3
Matrizen

Jörn Loviscach

Versionsstand: 20. März 2012, 16:02

Die nummerierten Felder sind absichtlich leer, zum Ausfüllen in der Vorlesung.
Videos dazu: http://www.j3L7h.de/videos.html

1 Matrix

Ein rechteckige Anordnung von mathematischen Objekten (typischerweise Zahlen) heißt Matrix (Mehrzahl: Matrizen) [matrix, matrices]. Das Wort stammt über Umwege vom latenischen Wort für Mutter (mater) ab. In Europa schreibt man runde Klammern um die rechteckige Anordnung:

$$\begin{pmatrix} 12 & -23 & \pi \\ 5 & -\sqrt{2} & 98 \end{pmatrix}$$

In den USA verwendet man statt dessen eckige Klammern, was sich in der Software niederschlägt (mit Ausnahmen wie Wolfram Alpha):

$$\begin{bmatrix} 12 & -23 & \pi \\ 5 & -\sqrt{2} & 98 \end{bmatrix}$$

Diese Matrix hier ist eine mit zwei Zeilen [rows] und drei Spalten [columns], also eine 2×3-Matrix.

Ein Spaltenvektor mit m Komponenten ist eine $m \times 1$-Matrix. Ein Zeilenvektor mit n Komponenten ist eine $1 \times n$-Matrix. Eine „quadratische“ Matrix [square matrix] hat genau so viele Spalten, wie sie Zeilen hat.

Matrizen werden meist mit großen Buchstaben bezeichnet, ihre Einträge [entry, entries] mit dem entsprechenden Kleinbuchstaben mit Indizes [index, indices] für Spalte und Zeile:

$$A = \begin{vmatrix} \overset{1}{} \\ \\ \\ \end{vmatrix} .$$

Dies entspricht auch der Art `A(1,2)`, wie man zweidimensionale Arrays in MATLAB® und Co. anspricht. In den C-Programmiersprachen ist es auch so, nur dass man dort mit der Zeile 0 und der Spalte 0 anfängt, also die Matrix als `double A[2][3];` definiert und mit `A[0][1] = -23.0;` auf einen Eintrag zugreift.

Um eine Matrix einzugeben, mischt man in MATLAB® und Co. die Schreibweisen für Zeilen (Leerzeichen – falls eindeutig – oder Komma) und Spalten (Semikolon oder Zeilenumbruch):

```
A = [12 -23 pi; 5 -sqrt(2) 98]
```

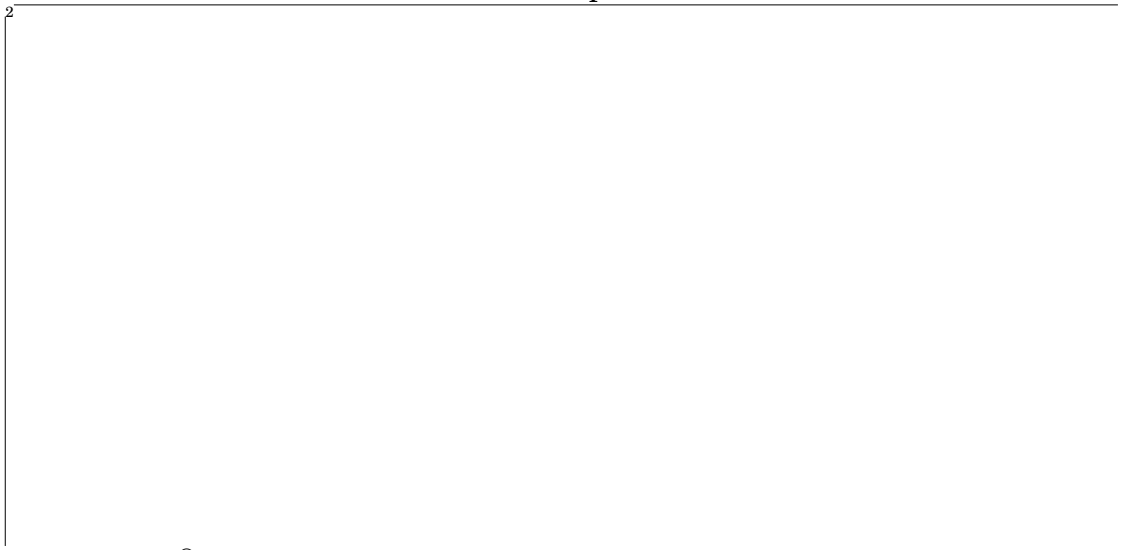
In Wolfram Alpha sind Schweifklammern angesagt:

```
{{12, -23, pi}, {5, -sqrt(2), 98}}
```

Matrizen beschreiben (einige) Transformationen zwischen verschiedenen Koordinatensystemen, können benutzt werden, um lineare Gleichungssysteme kompakt aufzuschreiben und treten auch direkt als physikalische Größen auf, zum Beispiel als Trägheitstensor. Oft hilft es, sich die Wirkung von Matrizen als geometrische Umformungen (Transformationen) vorzustellen. Dazu einige Beispiele am Ende.

2 Matrixoperationen

Matrizen gleicher Größe (Zahl der Zeilen, Zahl der Spalten) lassen sich addieren, subtrahieren und mit reellen Zahlen multiplizieren:



In MATLAB® und Co. ebenso wie in Wolfram Alpha werden diese Operationen ohne weitere Überraschungen mit $+$, $-$ und $\star$ geschrieben.

Das neutrale Element der Matrizenaddition ist die Matrix der passenden Größe, in der nur Nullen stehen. In MATLAB® und Co. erhält man die 2×3-Nullmatrix mit `zeros(2,3)`.

Die Menge aller Matrizen aus m Zeilen und n Spalten reeller Zahlen bildet einen Vektorraum über den reellen Zahlen. Dieser hat die Dimension

[3]

.

Die einfachste Operation mit Matrizen ist das „Stürzen“, die Transposition, geschrieben als ein hochgestelltes T:

[4]

In Büchern stehen Spaltenvektoren oft als transponierte [transposed] Zeilenvektoren, um Platz zu sparen:

[5]

In MATLAB® und Co. schreibt man `A'` für das (komplex konjugierte, aber das ändert ja bei reellen Zahlen nichts) Transponierte der Matrix oder des Vektors `A`, in Wolfram Alpha setzt man ein `Transpose[...]` herum.

Eine „symmetrische“ Matrix A ist gleich ihrer Transponierten: $A = A^T$. Symmetrische Matrizen müssen quadratisch sein. Beispiel:

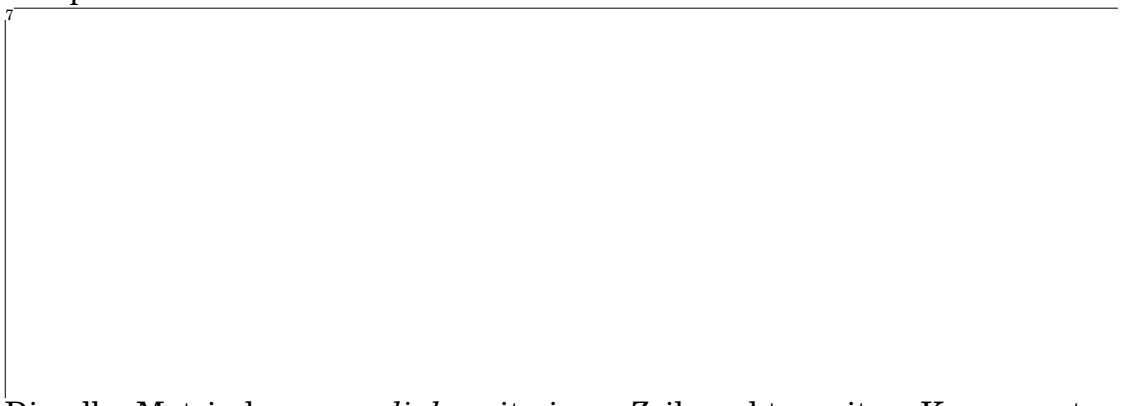
[6]

Zum Beispiel der Trägheitstensor wird durch eine symmetrische Matrix dargestellt.

3 Matrixprodukte

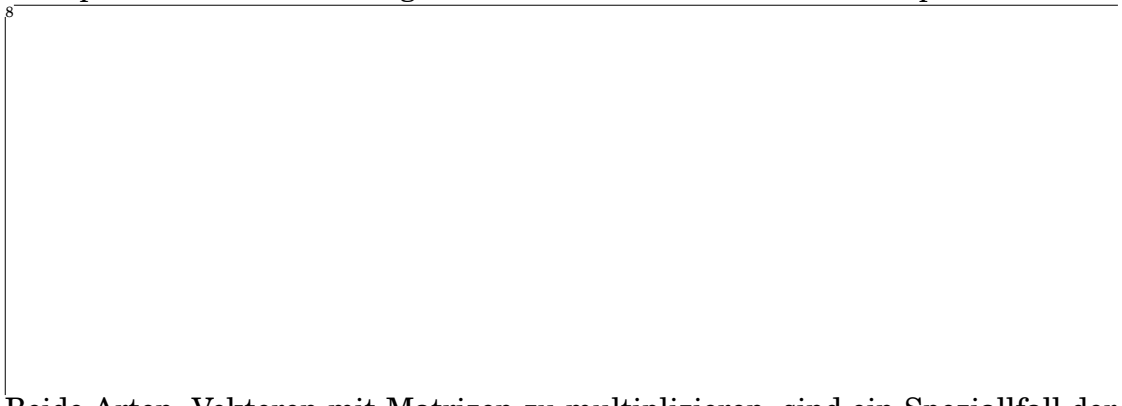
Die wesentliche Operation mit Matrizen ist die Multiplikation einer Matrix mit einem Vektor. Eine $m \times n$-Matrix kann *von rechts* mit einem Spaltenvektor mit n Komponenten multipliziert werden. Das Ergebnis ist ein Spaltenvektor mit m Komponenten:

[7]



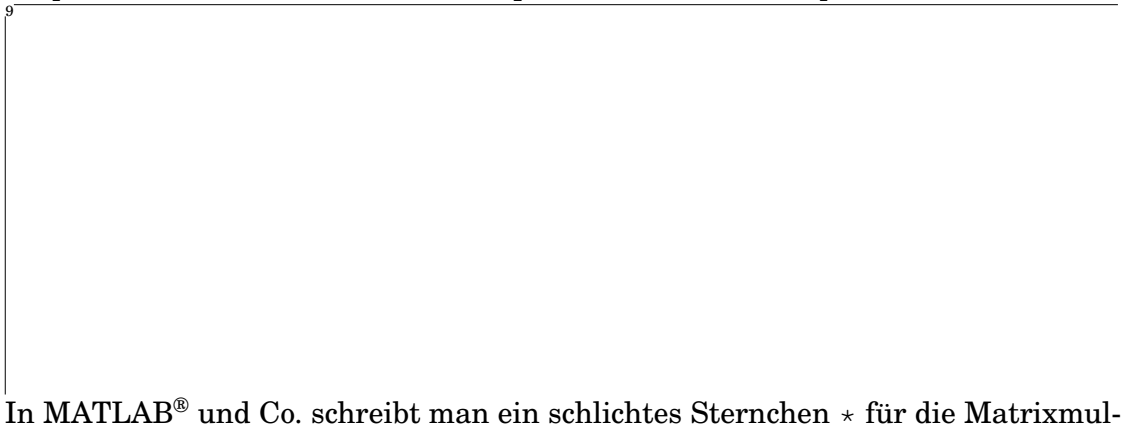
Dieselbe Matrix kann *von links* mit einem Zeilenvektor mit m Komponenten multipliziert werden. Das Ergebnis ist ein Zeilenvektor mit n Komponenten:

[8]



Beide Arten, Vektoren mit Matrizen zu multiplizieren, sind ein Speziallfall der Multiplikation zweier Matrizen miteinander. Eine $m \times n$-Matrix links und eine $n \times p$-Matrix rechts werden bei Multiplikation zu einer $m \times p$-Matrix:

[9]

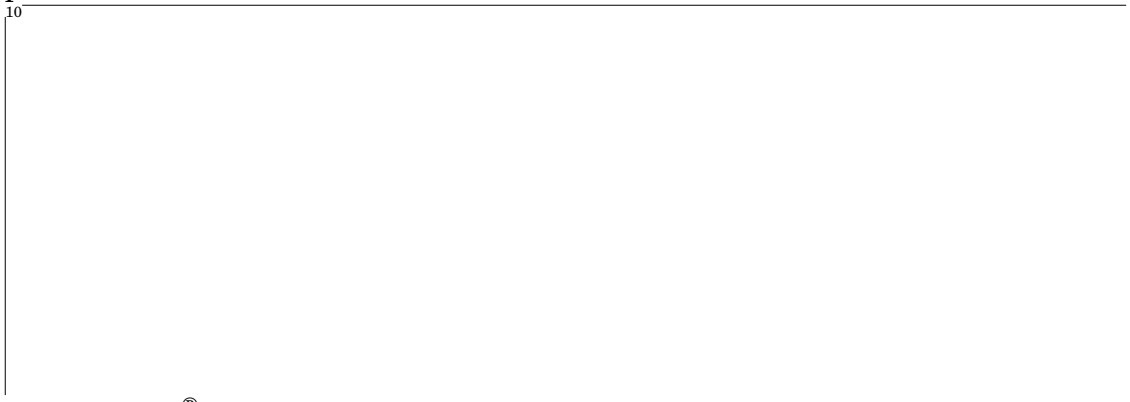


In MATLAB® und Co. schreibt man ein schlichtes Sternchen $\star$ für die Matrixmultiplikation, in Wolfram Alpha `.`, also einen ganz normalen Punkt.

Die Multiplikation von Matrizen mit Matrizen ist – wenn die Matrizen von der Zahl der Spalten/Zeilen her überhaupt multiplizierbar sind – assoziativ und distributiv, aber *nicht* kommutativ (→ Seminar, Praktikum). Das neutrale Element der Matrixmultiplikation ist die Einheitsmatrix [identity matrix] $\mathbf{1}$ oder $\mathbf{E}$ in der

passenden Größe:

[10]



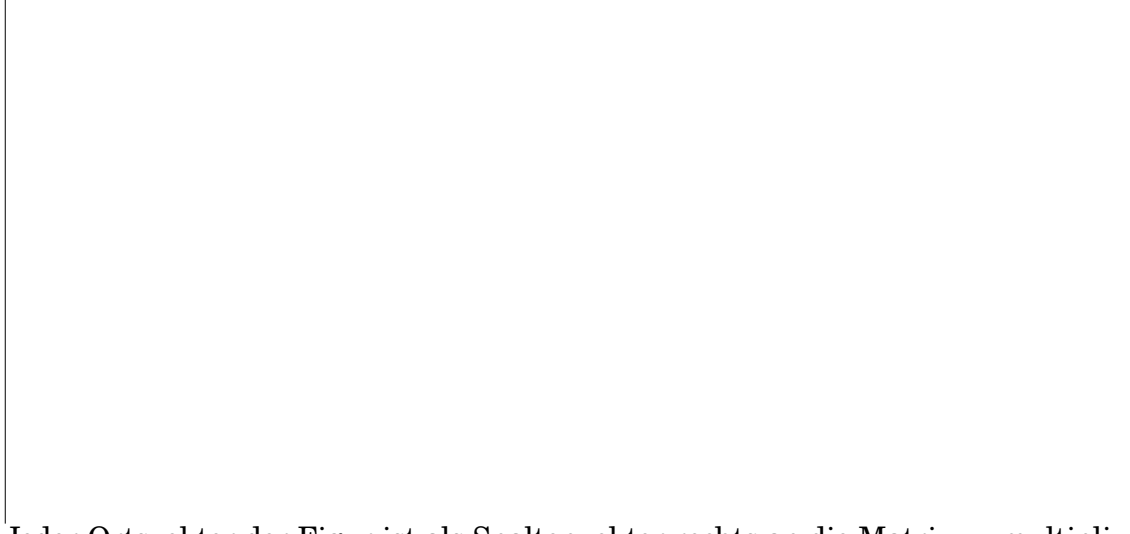
In MATLAB® und Co. erhält man zum Beispiel die 3×3-Einheitsmatrix mit `eye(3)`, in Wolfram Alpha mit `IdentityMatrix[3]`.

Eine quadratische Matrix A kann man auch mit sich selbst multiplizieren. (Warum muss sie dazu quadratisch sein?) Damit kann man quadratische Matrizen in positive ganze Potenzen setzen: $A^4 := AAAA$. Später befassen wir uns mit A^{-1}, der inversen Matrix von A. Sinnvollerweise (Warum?) wird damit gelten: $A^{-1}A = \mathbf{1} = AA^{-1}$. Unter bestimmten Voraussetzungen kann man auch $A^{1/2}$ und Ähnliches definieren. In MATLAB® und Co. ebenso wie in Wolfram Alpha verwendet man den Operator ^ zum Potenzieren von Matrizen, wie bei Zahlen.

4 Skalierungen

Die einfachste geometrische Operation mit Matrizen ist wohl die Skalierung aus dem Ursprung heraus. Zum Beispiel kann man alle geometrische Figuren um den Faktor 3 entlang der x-Achse und um den Faktor 2 entlang der y-Achse dehnen:

[11]



Jeder Ortsvektor der Figur ist als Spaltenvektor rechts an die Matrix zu multiplizieren. Das Ergebnis ist die transformierte Figur. (Was passiert, wenn man einen oder beide der Skalierungsfaktoren negativ wählt?)

Um diese und andere Transformationen in MATLAB® und Co. nachzuvollziehen, kann man die Spaltenvektoren einer Figur zu einer $2 \times m$-Matrix zusammenfügen:

```
x = [0 1 1 2 2 1 1 3 3 0 0
     0 0 1 1 2 2 3 3 4 4 0]
```

und diese Matrix mit `y = A*x` rechts an die Transformationsmatrix `A` multiplizieren. Zum Beispiel mit `plot(y(1,:),y(2,:)), axis equal` lässt sich das Ergebnis dann plotten.

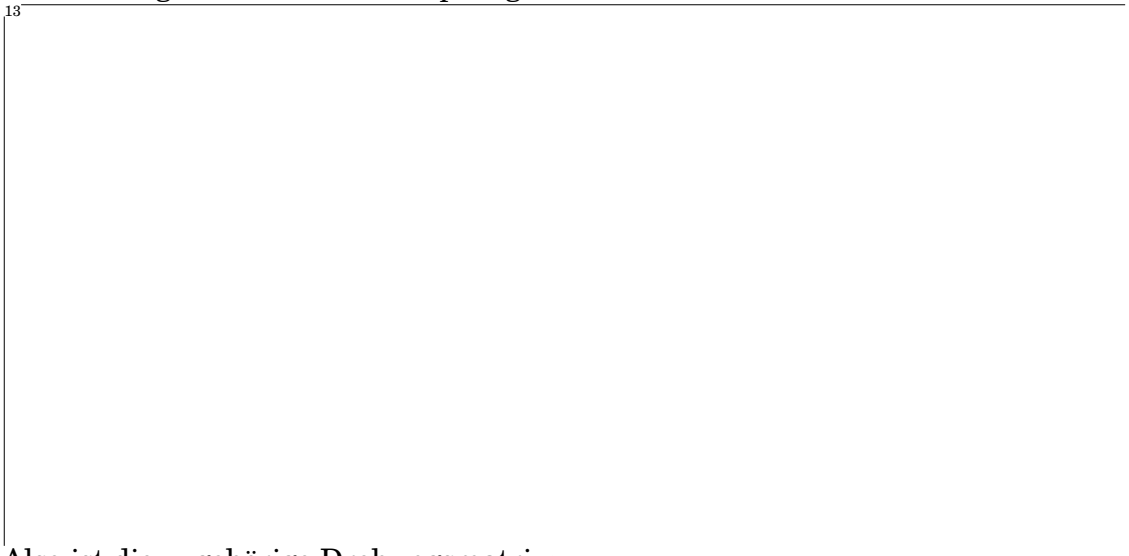
5 Drehungen

Um die Wirkung einer Matrix zu verstehen, kann man am einfachsten ansehen, wie sie auf die Vektoren der Standardbasis wirkt:

[12]

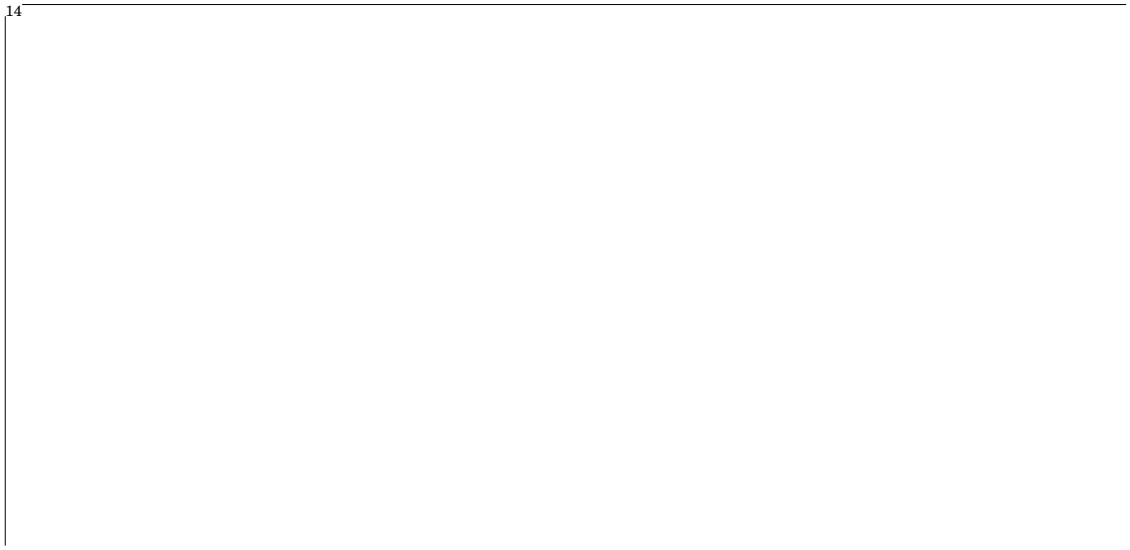
Sehen wir uns das für eine Drehung um +42° (mathematisch positiv, also gegen den Uhrzeigersinn) um den Ursprung an:

[13]



Also ist die zugehörige Drehungsmatrix:

14



Entsprechend lassen sich die Matrizen für beliebige Drehungen um den Ursprung bestimmen. (Kann man auch Drehungen um andere Punkte als den Ursprung so beschreiben?) In drei Dimensionen geht das ähnlich – ist aber deutlich schwieriger, weil nicht nur der Drehungswinkel, sondern auch die Drehungsachse festgelegt werden muss. Das Vorgehen bei Spiegelungen ist entsprechend (→ Praktikum).

6 Verschiebungen

Eine Verschiebung zum Beispiel um 2 in x-Richtung und 3 in y-Richtung lässt sich nicht – ohne Tricks – mit Hilfe einer Matrix schreiben. (Warum?) Statt dessen kann man rechnen:

15