

# Einführung in MATLAB/Simulink

Berechnung, Programmierung, Simulation

Bearbeitet von  
Angelika Bosl

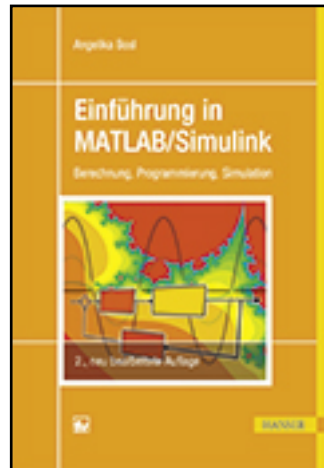
2., neu bearbeitete Auflage 2017. Buch. 331 S. Hardcover  
ISBN 978 3 446 44269 6

[Weitere Fachgebiete > EDV, Informatik > Datenbanken, Informationssicherheit,  
Geschäftssoftware > Mathematische & Statistische Software](#)

schnell und portofrei erhältlich bei

  
DIE FACHBUCHHANDLUNG

Die Online-Fachbuchhandlung [beck-shop.de](http://beck-shop.de) ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.



Leseprobe

Angelika Bosl

Einführung in MATLAB/Simulink

Berechnung, Programmierung, Simulation

ISBN (Buch): 978-3-446-44269-6

ISBN (E-Book): 978-3-446-44770-7

Weitere Informationen oder Bestellungen unter

<http://www.hanser-fachbuch.de/978-3-446-44269-6>

sowie im Buchhandel.

# Vorwort

MATLAB/SIMULINK ist ein äußerst leistungsfähiges interaktives Programmpaket für vorwiegend numerische Berechnungen im Ingenieurbereich. Diese Software ist weltweit verbreitet und in Fachkreisen bekannt für die Berechnung, Modellierung und Simulation technischer Systeme, sowohl an den Hochschulen als auch in der Industrie.

Hinter MATLAB verbirgt sich ein sehr umfangreiches Softwarepaket, gebündelt aus verschiedenen so genannten „Toolboxen“, Werkzeugen, die jeweils bestimmte Bereiche der Ingenieurwissenschaften abdecken. Die bekannteste Toolbox dürfte SIMULINK sein, ein unschlagbares Werkzeug zur grafischen Simulation technischer Abläufe und mathematischer Modelle. Weitere Toolboxen sind zum Beispiel die „Control System Toolbox“ mit spezifischen Befehlen für regelungstechnische Aufgaben oder die „Signal Processing Toolbox“ zur Signaldatenverarbeitung.

Des Weiteren gibt es für Toolboxen zu Berechnungen im Finanzwesen genauso wie in der Biologie, Messtechnik und Datenerfassung, Bilddatenverarbeitung und viele mehr. Eine aktuelle Liste ist auf der Homepage von MathWorks, dem Herausgeber von MATLAB unter [www.mathworks.de](http://www.mathworks.de) zu finden.

Wie viele umfangreiche Softwarepakete, kann auch MATLAB den unerfahrenen Benutzer bei den ersten Versuchen, die Software zu bedienen, regelrecht „erschlagen“. Die zugehörige Hilfe und ein Großteil der Fachliteratur sind auf Englisch und sehr spezialisiert in verschiedenen Bereichen. Das erklärte Ziel der vorliegenden Einführung in MATLAB/SIMULINK ist es deshalb, den Leser mit einer anschaulichen Anleitung, hilfreichen Hinweisen und Tipps für die Anwendung sowie mit praxisnahen Beispielen zu unterstützen.

Dem Erstbenutzer von MATLAB soll der Einstieg in die Software erleichtert werden, damit kein Frust beim ersten Programmstart aufkommt, sondern sofort erfolgreich mit der Bedienung der Software gestartet werden kann. Jeder sollte gleich in der Lage sein, sich auf dem Startbildschirm zu orientieren, verschiedene Befehle auszuführen und einfache Aufgaben zu lösen. Das Buch will ermutigen, sich näher mit der Software auseinanderzusetzen, und so ein erfolgreiches Arbeiten gewährleisten.

Dieses Lehrbuch kann und will nur eine Einführung sein, die zwar die wichtigsten, aber natürlich nicht alle Aspekte berücksichtigen kann. Die Syntax der Befehle, Grundlagen zum Verständnis von MATLAB und bestimmter Toolboxen sollen Hilfe zur Selbsthilfe geben, sodass auch spezifische eigene Aufgaben anschließend deutlich leichter in Eigenregie erarbeitet und gelöst werden können.

Weiterführende Literatur, auch für spezielle Fachgebiete, wird dem Interessierten, der die Anfänge hinter sich lassen und in die Tiefe einsteigen möchte, empfohlen, zum Beispiel der „Einstieg in das Programmieren mit MATLAB“ des Hanser Verlags<sup>1</sup>.

---

<sup>1</sup> Ulrich Stein: Einstieg in das Programmieren mit MATLAB, Hanser Verlag, 2009

Bevor es richtig losgeht, möchte ich mich bei allen ganz herzlich bedanken, die zum Entstehen dieses Buches beigetragen haben:

- Herrn Prof. Dipl.-Math. Wolfgang Georgi, durch dessen Buch „Einführung in LabVIEW“ des Hanser Verlags erst die Idee zu dieser MATLAB Einführung entstand und der mir immer mit Rat und Tat behilflich war.
- Herrn Prof. Dr.-Ing. Hans-Jürgen Adermann dafür, dass ich in seinem Labor „Regelungstechnik“ an der Hochschule Ravensburg-Weingarten MATLAB/SIMULINK kennenlernen und mich intensiv damit auseinandersetzen konnte. Seiner Vorlesung und den dazugehörigen praktischen Übungen ist das Kapitel zur Regelungstechnik zu verdanken.
- Herrn Dipl.-Ing. (FH) Wolfgang Reich für das Korrekturlesen des Manuskripts und seine guten Ratschläge und Nachfragen den Inhalt und Formulierungen betreffend.
- Allen meinen Freunden, die mich unermüdlich ermuntert haben, die Arbeit nicht aufzugeben und ihr Mitgefühl bezeugt haben.
- Und schließlich dem Hanser Fachbuchverlag Leipzig, im Besonderen Frau Werner, Herrn Feuchte und Frau Kaufmann, die viel Geduld mit mir bewiesen haben.

Baienfurt, Mai 2012

A. Bosl

## ■ Vorwort zur zweiten Auflage

Eine Software, die immer auf dem gleichen Stand bleibt, taugt nichts. Software muss sich ändern, muss neuen Gegebenheiten angepasst werden, muss sich weiter entwickeln. MATLAB/SIMULINK hat sich verändert. Seit der ersten Auflage, die noch auf MATLAB R2009a basiert, sind zur jetzigen Version MATLAB R2016a erhebliche Änderungen zu erkennen. Am Auffälligsten ist natürlich das äußere Erscheinungsbild. Sobald man sich näher mit den Funktionen und den einzelnen Toolboxen befasst, stellt man jedoch schnell fest, dass auch die Funktionalität erweitert und die Möglichkeiten noch umfangreicher wurden. Nicht ganz verständlich sind geringfügige Änderungen, wie andere Fehlermeldungen beim Eingeben eines falschen Befehls, die aber vermutlich vor allem denjenigen auffallen, die einen Text auf Änderungen überarbeiten müssen.

Eine erhebliche Änderung betrifft sogar die ursprüngliche Beschreibung von MATLAB/SIMULINK als „äußerst leistungsfähiges interaktives Programmpaket für numerische Berechnungen im Ingenieurbereich“. Wie dem aufmerksamen Leser vielleicht aufgefallen ist, wurde dieser erste Satz des Vorworts durch ein „vorwiegend“ ergänzt. Mit der „*Symbolic Math Toolbox*“ wurden die Toolboxen durch ein Werkzeug ergänzt, mit dem mathematische Gleichungen nicht numerisch, sondern analytisch gelöst, verändert und dargestellt werden können. MATLAB verwendet dafür den Begriff „symbolisch“ und die Bedeutung der „*Symbolic Math Toolbox*“ und ihre Möglichkeiten können leicht unterschätzt werden: MATLAB ist kein rein numerisches Programmpaket mehr!

Ein wenig vermisse ich den sehr einprägsamen Begriff „*M-File*“ für MATLAB-Programme, der aber vermutlich mit Absetzen der Fernsehserie „*X-Akten*“ (engl. „*X Files*“) ebenfalls sein Ende

find. Dafür wurde der Editor für MATLAB-Code (nicht mehr „*M-File Editor*“) in seiner Funktionalität erweitert.

Den Dankesworten der 1. Auflage möchte ich meinen herzlichen Dank an Frau Jacob vom Carl Hanser Verlag hinzufügen, die sehr viel Geduld mit mir aufbringen musste, bis ich die überarbeitete 2. Auflage schließlich fertig hatte.

Ich möchte dieses Buch meinem Vater Alexander Bosl widmen, der während der letzten Arbeiten an der zweiten Auflage, im März 2017 gestorben ist. Meinem Vater verdanke ich die Zuversicht in die eigenen Fähigkeiten, egal ob es darum geht, sich in eine mathematische Software einzuarbeiten oder ein Buch darüber zu schreiben.

Baienfurt, April 2017

A. Bosl

# Inhalt

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Einleitung</b> .....  | <b>15</b> |
| 1.1      | Warum MATLAB/SIMULINK? .....                                       | 15        |
| 1.2      | MATLAB-/SIMULINK-Versionen .....                                   | 16        |
| 1.3      | Installation der Software .....                                    | 18        |
| <b>2</b> | <b>Start der Arbeit mit MATLAB</b> .....                           | <b>21</b> |
| 2.1      | Grundlagen zum MATLAB-Desktop .....                                | 21        |
| 2.2      | MATLAB-Fenster .....   | 23        |
| 2.2.1    | „Command Window“, das Befehlsfenster .....                         | 23        |
| 2.2.2    | „Current Directory“, das aktuelle Arbeitsverzeichnis .....         | 24        |
| 2.2.3    | „Workspace“, der Arbeitsbereich oder Arbeitsspeicher .....         | 25        |
| 2.2.4    | „Command History“, die Chronik der Befehle .....                   | 29        |
| 2.3      | Funktionen der Menüleiste („ <i>Toolstrip</i> “) .....             | 31        |
| 2.4      | MATLAB-Hilfe und Beschreibungen der Befehle .....                  | 37        |
| <b>3</b> | <b>Zahlen, Vektoren und Matrizen</b> .....                         | <b>46</b> |
| 3.1      | Darstellung von Zahlen .....                                       | 46        |
| 3.2      | Umrechnung von Zahlen .....  | 48        |
| 3.3      | Definition von Variablen als Skalare, Vektoren oder Matrizen ..... | 52        |
| 3.3.1    | Definieren von Variablen .....                                     | 52        |
| 3.3.2    | Spalten- und Zeilenvektoren .....                                  | 53        |
| 3.3.3    | Matrizen Werte zuordnen .....                                      | 55        |
| 3.3.4    | Spezielle Matrizen .....   | 58        |
| 3.3.5    | Größe eines Vektors oder einer Matrix .....                        | 64        |
| 3.3.6    | Maximal- und Minimalwerte bestimmen .....                          | 65        |
| 3.3.7    | Statistische Charakteristika bestimmen .....                       | 66        |
| <b>4</b> | <b>Mathematische Berechnungen mit MATLAB</b> .....                 | <b>70</b> |
| 4.1      | Grundrechenarten .....   | 70        |
| 4.2      | Elementare mathematische Funktionen .....                          | 74        |
| 4.3      | Trigonometrische Funktionen .....                                  | 76        |

|        |  |    |
|--------|--|----|
| 4.4    | Relationale Operatoren .....                                 | 77 |
| 4.5    | Logische Operatoren .....                                    | 78 |
| 4.6    | Besonderheiten beim Rechnen mit Vektoren und Matrizen .....  | 82 |
| 4.6.1  | Vektoraddition und -subtraktion .....                        | 83 |
| 4.6.2  | Transponieren einer Matrix oder eines Vektors .....          | 83 |
| 4.6.3  | Invertieren einer quadratischen Matrix .....                 | 84 |
| 4.6.4  | Rang einer Matrix mit rank .....                             | 85 |
| 4.6.5  | Determinante einer quadratischen Matrix .....                | 86 |
| 4.6.6  | Matrixmultiplikation .....                                   | 88 |
| 4.6.7  | Multiplikation einer Matrix mit einem Skalar .....           | 90 |
| 4.6.8  | Potenzieren einer Matrix .....                               | 92 |
| 4.6.9  | Vektor-Matrix-Produkt .....                                  | 92 |
| 4.6.10 | Linke Matrixdivision (engl. „backslash division“) .....      | 93 |
| 4.6.11 | Rechte Matrixdivision (engl. „slash division“) .....         | 94 |
| 4.7    | Spezielle Matrixmanipulationen .....                         | 94 |
| 4.7.1  | Spezielle mathematische Befehle für Matrizen .....           | 94 |
| 4.7.2  | Spezielle Teilbereiche einer Matrix extrahieren .....        | 95 |
| 4.8    | Feldoperationen: Elementweise Verknüpfung von Vektoren ..... | 97 |
| 4.8.1  | Elementweise Multiplikation (engl. „array multiply“) .....   | 97 |
| 4.8.2  | Elementweise Division .....                                  | 98 |
| 4.8.3  | Elementweises Potenzieren .....                              | 99 |

## **5 Grafische Darstellungen von Funktionen .....** 101

|       |   |     |
|-------|---|-----|
| 5.1   | Einfache Grafiken und Diagramme mit plot .....                                  | 101 |
| 5.2   | Grafikeigenschaften – „Figure Properties“ .....                                 | 103 |
| 5.2.1 | Farbpaletten auswählen mit colormap .....                                       | 103 |
| 5.2.2 | „Figure Properties“ über die Befehlszeile definieren .....                      | 103 |
| 5.2.3 | „Properties“ über die Menüleiste im Grafikenster bestimmen .....                | 108 |
| 5.2.4 | Grafikeigenschaften („Properties“) mit dem „Property Editor“<br>verändern ..... | 115 |
| 5.3   | Mehrere Diagramme in einem Grafikenster .....                                   | 117 |
| 5.3.1 | Mehrere Kurven oder Diagrammtypen in einem Diagramm mit hold ...                | 118 |
| 5.3.2 | Unterdigramme in einem Grafikenster mit subplot .....                           | 118 |
| 5.4   | Grafiktypen im zweidimensionalen Bereich .....                                  | 119 |
| 5.5   | Grafiktypen im dreidimensionalen Bereich .....                                  | 130 |
| 5.6   | Grafiken erzeugen über den Tab „PLOTS“ der Titelleiste .....                    | 140 |

|          |  |            |
|----------|--|------------|
| <b>6</b> | <b>Programmieren in MATLAB .....</b>   | <b>144</b> |
| 6.1      | Editor .....   | 144        |
| 6.2      | Varianten der Programmiervorlagen .....  | 153        |
| 6.3      | „Script“ – Einfache Befehlsfolgen .....  | 154        |
| 6.4      | Kontrollstrukturen für die komplexere Programmierung .....   | 156        |
| 6.4.1    | for-Schleife.....  | 157        |
| 6.4.2    | while-Schleife.....  | 159        |
| 6.4.3    | if-elseif-else-Verzweigung .....   | 161        |
| 6.4.4    | switch-case-otherwise-Verzweigung .....  | 163        |
| 6.4.5    | try-catch-Fehlerkontrolle .....  | 165        |
| 6.4.6    | Weitere Befehle, die den Programmablauf beeinflussen .....   | 166        |
| 6.5      | Nützliche Befehle für die Programmierung unter MATLAB .....  | 169        |
| 6.6      | „Function“ – Funktionen in MATLAB.....   | 174        |
| 6.6.1    | Kopfzeile einer Funktion (Syntax) .....  | 174        |
| 6.6.2    | Aufbau einer Funktion .....  | 175        |
| 6.6.3    | Verschachtelte Funktionen .....  | 176        |
| 6.7      | „Class“ – Objektklassen in MATLAB.....   | 177        |
| <b>7</b> | <b>„Control System Toolbox“ –</b>  |            |
|          | <b>Alles was man für die Regelungstechnik braucht .....</b>  | <b>179</b> |
| 7.1      | Eingabe der Übertragungsfunktion $G_s$ eines Regelkreises .....  | 180        |
| 7.1.1    | Befehl <code>tf</code> .....   | 180        |
| 7.1.2    | Befehl <code>conv</code> zur Polynommultiplikation .....   | 181        |
| 7.2      | Zusammenschaltung von Modellen (Signalflussplan-Algebra) .....   | 182        |
| 7.2.1    | Reihen-, Serien- oder Kettenschaltung .....  | 182        |
| 7.2.2    | Parallelschaltung .....  | 183        |
| 7.2.3    | Übertragungsfunktion mithilfe der Laplace-Variablen $s$ .....  | 185        |
| 7.2.4    | Polform einer Übertragungsfunktion mit <code>zpk</code> .....  | 186        |
| 7.2.5    | Befehl <code>feedback</code> zur Berechnung des geschlossenen Regelkreises –<br>Führungsübertragungsfunktion ..... | 187        |
| 7.3      | Grafische Darstellungsmöglichkeiten für Übertragungsfunktionen .....   | 188        |
| 7.3.1    | Impulsantwort (Gewichtsfunktion) mit <code>impz</code> .....   | 189        |
| 7.3.2    | Sprungantwort (Übergangsfunktion) mit <code>step</code> .....  | 191        |
| 7.3.3    | Bode-Diagramm (Frequenzgang) mit <code>bode</code> .....   | 194        |
| 7.3.4    | Nyquist-Ortskurve mit <code>nyquist</code> .....   | 196        |
| 7.3.5    | Nichols-Ortskurve mit <code>nichols</code> .....   | 198        |
| 7.3.6    | Pol- und Nullstellendiagramm mit <code>pzmap</code> .....  | 199        |
| 7.3.7    | Wurzelortskurve (WOK) mit <code>rlocus</code> .....  | 201        |



|          |  |            |
|----------|--|------------|
| 7.4      | Charakteristika einer Übertragungsfunktion .....   | 202        |
| 7.4.1    | Befehl <code>po1e</code> zur Berechnung der Pole einer Übertragungsfunktion .....                                | 202        |
| 7.4.2    | Befehle <code>tzero</code> (engl. transmission zeros) und <code>zero</code> zur Berechnung der Nullstellen ..... | 203        |
| 7.4.3    | Befehl <code>get</code> zur Ausgabe der Eigenschaften einer Übertragungsfunktion .....                           | 203        |
| 7.4.4    | Befehl <code>set</code> zum Setzen von Eigenschaften einer Übertragungsfunktion .....                            | 207        |
| 7.4.5    | Befehl <code>margin</code> .....   | 211        |
| 7.5      | Einfacher Reglerentwurf mit MATLAB .....   | 213        |
| 7.5.1    | Bestimmung des Verstärkungsfaktors $K_V$ mit dem Bode-Diagramm .....   | 216        |
| 7.5.2    | Bestimmung des Regel- oder Verstärkungsfaktors $K_V$ mithilfe der Wurzelortskurve (WOK) .....                    | 228        |
| 7.5.3    | „Control System Designer“ zum Reglerentwurf – <code>sisotool</code> .....  | 231        |
| 7.5.3.1  | Tab „Control System“ .....   | 232        |
| 7.5.3.2  | Tab „ROOT LOCUS EDITOR“, „BODE EDITOR“ bzw. „NICHOLS EDITOR“ .....   | 232        |
| 7.5.3.3  | Tab „VIEW“ .....   | 235        |
| 7.5.3.4  | „Graphical Tuning“ – Grafische Methoden zur Regleroptimierung .....  | 235        |
| 7.5.3.5  | „Automated Tuning“ – Automatisierte Regleroptimierung anhand vorgegebener Parameter .....                        | 248        |
| <b>8</b> | <b>Einführung in die SIMULINK-Toolbox .....</b>  | <b>252</b> |
| 8.1      | Erste Schritte in SIMULINK .....   | 252        |
| 8.2      | Wichtige Funktionen in der Menüleiste einer SIMULINK-Simulation .....  | 258        |
| 8.2.1    | Menüpunkt „File“ .....   | 259        |
| 8.2.2    | Menüpunkt „Edit“ .....   | 267        |
| 8.2.3    | Menüpunkt „View“ .....   | 268        |
| 8.2.4    | Menüpunkt „Display“ .....  | 271        |
| 8.2.5    | Menüpunkt „Diagram“ .....  | 275        |
| 8.2.6    | Menüpunkt „Simulation“ .....   | 278        |
| 8.2.7    | Menüpunkt „Analysis“ .....   | 280        |
| 8.2.8    | Menüpunkt „Code“ .....   | 285        |
| 8.2.9    | Menüpunkt „Tools“ .....  | 285        |
| 8.3      | Kurzbeschreibung der Icons der Symbolleiste („ <i>Toolbar</i> “) .....   | 286        |
| 8.4      | Kurzbeschreibung der wichtigsten SIMULINK-Blöcke .....   | 287        |
| 8.5      | Tipps & Tricks für Regelkreis-Simulationen .....   | 292        |
| 8.6      | Tipps zur Auswertung grafischer Ergebnisse des <i>Scope</i> .....  | 305        |
| 8.6.1    | Ändern der grafischen Darstellung im Bildbearbeitungsprogramm .....  | 305        |
| 8.6.2    | Konfigurierbare Darstellung des <i>Scope</i> -Fensters über MATLAB .....   | 305        |

---

|          |   |            |
|----------|---|------------|
| <b>A</b> | <b>MATLAB-Befehlsliste für die Abbildungen<br/>der zweidimensionalen Grafikbeispiele in Abschnitt 5.4.....</b>                            | <b>308</b> |
| <b>B</b> | <b>MATLAB-Befehlsliste für die Abbildungen<br/>der dreidimensionalen Grafikbeispiele in Abschnitt 5.5.....</b>                            | <b>312</b> |
| <b>C</b> | <b>MATLAB-Programm zur Berechnung eines optimierten Reglers<br/>mithilfe des Bode-Diagramms und des <code>margin</code>-Befehls .....</b> | <b>316</b> |
|          | <b>Literatur .....</b>  | <b>319</b> |
|          | <b>Index .....</b>  | <b>321</b> |

# 2

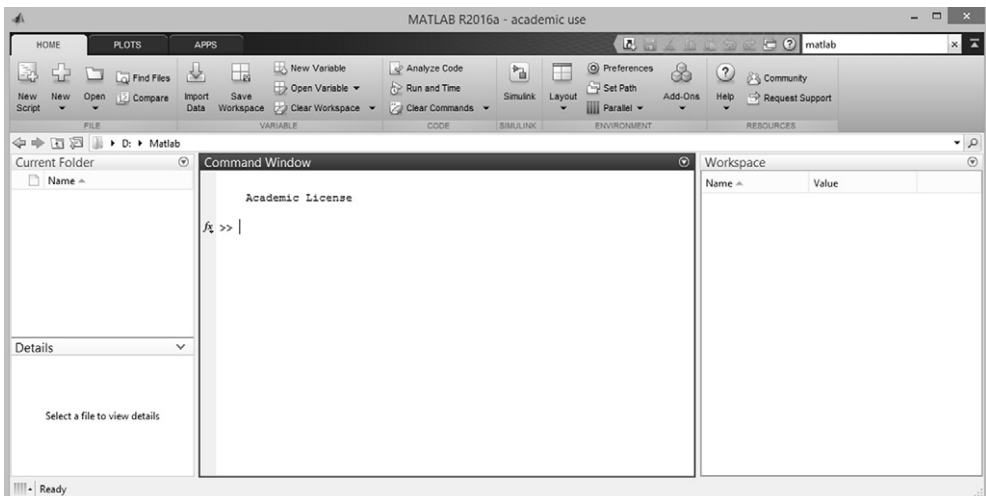
## Start der Arbeit mit MATLAB

*In diesem Kapitel soll die Basis für einen erfolgreichen Start mit MATLAB geschaffen werden, denn nichts ist ärgerlicher beim Öffnen eines neuen, unbekanntes Programms, als wenn man nicht gleich loslegen kann, sondern erst mühsam herausfinden muss, was wo zu finden ist und wozu die vielen Icons auf dem Bildschirm wohl gut sein könnten.*

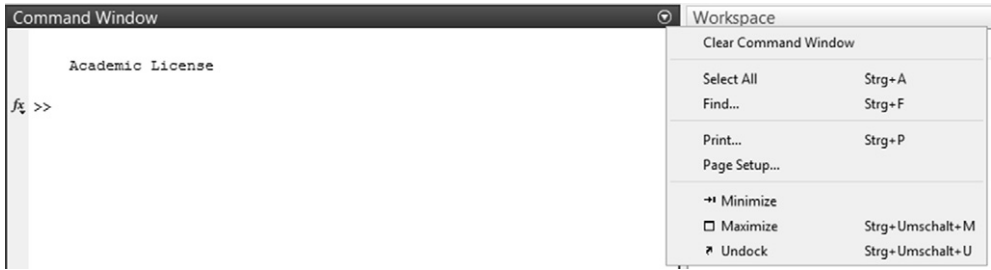
### ■ 2.1 Grundlagen zum MATLAB-Desktop

Nach dem Starten von MATLAB erscheint der MATLAB-Desktop standardmäßig mit vier Fenstern unter der MATLAB-Taskleiste (siehe Bild 2.1).

Über ein Kontextmenü, das mit Mausclick auf das Pfeilsymbol rechts vom Fensternamen, geöffnet wird, siehe Bild 2.2, können verschiedene Aktionen durchgeführt werden. Der Inhalt von jedem Fenster kann gelöscht werden, z. B. löscht „Clear Command Window“ jeden Text aus dem „Command Window“. Es können alle Daten ausgewählt werden, bestimmte Suchbegriffe gefunden werden und der Inhalt kann ausgedruckt werden. Am wichtigsten sind jedoch die Befehle, die die Fenster an sich betreffen. Mit „Minimize“ wird ein Fenster minimiert, das Fenster ist nur noch über einen Tab am Seitenrand zu öffnen, mit „Maximize“ wird ein Fenster auf die Gesamtgröße des MATLAB-Fensters vergrößert, alle anderen Fenster treten in den



**Bild 2.1** MATLAB – hier die Version der Campuslizenz – nach dem Start mit standardmäßig vier Fenstern: „Current Directory“, „Details“, „Command Window“ und „Workspace“ (von links nach rechts)



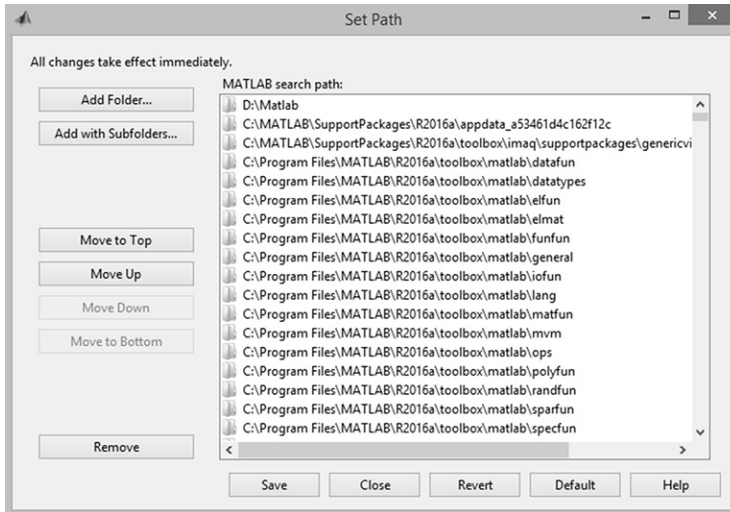
**Bild 2.2** Optionen zum Anpassen der verschiedenen Fenster des MATLAB-Desktops

Hintergrund. „Undock“ löste ein Fenster aus dem Verbund zu einem eigenständigen Fenster, der gegenteilige Befehl lautet „Dock“ für ein separates Fenster. Mit „Close“ können alle Fenster, außer dem Command Window, welches immer offen sein muss, geschlossen werden. Mit „Restore“ kann die ursprüngliche Anordnung der Fenster wieder hergestellt werden.

In der Menüleiste findet sich in der Gruppe „ENVIRONMENT“ die Option „Layout“ mit der der Ursprungszustand („Default“) ebenfalls wieder hergestellt werden kann, für den Fall, dass z. B. ein Fenster aus Versehen geschlossen wurde und nicht mehr gefunden wird.

### „Set Path...“ – Einbinden eigener Verzeichnisse

Auf der Taskleiste findet sich in der Gruppe „ENVIRONMENT“ der Befehl „Set Path“. In dem sich öffnenden Dialogfenster, siehe Bild 2.3, besteht die Möglichkeit, eigene Dateiverzeichnisse dem MATLAB-Suchpfad hinzuzufügen, mit Unterverzeichnissen („Add with Subfolders...“) oder ohne („Add Folder...“). Dies ist sehr nützlich, wenn in einem speziellen Verzeichnis oder Ordner eigene MATLAB-Dateien angelegt werden, die von MATLAB nur dann aufgerufen wer-



**Bild 2.3** „Set Path“-Dialogfenster zum Hinzufügen eigener Verzeichnisse

den können, wenn dieses Verzeichnis oder die Verzeichnisse dem MATLAB-Suchpfad hinzugefügt wurden. Zum Abschluss muss der Suchpfad mit „Save“ gespeichert werden, damit die Änderungen auch übernommen werden.

Natürlich kann auch unter dem Standardverzeichnis gearbeitet werden, das MATLAB automatisch im Verzeichnis: `..\Eigene Dateien\MATLAB`.

## ■ 2.2 MATLAB-Fenster

Im Folgenden werden die verschiedenen Fenster und ihre jeweilige Funktion erläutert, die beim Start von MATLAB zu sehen sind.

### 2.2.1 „Command Window“, das Befehlsfenster

In der Mitte sticht das wichtigste und größte Fenster heraus, das Befehlsfenster oder „*Command Window*“, in dem hauptsächlich gearbeitet wird. Hier werden die Befehle eingegeben und Funktionen gestartet, aber auch die Ergebnisse der Berechnungen wiedergegeben, oder manchmal Fehlermeldungen, siehe Bild 2.4. Die interaktive Bedienung von MATLAB gestaltet sich sehr einfach mithilfe einer Interpretersprache. Alternativ oder in Ergänzung zur interaktiven Bedienung können MATLAB-Befehlsfolgen als Batchprogramme bzw. als MATLAB-Code ablaufen (siehe Kap. 6).

Hinter dem Zeichen `>>` lädt MATLAB dazu ein, Eingaben zu machen. Wenn dieses Zeichen fehlt, befindet sich MATLAB noch in einer – etwas länger dauernden – Berechnung oder eine Eingabe wurde noch nicht richtig abgeschlossen.

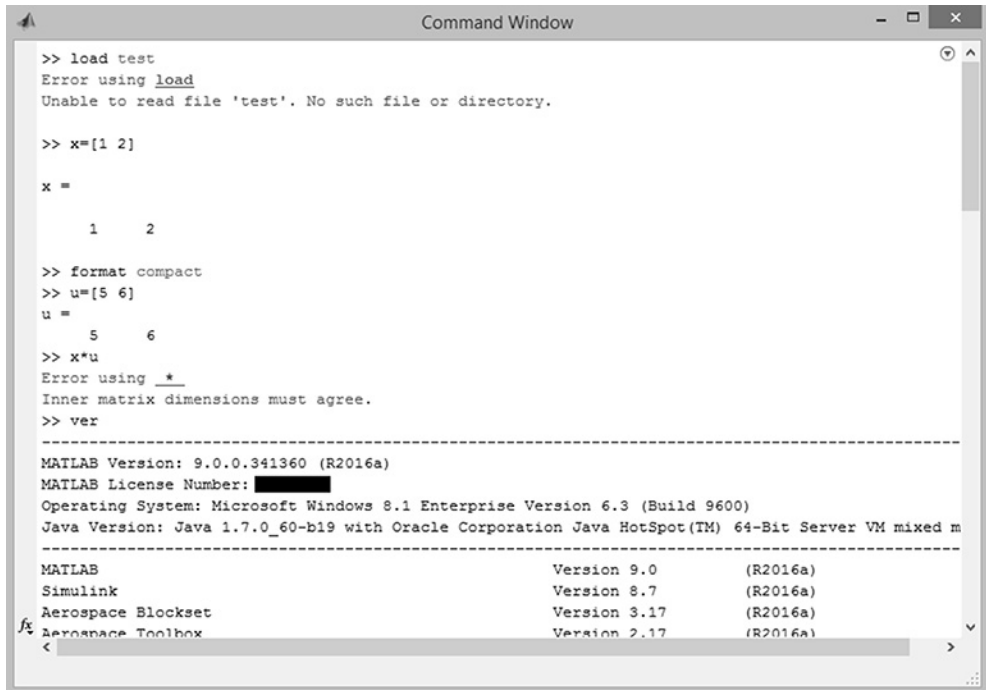
Alle Befehle und Variablenzuweisungen werden immer im „*Command Window*“, dem Befehlsfenster von MATLAB, hinter dem `>>`-Eingabezeichen eingegeben, siehe willkürliche Beispiele in Bild 2.4.

Es können hinter der `>>`-Eingabeaufforderung oder hinter auszuführenden MATLAB-Befehlen aber auch Eingaben gemacht werden, die nur zur Kommentierung dienen. Diese Kommentare werden durch das Prozentzeichen `%` eingeleitet und durch die Eingabetaste abgeschlossen. Kommentare werden von MATLAB automatisch in grüner Schrift kenntlich gemacht. Normalerweise macht ein Kommentar im „*Command Window*“ wenig Sinn, in den MATLAB-Programmen, vgl. Kap. 6, sind sie jedoch sehr nützlich.

```
>> % Dies ist ein Kommentar!
```



In diesem Buch werden Kommentare hin und wieder verwendet, um einzelne Befehle direkt, schnell und in Kürze zu erläutern.



```

Command Window

>> load test
Error using load
Unable to read file 'test'. No such file or directory.

>> x=[1 2]

x =

     1     2

>> format compact
>> u=[5 6]
u =

     5     6

>> x*u
Error using *
Inner matrix dimensions must agree.

>> ver

-----
MATLAB Version: 9.0.0.341360 (R2016a)
MATLAB License Number: ██████████
Operating System: Microsoft Windows 8.1 Enterprise Version 6.3 (Build 9600)
Java Version: Java 1.7.0_60-b19 with Oracle Corporation Java HotSpot(TM) 64-Bit Server VM mixed m
-----

MATLAB                               Version 9.0           (R2016a)
Simulink                               Version 8.7           (R2016a)
Aerospace Blockset                     Version 3.17          (R2016a)
Aerospace Toolbox                       Version 2.17          (R2016a)

```

**Bild 2.4** „Command Window“, die Arbeitsfläche, über die MATLAB-Befehle eingegeben werden können, mit ein paar exemplarisch eingegebenen, zum Teil falschen Befehlen, Ergebnissen und den entsprechenden Fehlermeldungen (normalerweise in rot), sowie die Demonstration des Befehls `ver` zum Anzeigen der MATLAB-Version und der installierten Toolboxes

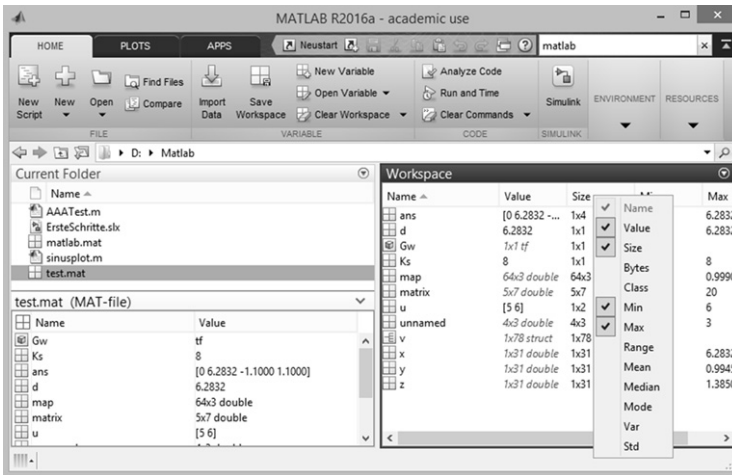
## Befehl `ver`

In Bild 2.4 ist auch noch ein weiterer sehr nützlicher MATLAB-Befehl zu sehen, der Befehl für „Version“. Nach der Eingabe von `ver` im „*Command Window*“ wird die genaue Bezeichnung der verwendeten MATLAB-Version, das Release-Jahr sowie a für die Frühjahrs- oder b für die Herbst-Ausgabe, das verwendete Betriebssystem, sowie vor allem die verfügbaren Toolboxes und deren Versionsnummern aufgelistet. Dies ist die schnellste Möglichkeit, um einen genauen Überblick über die installierte MATLAB-Version und die zur Verfügung stehenden Toolboxes zu bekommen.

## 2.2.2 „Current Directory“, das aktuelle Arbeitsverzeichnis

Links befindet sich standardmäßig eine Spalte des aktuellen Verzeichnisses, unter Windows normalerweise das Verzeichnis „*MATLAB*“ unter „*Eigene Dateien*“. Hier kann man gleich erkennen, ob bereits MATLAB-Dateien erzeugt wurden, die normalerweise an der Endung zu erkennen sind: `.m` für MATLAB-Programme, die Funktionen und Programme enthalten, oder `.mat` für abgespeicherte Variablen und Parameter, siehe Bild 2.5, linke Spalte.

Unterhalb des Fensters mit dem Arbeitsverzeichnis, „*Current Directory*“, kann sich noch ein weiteres Fenster mit Details zu den einzelnen `.mat`-Dateien verstecken, zu erkennen an dem



**Bild 2.5** Auf der linken Seite „*Current Directory*“, das aktuelle Arbeitsverzeichnis, mit den angezeigten Details der Datei `matlab.mat`. Auf der rechten Seite „*Workspace*“, der Arbeitsbereich mit den momentan verfügbaren Variablen. Bei Programmstart ist dieser Bereich leer. Mit rechter Maustaste in die Titelleiste von „*Workspace*“ können weitere Daten der Variable angezeigt werden, z. B. Größe, Min- und Max-Werte etc., siehe geöffnetes Kontextmenü. Für diesen Screenshot wurde das „*Command Window*“ „*undocked*“, d. h. aus dem MATLAB-Fenster gelöst, um nur die beiden genannten Fenster zeigen zu können

^-Zeichen rechts unten in der Ecke des „*Current Directory*“. Wenn eine `.mat`-Datei mit der Maus markiert ist und das ^-Zeichen wird mit der Maus angeklickt, öffnet sich das „*Details*“-Fenster und die in der `.mat`-Datei gespeicherten Variablen werden aufgelistet. Dies kann hilfreich sein, wenn man einige `.mat`-Dateien abgespeichert hat und versucht, eine bestimmte wiederzufinden.

### 2.2.3 „*Workspace*“, der Arbeitsbereich oder Arbeitsspeicher

Rechts oben ist der „*Workspace*“ abgebildet, der aktuelle Arbeitsbereich von MATLAB. Aufgelistet sind die im Moment im Arbeitsspeicher abgelegten Variablen und Parameter, siehe Bild 2.5, rechte Spalte.

Die Variablen sind zuerst nach großen und kleinen Anfangsbuchstaben, dann alphabetisch sortiert. Bei den Variablen wird also auf Groß- und Kleinschreibung geachtet. „*Value*“ gibt entweder den Wert der Variable an, z. B. `Ks = 8`, oder den Typ der Variable. `Gw` ist z. B. eine Übertragungsfunktion (`tf` steht dabei für engl. *transfer function*), `z` ist ein Vektor der Dimension  $1 \times 100000$  im Zahlenformat „*double*“.

Durch „*Drag & Drop*“ können Variablen vom „*Workspace*“ in das „*Command Window*“ gezogen und dort in Befehle eingebunden werden. Dies kann hilfreich sein, wenn man umständliche oder lange Variablennamen verwendet haben sollte, bei denen man sich bei der manuellen Eingabe über die Tastatur leicht vertippen könnte.

Wenn mit der rechten Maustaste in die Titelleiste von „*Current Folder*“ oder „*Workspace*“ geklickt wird, kann man die Liste nach unterschiedlichen Kriterien sortieren, oder wei-

tere Informationen zu den angezeigten Dateien im Arbeitsverzeichnis bzw. zu den Variablen im „*Workspace*“ in weiteren Spalten ausgeben lassen, siehe Bild 2.5, Kontextmenü zu „*Workspace*“, linke Seite.



Bei MATLAB ist unbedingt auf Groß- und Kleinschreibung von Variablenamen zu achten! Wie in der Sortierung in Bild 2.5 zu sehen ist, werden die Variablen unterschieden nach Groß- und Kleinbuchstaben sortiert. Zu lange, phantasievolle Variablenamen mit unterschiedlicher Groß- und Kleinschreibung können bei wiederholter Eingabe falsch geschrieben werden. Viele Fehlermeldungen resultieren aus unterschiedlichen Schreibweisen oder inkonsistenter Groß- und Kleinschreibung, deshalb unbedingt einfache, logische Variablenamen wählen!

## Variable Editor

Durch Doppelklicken mit der linken Maustaste auf eine der Variablen im „*Workspace*“ wird diese Variable in dem „*Variable Editor*“ geöffnet, siehe Bild 2.6.

Im „*Variable Editor*“ kann die Variable verändert werden, d. h., es können einzelne Zahlen der Matrix durch andere Werte ersetzt werden oder es können zusätzliche Spalten oder Zeilen eingefügt werden. Wenn nur eine einzelne Zahl außerhalb der bestehenden Spalten und Zeilen hinzugefügt wird, füllt MATLAB die fehlenden Zeichen der unvollständigen Spalten bzw. Zeilen der Matrix mit Nullen auf. Wird z. B. die folgende Matrix im „*Command Window*“ eingegeben:<sup>1</sup>

```
>> matrix=[1 2 3 4 5
6 7 8 9 0
11 12 13 14 15
16 17 18 19 20]
matrix =
     1     2     3     4     5
     6     7     8     9     0
    11    12    13    14    15
    16    17    18    19    20
```

Dann wird im „*Workspace*“ doppelt mit der Maus auf `matrix` geklickt, sodass sich der „*Variable Editor*“ oberhalb des „*Command Window*“ öffnet. Nun kann die Variable `matrix` z. B. um eine weitere Zeile und eine Spalte ergänzt werden, siehe Bild 2.6. Leere Felder werden automatisch mit Nullen aufgefüllt, sodass die Struktur erhalten bleibt.

Der „*Variable Editor*“ ist damit hilfreich, wenn Variablen überprüft, im Nachhinein korrigiert oder ergänzt werden müssen.

Der „*Variable Editor*“ kann auch über die Menüleiste geöffnet werden, über die Gruppe „*VARIABLE*“ → „*Open Variable*“. Mit „*New Variable*“ öffnet sich der „*Variable Editor*“ ebenfalls und in der Menüleiste wird der Tab „*VARIABLE*“ angezeigt. In der Gruppe „*SELECTION*“ wird

<sup>1</sup> Hinweis zur Wiedergabe von MATLAB-Befehlen und der Ausgabe der Ergebnisse im „*Command Window*“ in diesem Buch: Der eingegebene MATLAB-Befehl steht immer direkt hinter dem Eingabezeichen `>>`. Das Ergebnis oder die „Antwort“ von MATLAB auf den eingegebenen Befehl wird in diesem Buch eingerückt darunter abgebildet.



The screenshot shows the MATLAB Variable Editor interface. The main window displays a grid for the variable 'matrix', which is a 5x7 double matrix. The grid contains the following values:

|    | 1  | 2  | 3  | 4  | 5  | 6 | 7 | 8 |
|----|----|----|----|----|----|---|---|---|
| 1  | 1  | 2  | 3  | 4  | 5  | 0 | 0 |   |
| 2  | 6  | 7  | 8  | 9  | 0  | 0 | 0 |   |
| 3  | 11 | 12 | 13 | 14 | 15 | 0 | 0 |   |
| 4  | 16 | 17 | 18 | 19 | 20 | 0 | 0 |   |
| 5  | 1  | 2  | 3  | 4  | 5  | 6 | 7 |   |
| 6  |    |    |    |    |    |   |   |   |
| 7  |    |    |    |    |    |   |   |   |
| 8  |    |    |    |    |    |   |   |   |
| 9  |    |    |    |    |    |   |   |   |
| 10 |    |    |    |    |    |   |   |   |

The 'Command Window' shows the following command and output:

```
>> matrix=[1 2 3 4 5;
6 7 8 9 0;
11 12 13 14 15;
16 17 18 19 20];
matrix =
     1     2     3     4     5
     6     7     8     9     0
    11    12    13    14    15
    16    17    18    19    20
```

**Bild 2.6** „Variable Editor“ mit Inhalt der Matrix *matrix*, ursprünglich bestehend aus 5 Spalten und 4 Zeilen. Zu den bestehenden Zeilen wurde eine weitere Zeile hinzugefügt. Fehlende Werte in Spalte 6 und 7 wurden durch MATLAB mit Nullen ergänzt. Zu beachten ist auch, dass ein neuer Tab „VARIABLE“ in der Menüleiste erschienen ist, in dem die Optionen zum Bearbeiten der Variablen aufgezogen werden

angezeigt, in welcher Zeile und Spalte der Cursor steht. Die letzte Zeile bzw. Spalte einer Variablen wird mit „end“ bezeichnet. In der Gruppe „EDIT“ können Zeilen oder Spalten eingefügt oder gelöscht werden, eine interessante Option ist auch das Transponieren („Transpose“), d. h. Zeilen und Spalten werden vertauscht. In der Gruppe „VARIABLE“ finden sich die Befehle zum Öffnen oder Ausdrucken einer bestehenden Variablen bzw. eine neue Variable kann erstellt werden.

### Speichern von Variablen mit dem Befehl save

Alle im „Workspace“ angezeigten Variablen und Parameter sind allerdings nur temporär gespeichert. Sobald MATLAB beendet wird, sind alle Werte verloren, wenn sie nicht in einer .mat-Datei gespeichert wurden.

Am schnellsten geht das Speichern der Variablen über die Menüleiste. Unter „File“ → „Save Workspace As...“, bzw. <Strg> + <S>, können die Variablen in einer selbst zu benennenden Datei abgespeichert werden, die automatisch die Endung .mat erhält

Über das „Command Window“ können die Variablen auch mit dem Befehl `save Dateiname` gesichert werden. Der Befehl `save` bietet verschiedene Auswahlmöglichkeiten zum Abspeichern von Variablen:

```
>> save Ks
```

Wenn eine Variable mit der Bezeichnung `Ks` existiert, wird mit `save Ks` genau diese eine Variable `Ks` in einer `.mat`-Datei mit der Bezeichnung `Ks.mat` im aktuellen Arbeitsverzeichnis („*Current Directory*“) abgespeichert.

```
>> save Test Ks u x z
```

Mehrere ausgewählte Variablen können gespeichert werden, wenn diese durch Leerzeichen getrennt hinter einem Dateinamen aufgelistet werden. `Test` ist in diesem Fall der Dateiname der erzeugten `.mat`-Datei. `Ks`, `u`, `x` und `z` sind die ausgewählten Variablen. Der Dateiname sollte allerdings nicht identisch sein mit einem bereits verwendeten Variablennamen.

```
>> save Ks u x z
```

Mit diesem Befehl wird eine Datei mit der Bezeichnung `Ks.mat` erzeugt, die allerdings nur die Variablen `u`, `x` und `z` enthält, nicht aber `Ks`, da `Ks` als Dateiname verwendet wurde.

Diese Variante des Sicherns von Variablen ist nützlich, wenn man viel ausprobiert und dabei auch viele unnütze Variablen erzeugt hat, die nicht erhalten werden sollen. Sollen alle Variablen gespeichert werden, ist der folgende Befehl einfacher:

```
>> save Test
```

Wird hinter dem `save`-Befehl nur der Dateiname der `.mat`-Datei eingegeben, werden alle Variablen des MATLAB-Arbeitsbereichs („*Workspace*“) gespeichert. Aber Achtung, der Dateiname darf noch nicht als Variablenname verwendet worden sein!

### Laden von gespeicherten Variablen mit dem Befehl `load`

```
>> load test
```

Die mit dem `save`-Befehl gespeicherten Variablen können mit `load Dateiname` wieder in den Arbeitsbereich geladen werden, z. B. wenn man MATLAB wieder neu gestartet hat und die Sitzung vom vorherigen Mal wieder dort weiterführen möchte, wo man das letzte Mal aufgehört hat.

Das Laden von gespeicherten Variablen geht natürlich auch über die Menüleiste. Unter „*File*“ → „*Open*“ kann eine beliebige, früher abgespeicherte `.mat`-Datei aufgerufen werden.

Noch schneller geht es per Doppelklick mit der linken Maustaste im Fenster „*Current Directory*“ auf eine der angezeigten `.mat`-Dateien.



Während bei den Variablen auf Groß- und Kleinschreibung geachtet werden muss, spielt dies bei den Dateinamen unter Microsoft Windows keine Rolle!

### Löschen von Variablen mit dem Befehl `clear`

MATLAB muss selbstverständlich nicht neu gestartet werden, wenn man alle Variablen im Arbeitsspeicher loswerden möchte. Mit dem `clear`-Befehl lassen sich einzelne oder alle Variablen im Arbeitsbereich löschen.

```
>> clear
```

```
>> clear all
```

Mit `clear` oder `clear all` werden alle Variablen im Arbeitsspeicher gelöscht. Mit diesem Befehl ist also vorsichtig umzugehen, bevor alle Variablen plötzlich unerwünscht weg sind.

# Index

2D-Grafik 119, 121

3D-Grafik 130

3D-Objekt 135

## A

abs 74

Absolutwert 74

Abszisse 101

Achsen

– Achsenbegrenzung 109, 139

– Achsenbeschriftung 105, 110

– Achsenskalierung 105

acos 77

– acosd 77

– acosh 77

Addition 71

– Unäre Addition 71

Add-Ons 34

Add-Ons Explorer 35

Aktivierung 19

all 79

Alles auswählen 267

Amplitude

– Amplitudengang 194, 211, 217

– Amplitudenrand 211, 217, 220

– Amplitudenverstärkung 194

– Amplitudenwerte 198, 199, 211

Analyze Code 33

and 79

angle 74

ans 52

any 79

APPS 35

Arbeitsspeicher 25

Arbeitsverzeichnis 24

area 123

Argument 194

Array multiply 97

arrow 110

asin 77

– asind 77

– asinh 77

atan 77

– atand 77

– atanh 77

attributes 177

Ausgangsblöcke 289

Ausschneiden 117, 267

Axes Properties 109

axis 105, 139

## B

Balkendiagramme 122, 134, 141

bar 122, 141

– barh 122

bar3 135, 141

– bar3h 135

base2dec 49

base2dec(str, basis) 49

Basic Fitting 111, 113, 114

Befehlsfenster 23

Beobachtungspunkt 139

Betrag 74, 195, 217

Betragskennlinien 194

bin2dec 49

bin2dec(str) 49

bitand 80

bitcmp 80

bitget 80

bitor 80

bitset 80

bitshift 80

Bitweise Operatoren 80

bitxor 80

Blank M-File 154

Blockschaltbild 182

bode 194  
Bode-Diagramm 188, 194, 195, 210, 211,  
215, 216, 221, 222, 226  
Boole'sche Algebra 79  
box 140  
break 166  
Brush 111

**C**

Camera Toolbar 109, 113  
Campuslizenz 18  
ceil 51, 75  
cell 50  
char 50  
Charakteristika einer Übertragungsfunktion  
202  
clabel 121  
classdef 177  
clc 170  
clear 28, 170  
clear all 170  
Clear Commands 33  
Clear Workspace 33  
Clock (Block) 288  
Close 22  
Color 117  
colormap 103  
ColorOrder 104  
comet 124  
comet3 137  
Command History 29  
Command Window 23  
Community 34  
Compare 32  
compass 125  
coneplot 138  
Configuration Defaults 260  
Configuration Parameters 297  
conj 75  
Constant (Block) 288  
continue 167  
contour 121, 141  
contour3 133, 141  
contourf 123  
contourslice 133

Control System Toolbox 179  
conv 181  
convolution 181  
Copy 117, 267  
– Copy Figure 109  
– Copy Model To Clipboard 267  
– Copy Options 109  
corrcoef 68  
cos 77  
– cosd 77  
– cosh 77  
cov 68  
Cramer'sche Regel 86  
Current Directory 24  
Cut 117, 267  
cylinder 135

**D**

Dämpfung 228  
daspect 140  
data aspect ratio 140  
Data Cursor 110  
Data Import/Export 263  
Data Statistics 111, 113, 114  
dB 194, 212, 217  
dec2base 49  
dec2base(a,basis) 49  
dec2bin 49  
dec2bin(a) 49  
dec2hex 49  
dec2hex(a) 49  
delay time 208  
Delete 117, 267  
Demux (Block) 291  
denominator 180  
Derivative (Block) 289  
Desktop 21, 111  
det 86  
Determinante 86  
diag 64, 96  
Diagnostics 264  
Diagramme 101  
– Diagrammeigenschaften 109  
– Diagrammtitel 105, 110, 211  
– Diagrammtyp 120

diary 170  
diff 95  
Differenz zwischen Elementen einer Matrix 95  
Dirac-Impuls 189  
disp 171  
Display (Block) 289  
Division 72  
Dock 22  
Documentation 37  
Doppelpfeil 110  
double 50  
double arrow 110  
Drahtgitternetze 134  
Drehen 110  
dreidimensionale Grafiktypen 130, 133  
dreidimensionale Objekte 135  
Drucken 265  
Dynamisches System 194

## E

echo 170  
Edit 109, 267  
Edit Plot 110  
Editiermodus 110  
Editor 144  
eig 62  
Eigenschaften einer Grafik 105  
Eigenschaften einer Übertragungsfunktion 203, 207  
Eigenvektoren 62  
Eigenwert 62  
Einfügen 267  
Eingabe des Benutzers 171  
Eingangsblöcke 288  
Eingangsgröße 296  
Einheitsmatrix 58  
Einschwingzeit 189  
Elementweise Verknüpfung 97  
– Elementweise Division 98  
– Elementweise Multiplikation 97  
– Elementweises Potenzieren 99  
Ellipse 110  
ellipsoid 136  
ENTWEDER-ODER 79

eq 78  
error 264  
errorbar 122  
events 177  
exp 74  
explode 135  
Exponentialfunktion 74  
eye 58  
ez 120  
ezcontour 122  
ezcontourf 123  
ezmesh 134  
ezplot 120, 121  
ezplot3 133  
ezpolar 125  
ezsurf 137

## F

Faktorierte Form 186  
Faltung 181  
Farben  
– Farbenwerte 106  
– Farbpalette 103  
– Farbschattierungen 140  
– Farbtafel 117  
– Farbübergänge 140  
feather 124  
feedback 187  
Fehlerbalkendiagramme 122  
Fehlermeldung 264  
Feldoperationen 97  
Feste Schrittweite 261  
figure 102, 189  
Figure Properties 103, 108, 109  
File 108  
fill 123  
fill3 135  
Find 267  
Find Files 32  
fix 51  
fixed-step 261  
Flächendiagramme 123, 135  
flintmax 80  
float 49  
Floating Scope (Block) 289

floor 51  
 format 47  
 format + 47  
 format bank 47  
 format compact 48  
 format hex 47  
 format long 47  
 format longe 47  
 format longeng 47  
 format longg 47  
 format loose 48  
 format rat 47  
 format short 47  
 format shorte 47  
 format shorteng 47  
 format shortg 47  
 for-Schleife 157, 166  
 frequency response 204  
 Frequenzantwort 194, 204  
 Frequenzbereich 197  
 Frequenzgang 194  
 Frequenzliniendiagramm 194  
 From File (Block) 288  
 From Workspace (Block) 288  
 Führungssprungantwort 194, 219, 223, 227  
 Führungsübertragungsfunktion 187  
 function 174  
 Function (Block) 292  
 Function Browser 41  
 Function M-File 174  
 Funktion 174  
 Funktionsgleichung einer Kurve 111

## G

gain 198  
 Gain (Block) 290  
 gain margin 211  
 Gauß'scher Algorithmus 93  
 ge 78  
 Generate Code 108  
 Geschlossener Regelkreis 187  
 get 203  
 Gewichtsfunktion 189  
 Gitternetzlinien 105  
 Gleich 78

Grafiken 101  
 – Grafikbefehle 190  
 – Grafikeigenschaften 103, 109, 190  
 – Grafikfenster 189  
 – Grafiktitel 105  
 – Grafiktypen 119, 130  
 grid 105, 190, 198, 200, 201  
 Größe eines Vektors 64  
 Größer als... 78  
 Größer als oder gleich... 78  
 Ground (Block) 288  
 Grundrechenarten 70  
 gt 78  
 gtext 105

## H

Harmonische Schwingung 219  
 Help 111  
 help 43  
 hex2dec 49  
 hex2dec(str) 49  
 hex2num 49  
 hex2num(str) 49  
 Hilfe 37  
 Histogramme 122  
 Höhenprofile 136  
 hold 118  
 – hold off 102  
 – hold on 102  
 HOME 32  
 HSV-Farbraum 104

## I

if-elseif-else-Verzweigung 161  
 imag 75  
 image 123  
 imagesc 123  
 Imaginärwert 75, 198  
 Import Data 32, 108  
 Impulsantwort 188, 189  
 impulse 189, 191  
 imread 123  
 Inf 46  
 input 171

InputDelay 207  
 Insert 110  
 Installation 18  
 int8 50  
 int16 50  
 int32 50  
 int64 50  
 Integrator (Block) 289  
 interp 137  
 intersect 81  
 intmax 80  
 inv 61  
 Inverse Matrix 61, 85  
 Invertieren 61, 84  
 ismember 81  
 Isolinien 121, 133

## K

Kartesisches Koordinatensystem 196  
 Kegel 138  
 Kettenschaltung 182  
 Kleiner als... 78  
 Kleiner als oder gleich... 78  
 Kommentare 23, 155  
 Kompensation 186  
 Komplexe Zahlen 47  
 Komplexe Zahlenebene 196  
 Konjugiert Komplexe 75, 186  
 Kontrollstrukturen 156  
 Koordinaten 110  
 Kopfzeile einer Funktion 175  
 Kopieren 117  
 Korrelationskoeffizient 68  
 Kosinus 77  
 Kovarianz 68  
 Kritischer Betragswert 194  
 Kritischer Phasenwert 194  
 Kritischer Punkt 196, 215, 217  
 Kuchendiagramme 123, 135  
 Kugelobjekte 136  
 Kurveigenschaften 109

## L

Laden 28

Länge eines Vektors 65  
 Laplace-Variable 185  
 lasterr 166  
 Laufzeit 208  
 Layout 33  
 ldivide 72, 99  
 le 78  
 legend 105, 110, 190, 306  
 Legende 105, 110  
 length 65  
 Lichtquelle 137  
 line 110, 121  
 Line Style 117  
 Line Width 117  
 Lineares Gleichungssystem 86  
 LineWidth 107  
 Linien 110  
 – Linienart 117  
 – Liniendicke 107, 117  
 – Linientypen 106  
 Liniendiagramme 121, 133, 141  
 Link 111  
 Linke Matrixdivision 93  
 Linke-Hand-Regel 196  
 linspace 54  
 listeners 177  
 load 28  
 log 74  
 log10 74  
 Logarithmus 74  
 logical 50  
 Logische Operatoren 78  
 loglog 121  
 logspace 54, 195  
 Löschen 117  
 lt 78  
 LTI Systems 294  
 LTI Systems (Block) 291

## M

magic 63  
 Magisches Quadrat 63  
 magnitude 195  
 Manual Switch (Block) 291, 301  
 margin 211, 212, 217, 218, 220–222, 224, 226

Marker 107, 117  
– Marker Size 117  
– MarkerEdgeColor 107  
– MarkerFaceColor 107  
– Markerfüllungen 107  
– Markergröße 107, 117  
– MarkerSize 107  
– Markerumrandungen 107  
Maschennetzdiagramme 134  
Math Operations 290  
Mathematische Funktionen 74  
Mathematische Verknüpfungen 290  
MATLAB Function (Block) 292  
MATLAB-Programm 222  
Matrix multiply 98  
Matrixdivision 72  
Matrixmultiplikation 72, 88  
Matrizen 55  
max 65  
Maximalwerte 65  
Maximize 21  
mean 67  
median 67  
Mengenoperatoren 81  
mesh 134  
meshc 134  
meshz 134  
methods 177  
min 65  
Minimalwerte 65  
Minimize 21  
Minimum Stability Margins 197  
minus 71  
Mittelwert 67  
mldivide 72, 93  
mod 76  
Model  
– Model Properties 259  
Modulo 76  
mpower 73  
mrdivide 72, 94  
mtimes 72, 98  
Multiplikation 72  
Multiport Switch (Block) 291  
Mux (Block) 291, 295

**N**

ne 78  
Nenner 180  
nested functions 176  
New 32  
New Script 32  
New Variable 32  
nichols 198  
Nichols-Ortskurve 188, 198, 199  
NICHT 79  
not 79  
Nullmatrix 58  
Nullstellen 203  
num2hex 49  
num2hex(a) 49  
numerator 180  
nyquist 196  
Nyquist-Ortskurve 188, 196, 215, 216  
Nyquist-Stabilitätskriterium 196, 197

**O**

Oberflächendiagramme 136  
object properties 203  
Object-oriented Programming 177  
objects 177  
Objektklasse 177  
Objektklassendefinition 177  
Objektorientiertes Programmieren 144, 177  
ODER 79  
ones 59  
Open 32  
Open Variable 33  
Optimization 263  
or 79  
Ordinate 101  
Ortskurve 196  
OutputDelay 207

**P**

Package App 35  
packages 177  
Pan 110  
Parallel 34  
parallel 183



- Parallelschaltung 183  
 pareto 122  
 Passcode 19  
 Paste 267  
 patch 135  
 pcolor 124  
 Peak Response 189, 197  
 peaks 121, 136  
 Pfeil 110  
 Pfeildiagramme 124  
 Phase  
 – Phasengang 211  
 – Phasenkenlinien 194  
 – Phasenrand 211, 217, 220  
 – Phasenreserve 217  
 – Phasenverschiebung 194, 195  
 – Phasenwert 198, 199, 211  
 – Phasenwinkel 74  
 phase 198  
 phase margin 211  
 PID-Regler 184, 225, 226  
 pie 123  
 pie3 135  
 PI-Regler 184, 214  
 plot 101, 121, 141, 306  
 Plot Browser 109, 113  
 Plot Edit Toolbar 109  
 Plot Selector 140  
 plot3 133, 141  
 plotmatrix 126  
 PLOTS 34  
 plotyy 121  
 plus 71  
 Pol- und Nullstellendiagramm 199  
 polar 125  
 Polargitternetzlinien 125  
 Polarkoordinaten 125, 196  
 pole 202  
 Pole einer Übertragungsfunktion 202  
 Polform 181, 186  
 Polstellen 203  
 Polygone 123, 134  
 Polygonnetz 134  
 Polygonnetzdiagramme 134  
 Polynom 181  
 Polynommultiplikation 181  
 Potenzieren 73  
 Potenzieren einer Matrix 92  
 power 73  
 Preferences 34, 109  
 prod 95  
 Produkt der Elemente einer Matrix 95  
 Programmieren 144  
 Properties 105, 177  
 Property Editor 109, 113, 117, 126, 193, 197,  
 198, 217  
 $PT_1$ -Glieder 180  
 $PT_2$ -Glieder 215  
 Punkttypen 106  
 pzmap 199
- Q**
- Quadratwurzel 74  
 Quick Access Toolbar 37  
 quiver 124  
 quiver3 137
- R**
- Radialdiagramme 125  
 rand 59  
 – randi 61  
 – randn 60  
 randi 59  
 Rang einer Matrix 85  
 rank 85  
 rdivide 72, 99  
 real 75  
 Realwert 75, 198  
 Rechte Matrixdivision 94  
 Rechteck 110  
 rectangle 110  
 Redo 267  
 Regelgröße 187  
 Regelkreis 180, 213  
 Regelstrecke 180, 213  
 Regelungstechnik 179  
 Regelverhalten 220  
 Regler 180  
 Reglerentwurf 213, 229, 230  
 Regleroptimierung 224  
 Reihen-, Serien- oder Kettenschaltung 182

Reihenschaltung 182, 216  
 Relationale Operatoren 77  
 rem 76  
 Remainder 76  
 Request Support 34  
 Reset View 111  
 Restore 22  
 return 167  
 RGB-Farbraum 104  
 Richtungsdiagramme 124, 137  
 rlocfind 228  
 rlocus 201, 228  
 root locus 201  
 rose 125  
 Rotate 3D 110  
 Rotieren 110  
 round 51, 75  
 Rückführung 187, 188  
 Rückgängigmachen 267  
 Run and Time 33  
 Runden 51

## S

Save  
 – Save As... 108, 259  
 – Save Workspace As... 27, 109  
 save 27  
 Save Workspace 32  
 scatter 125  
 scatter3 138  
 Schachbrettmuster-Diagramme 124  
 Schleifen 156  
 Schnittfläche 137  
 Schwingen 215  
 Scope 295  
 Scope (Block) 289  
 Seitenverhältnis 140  
 Select All 267  
 semilogx 121  
 semilogy 121  
 Serienschaltung 182  
 series 182  
 set 197, 198, 200, 201, 207  
 Set Path 22, 34  
 setdiff 81  
 Set-Operatoren 81  
 Settling Time 189  
 setxor 81  
 shading 137, 140  
 Shortcuts 31  
 Show Code 117  
 Show Property Editor 117  
 sign 74  
 Sign (Block) 290  
 Signal Generator 296  
 Signal Generator (Block) 288  
 Signal Routing 291  
 Signalflussplan-Algebra 182  
 Signalführung 291  
 signum 74  
 Simulation Parameters 297  
 SIMULINK 252  
 – Simulink Preferences 260  
 – Simulink Start Page 252, 253  
 simulink 252  
 sin 77  
 – sind 77  
 – sinh 77  
 single 50  
*single input single output* 213  
 Sinks 289  
 Sinus 77  
 SISO-System 213  
 size 64  
 slice plane 137  
 Slider Gain (Block) 290, 294  
 Solver 260, 261  
 Sources 288  
 Spaltenvektoren 53  
 Speichern 259  
 sphere 136  
 Sprungantwort 188, 191, 192, 210, 215, 216,  
 224  
 spy 126  
 sqrt 74  
 stabil 196  
 Stabiles System 215  
 Stabilität 214  
 – Stabilitätsgrenze 197, 219  
 – Stabilitätsgüte 217, 220  
 – Stabilitätsverhalten 228  
 stairs 121

Standardabweichung 68  
 state space 204  
 Statistik 66, 111  
 std 68  
 steam 123  
 stem 123  
 stem3 134  
 step 191  
 Step (Block) 288, 302  
 Stochastik 66  
 Stop Time 302  
 streamline 138, 141  
 streamribbon 138, 141  
 streamslice 137  
 streamtube 139, 141  
 Streudiagramme 125  
 Strömungsdiagramme 141  
 Strömungslinien-Diagramme 137, 138  
 Strömungsröhren-Diagramme 139  
 struct 50  
 Stufendiagramme 121  
 subclasses 177  
 subfunctions 176  
 subplot 118, 202, 215  
 Subsystem (Block) 291  
 Subtract (Block) 290  
 Subtraktion 71  
 – Unäre Subtraktion 71  
 Suchen 267  
 sum 64, 95  
 Sum (Block) 290  
 Summe der Elemente einer Matrix 95  
 superclasses 177  
 surf 136  
 surface 136  
 surfc 137  
 surf1 137  
 switch-case-otherwise-Verzweigung 163  
 System 180

## T

tan 77  
 – tand 77  
 – tanh 77  
 Tangens 77

Terminator (Block) 289  
 text arrow 110  
 text box 110  
 Textfeld 110  
 Textpfeil 110  
 tf 180  
 times 72, 98  
 title 105, 110, 190, 211, 306  
 To File (Block) 289  
 To Workspace (Block) 289  
 Tools 110  
 Toolstrip 31  
 Totzeit 208  
 transfer function 180  
 Transfer Function (Block) 290  
 Transponieren 83  
 Transportzeit 208  
 Trigger (Block) 292  
 Trigonometrische Funktionen 76  
 tril 97  
 triu 96  
 try-catch-Fehlerkontrolle 165  
 tzero 203

## U

Übergangsfunktion 191  
 Übergangszeit 228  
 Überschwingweite 228  
 Übertragungsfunktion 180  
 Übertragungsverhalten 194  
 uint8 50  
 uint16 50  
 uint32 50  
 uint64 50  
 uminus 71  
 UND 79  
 Undo 267  
 Undock 22  
 Ungleich 78  
 union 81  
 unique 81  
 Unterdiagramme 118  
 Unterfunktionen 176  
 Unterklassen 177  
 uplus 71

**V**

Value 25  
var 67  
Variable Editor 26  
Variable Schrittweite 261  
variable step 261  
Variablenamen 26  
Varianz 67  
Vektoraddition 83  
Vektorendiagramme 124  
Vektor-Matrix-Produkt 92  
Vektorsubtraktion 83  
ver 24  
Verschachtelte Funktionen 176  
Verschieben der Kurve 110  
Verstärkungsfaktor 203, 216, 220, 228  
Verstärkungsrand 217  
Verzugszeit 208  
View 109  
view 139  
Virtuelle Kamera 109  
Volumetrische Diagramme 138  
Vorzeichen 74

**W**

warning 264  
Warnmeldung 264  
Wasserfall-Diagramme 133  
waterfall 133  
while-Schleife 159, 166  
who 29  
whos 29

Wiederherstellen 267  
Window 111  
Winkelhalbierende 229  
WOK 201  
Workspace 25  
Wurzelortskurve 188, 201, 215, 216, 228, 230

**X**

xlabel 105, 110  
xor 79

**Y**

ylabel 105, 110

**Z**

Zahlen 46  
Zahlenklassen 49  
Zähler 180  
Zeilenvektoren 53  
zero 203  
Zero-Pole-Gain 186  
zeros 58  
Zoom 110  
zpk 186  
Zufallswerte 59  
Zufallszahlen 59  
Zurücksetzen der Ansicht 111  
Zusammenschaltung von Modellen 182  
Zustandsgleichung 204  
zweidimensionale Grafiktypen 119, 121  
Zylinderobjekte 135