3 \\ 4 + j \end{pmatrix}$.
- d) Berechnen Sie $\boldsymbol{\omega} = \mathbf{v}^T$.
- e) Erzeugen Sie die Matrix $\mathbf{M} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & j & 1 \\ j & j + 1 & -3 \end{pmatrix}$
- f) Erweitern Sie \mathbf{M} zu $\mathbf{V} = \begin{pmatrix} \mathbf{M} & \mathbf{M} \\ \mathbf{M} & \mathbf{M} \end{pmatrix}$.
- g) Ordnen Sie die 4. Zeile von \mathbf{V} einem neuen Vektor $\mathbf{z4}$ zu.
- h) Ersetzen Sie in \mathbf{V} die Zelle in Zeile 4 und Spalte 2 durch den Wert 5.8.
- i) Im Command-Window soll folgender Text ausgegeben: „Blindleistung = $\langle Q \rangle$ var“. Anstatt $\langle Q \rangle$ soll dabei variabel der Wert erscheinen, den Q durch vorherige Operationen erhalten hat. Weisen Sie in diesem Fall Q den Wert 40 zu. Informieren Sie sich in der Matlab-Hilfe über die Funktionen *num2str* und *disp* oder *sprintf*.

2. Mathematische Operationen

- a) Definieren Sie $\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$ und $\mathbf{B} = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}$.
- b) Invertieren Sie \mathbf{A} und \mathbf{B} .
- c) Multiplizieren Sie jedes Element von \mathbf{A} mit 3.
- d) Multiplizieren Sie \mathbf{A} und \mathbf{B} .
- e) Multiplizieren Sie \mathbf{A} und \mathbf{B} elementweise.
- f) Erzeugen Sie einen Zeitvektor $\mathbf{t} = (0.0 \ 0.1 \ 0.2 \ \dots \ 0.5)$.
- g) Berechnen Sie für alle Elemente t_i von \mathbf{t} die Funktionswerte $y_i = \sin(2\pi \cdot 6 \cdot t_i) + \cos(2\pi \cdot 3 \cdot t_i) + e^{-0.1 \cdot t_i}$ und weisen Sie sie einem Vektor \mathbf{y} zu.
- h) Berechnen Sie für alle Elemente t_i von \mathbf{t} die Funktionswerte $z_i \sin(2\pi \cdot 6 \cdot t_i) \cdot \cos(2\pi \cdot 3 \cdot t_i) \cdot e^{-0.1 \cdot t_i}$ und weisen Sie sie einem Vektor \mathbf{z} zu.
- i) Stellen Sie \mathbf{t} , \mathbf{y} und \mathbf{z} als dreispaltige Matrix mit dem Namen **TAB** dar.
- j) Der Vektor $p = (1024 \ 1000 \ 100 \ 2 \ 1 \ \frac{1}{\sqrt{2}})$ enthalte Leistungsverstärkungen verschiedener Verstärkerschaltungen. Berechnen Sie zu die zugehörigen Werte in dB.

3. Grafische Ausgabe

- a) Weisen Sie einem Vektor \mathbf{t} für den Zeitbereich $0 \dots 2s$ die folgenden Spannungen zu:
 u_1 sei eine 3V-Sinusspannung mit einer Frequenz von 3Hz,
 u_2 sei eine um -30° phasenverschobene 5V-Kosinusspannung mit einer Frequenz von 5Hz,
 u_3 sei eine von 6V abklingende Exponentialfunktion mit der Zeitkonstante $\tau = 2s$.

- b) Stellen Sie diese drei Spannungen folgendermaßen dar:
- in getrennte Plotfenster (*figure*),
 - verschiedenfarbig in ein Plotfenster (*plot*),
 - in ein Plotfenster mit drei Unterfenstern (*subplot*).
- c) Dokumentieren Sie die Ergebnisse aus iii) mit den Farben von ii) in einem MS-Word- oder OpenOffice-Dokument.
Stellen Sie die Grafiken dazu einheitlich je in einem $\pm 7.5V$ -Messfenster dar, betiteln und beschriften Sie sie sinnvoll und versehen Sie sie mit einem Gitternetz.

4. Programmierung

Erstellen Sie ein neues m-File (Matlab-Skript-File) und erstellen Sie die Befehle für das folgende Programm. Testen Sie anschließend Ihr m-File durch Eingabe des Dateinamens.

Definieren Sie einen Zeitvektor t für den Zeitbereich $0 \dots 40ms$ und ein $10\mu s$ -Raster. Berechnen Sie für die in t angegebenen Zeiten t_i die Summe

$$u(t_i) = \sum_{k=1}^{k_{max}} u_k(t_i)$$

der Signale u_k , wobei jedes Signal u_k sinusförmig mit der Frequenz $f_k = k \cdot 50Hz$ ist, die Amplitude $A_k = 0$ für gerade k bzw. $A_k = \frac{4}{k\pi}$ für ungerade k besitzt und $k_{max} = 50$ beträgt.

Verwenden Sie dazu eine *for ... end* Schleife und stellen Sie die Summenfunktion graphisch dar. (Tipp: Nutzen Sie die Schrittweite 2 in der *for*-Schleife). Experimentieren Sie mit unterschiedlichen Werten von k_{max} und finden Sie heraus, wieviele Schwingungen überlagert werden müssen, um eine hinreichende Qualität der Wellenform zu erreichen.

5. Funktionen

- Schreiben Sie eine MATLAB Funktion *test* zur Berechnung der nichtlinearen Kennlinie

$$y = \begin{cases} 1 - e^{-x} & \text{für } x \geq 0 \\ -(1 - e^x) & \text{für } x < 0 \end{cases}$$

- Die Funktion *test* soll dann durch den Aufruf von *test(x)* den jeweiligen Funktionswert für x zurückliefern. Die Funktion soll so geschrieben sein, dass x eine Zahl, aber auch ein Zeilenvektor sein darf.
- Plotten Sie den Graf dieser Kennlinie im Bereich $-5 \leq x \leq 5$.

Literatur

- [1] Angermann, Beuschel, Rau, Wohlfarth: *Matlab – Simulink – Stateflow*, Oldenbourg Verlag, 2005. ISBN 3-486-57719-0
- [2] Schweizer, W.: *Matlab kompakt*. Oldenbourg Verlag, München, 2008. ISBN 3-486-58735-8

Links

- www.etech.haw-hamburg.de/matlab - Download von Matlab (Campuslizenz der HAW)
<http://www.imrtweb.ethz.ch/matlab/> - Interaktives Tutorial der ETH Zürich (deutschsprachig)
www.mathworks.de