Eingabe benutzerdefinierter Dgln

Der Benutzer kann maximal 16 eigene, sog. benutzerdefinierte **explizite Dgln. erster Ordnung**

$$\dot{y}_i = f_i\left(y_1, ... y_n, t\right) \qquad i = 1, 2, ... n \leq 16$$

eingeben und numerisch berechnen lassen. Dgln. sind explizit, wenn sie nach den höchsten Ableitungen (hier den *ersten* Ableitungen) aufgelöst sind.

Der Benutzer muss folgende Eingaben tätigen:

- **Name der Dgln**. (z. B. „Torsionsschwinger"). MECHANICUS setzt *automatisch* „Benutzerdef_" voran und arbeitet daher mit dem Namen „Benutzerdef_Torsionsschwinger". Der Anwender sollte daher bei der Namensgebung der Dgln. „Benutzerdef_" *nicht* eingeben. [1] Der Name der Dgln. darf nur aus einem einzigen Wort bestehen. Evtl. sind Unterstriche ‚_' zu verwenden.
- **Zahl der Variablen und Zahl der Parameter**. (Die Maximalzahl beträgt jeweils 16.)
- **Zahl von** (evtl. verwendeten) **Zusatzfunktionen.** [2] (Die Maximalzahl ist auch hier 16.)
- **Namen der Variablen** (z. B. „Phi" und „Phi-Pkt" oder „Radius r" und „r-Pkt").
- **Namen der Parameter** (z. B. „Masse m1" , „Masse m2" , „Reibung c").
- **Namen der evtl. Zusatzfunktionen** (z. B. „Energie" oder „Drehimpuls").

  Die Namen der Variablen, der Parameter und der (evtl. verwendeten) Zusatzfunktionen treten nicht in Gln. auf, sondern werden in folgenden Fällen *auf dem Bildschirm* gezeigt: Vor numerischen Rechnungen muss der Anwender die Anfangsbedingungen und Parameterwerte eingeben; bei der Grafik werden die Namen der Kurven angezeigt. Daher unterliegt die Namensgebung keinen Einschränkungen: Alle Zeichen (auch Umlaute ä, ö, ü) sind erlaubt und die Namen können aus mehreren getrennten Wörtern bestehen. Allerdings sollten die Namen nicht zu lang sein, da sie andernfalls nicht mehr vollständig in den Textfeldern dargestellt werden. (Enthält ein Variablenname oder ein ZFkName einen Unterstrich ‚_', so entfällt dieser Unterstrich bei der Kurvenansicht; stattdessen wird der darauf folgende Buchstabe des Namens tief gesetzt.)

- ***n* explizite Dgln. erster Ordnung.**

Bei numerischen Berechnungen und bei der Grafik arbeitet MECHANICUS mit den benutzerdefinierten Dgln. genauso wie mit den eingebetteten Dgln. **Die Bedienung von MECHANICUS ist also für alle Systeme und Dgln. völlig identisch.** Dies ist sicherlich ein Gewinn für die Bedienerfreundlichkeit des Programms.

Bei den benutzerdefinierten Dgln. ist allerdings keine Animation möglich und das Bild links oben im Hauptfenster, das das System darstellt, fehlt [3].

---

[1] Wegen des automatisch eingesetzten „Benutzerdef_" kann der eingetippte Name auch mit einer Zahl beginnen.

[2] Sinn und Bedeutung der Zusatzfunktionen sowie ihre Programmierung werden erst am Ende dieser Hilfestellung erläutert. Daher wird die Zahl der Zusatzfunktionen (abgekürzt ZFk) in den ersten 3 Beispielen gleich Null gesetzt, so dass sich der Leser vorerst noch nicht mit den Zusatzfunktionen beschäftigen muss.

Bei der Eingabe der Dgln. sind einige Punkte zu beachten:

- *Am Ende jeder Gl. sollte unbedingt ein Semikolon* (Strichpunkt) *stehen.*

  Wenn eine Gl. am Ende kein Semikolon hat, so wird das Ergebnis dieser Gl. nach jedem Aufruf im Command-Window von MatLab ausgeschrieben. Dies kann die Rechenzeit durchaus um einen dreistelligen Faktor erhöhen. MECHANICUS bemerkt und meldet ein fehlendes Semikolon.

- *Drei Punkte … am **Ende** einer Zeile* sagen MatLab, dass die Gl. in der folgenden Zeile fortgesetzt wird.
- Die Namen t für die Zeit, y(1), y(2), … für die Variablen und yPkt(1), yPkt(2), … für die Zeitableitungen der Variablen dürfen *nicht* (!) geändert werden.
- *Kommentare werden mit dem Prozentzeichen % eingeleitet.*
- MatLab *unterscheidet zwischen Groß- und Kleinschreibung*.
- **Bei einer Änderung der Zahl der Variablen werden alle bis dahin vom Benutzer getippten Variablennamen und die Dgln. gelöscht.** Zusatzfunktionen werden nicht gelöscht. Evtl. ist eine Speicherung der eingetippten Dgln. im *Zwischenspeicher* empfehlenswert, bevor die Variablenzahl geändert wird.
- **Bei einer Änderung der Zahl der Parameter werden alle bis dahin vom Benutzer getippten Parameternamen und die Dgln. gelöscht.** Zusatzfunktionen werden nicht gelöscht. Evtl. ist eine Speicherung der eingetippten Dgln. im *Zwischenspeicher* empfehlenswert, bevor die Parameterzahl geändert wird.
- **Bei einer Änderung der Zahl der Zusatzfunktionen werden die Namen der Zusatzfunktionen und die Gln. für die Zusatzfunktionen gelöscht**, nicht aber die Variablen- und Parameternamen und auch nicht die Dgln.

Bemerkung für MatLab-Kenner: Wenn die Zahl der Dgln. 1. Ordnung größer als Eins ist, so reserviert MECHANICUS *automatisch* Speicherplatz für yPkt; dadurch werden zeitraubende, dynamische Speicherbelegungen vermieden. Auch für die Zusatzfunktionen reserviert MECHANICUS automatisch Speicherplatz.

MECHANICUS überprüft vor dem Speichern grob, ob die eingetippten Gln. numerisch berechnet werden können, und meldet dem Benutzer, welche Gl. nicht berechnet werden kann. MECHANICUS bringt z. B. eine Fehlermeldung, wenn versehentlich san(y(1)) anstelle von sin(y(1)) eingetippt wurde, wenn eine Klammer vergessen wurde, wenn ein unbekannter Parameter in den Dgln. auftritt usw.…

---

[3] Der Benutzer kann ein eigenes pcx- oder jpg-Bild einbinden. Das Bild muss die Endung pcx oder jpg und den gleichen Vornamen wie die Dgln. haben – in unserem Fall also z. B. „Benutzerdef_Torsionsschwinger.pcx“. Das Bild steht – wie die anderen pcx-Bilder – im Verzeichnis C:\Mechanicus\Bilder. Das Verhältnis Breite/Höhe sollte ungefähr 1,4 sein.

In gleicher Weise kann der Benutzer auch eine Word-Datei mit der Endung doc oder eine Acrobat-Datei mit der Endung pdf im Verzeichnis C:\Mechanicus\Anmerkungen einfügen. Auch diese Dateien müssen den gleichen Vornamen wie die Dgln. haben – in unserem Fall also „Benutzerdef_Torsionsschwinger.doc“ oder „Benutzerdef_Torsionsschwinger.pdf“. Diese Dateien mit beliebigen Anmerkungen, Rechnungen und Kommentaren zu den Dgln. können innerhalb von MECHANICUS im Pulldown-Menü „Benutzerdef. Dgln“ geöffnet, gelesen oder auch geändert werden.

Es wird aber nicht garantiert, dass MECHANICUS alle denkbaren Fehler findet. Natürlich kann MECHANICUS auch nicht ermitteln, ob die Dgln. falsch aufgestellt wurden, ob also z. B. fälschlich cos(y(1)) anstelle von sin(y(1)) auftritt.

MECHANICUS macht keine genauere Angabe zum Fehler. Nach der Bestätigung der Fehlermeldung springt der Cursor in das Feld mit dem gemeldeten Fehler.

Im Folgenden werden einige mögliche Fehler aufgeführt.

---

1. Beispiel: Für die leere Zuweisung = Par_Vektor(1) in der ersten Zeile wird die Fehlermeldung

Die 1-te Gl. = …. enthält einen Fehler.

gegeben, da links vom Gleichheitszeichen Nichts steht.

---

2. Beispiel: Für die richtige Dgl. yPkt(2) = - D/m * y(1) wird die Fehlermeldung

Die 4-te Gl. yPkt(2) = …. enthält einen Fehler.

gegeben, wenn der Parameter *D* oder *m* zuvor keine Zuweisung – etwa in der Form m = Par_Vektor(1) – erhalten hat.

---

3. Beispiel: Für die zwei Gln.

```
SinTheta = sin( y(3) );        % Dies ist eine Abkürzung
yPkt(2) = y(1) * sinTheta;
```

wird die Fehlermeldung

Die 5-te Gl. yPkt(2) = …. enthält einen Fehler.

gegeben, weil SinTheta in der ersten Gl. *groß* geschrieben wird, in der Dgl. aber *klein*. Da MatLab zwischen Groß- und Kleinschreibung unterscheidet, ist die Variable sinTheta in der Dgl. nicht definiert.

---

4. Beispiel: Wenn z. B. die 3. Zeile einen Kommentar enthält, der *nicht* mit dem Kommentarzeichen % eingeleitet wird, so wird die Fehlermeldung

Die 3-te Gl. …. enthält einen Fehler.

gegeben. In diesem Fall muss das Zeichen % vor dem Kommentar geschrieben werden.

Die eingetippten Namen und Gln. werden von MECHANICUS erst gespeichert, wenn keine Fehler mehr gemeldet werden. Wenn der Benutzer einen gemeldeten Fehler nicht näher einkreisen kann, so empfiehlt es sich, Teile der falschen Gl. mit dem Prozentzeichen % auszukommentieren und anschließend die Gl. nach und nach zu erweitern. Die Auskommentierung empfiehlt sich auch, wenn die Arbeit aus zeitlichen Gründen verfrüht gespeichert werden muss, ohne dass der von MECHANICUS gemeldete Fehler gefunden wurde. In einer der folgenden Sitzungen können die Dgln. unter dem Menüpunkt „Dgln. ansehen oder ändern“ in Ruhe korrigiert werden.

Am Ende dieser Hilfe-Datei werden **wichtige MatLab-Funktionen** aufgezählt.

Im Folgenden werden die einzelnen Arbeitsschritte an Hand von drei Beispielen erläutert. Dabei werden die vom Benutzer getippten Eingaben **fett** und **in blauer Farbe** geschrieben. Die Zahl der Zusatzfunktionen (abgekürzt ZFk) wird in allen drei Beispielen gleich Null gesetzt, weil wir erst nach diesen drei Beispielen auf die Zusatzfunktionen eingehen möchten.

Wenn bestimmte Ausdrücke in den Dgln. häufiger vorkommen, so empfiehlt sich zur Verkürzung der Rechenzeit und evtl. auch zur Erhöhung der Übersichtlichkeit die Einführung von sinnvollen, d. h. aussagekräftigen **Abkürzungen**. Beispiele sind:

v = m1 / (m1 + m2);

SinPhi = sin(y(1));

Die Abkürzungen können danach in den Dgln. verwendet werden.

Beispiel 1 Gefechtsmodell

Zwei feindliche Armeen mit den Mannschaftsstärken $N_{\text{Blau}}(t)$ und $N_{\text{Rot}}(t)$ und mit den Feuerkräften α_{Blau} und α_{Rot} führen eine Schlacht. [4] Schlachtverlauf und Verluste können in gröbster Näherung durch zwei gekoppelte Dgln. beschrieben werden:

$$\dot{N}_{\text{Blau}}(t) = -\alpha_{\text{Rot}}\ N_{\text{Rot}}(t)$$

$$\dot{N}_{\text{Rot}}(t) = -\alpha_{\text{Blau}}\ N_{\text{Blau}}(t)$$

Abb. 1 Nur der linke Teil der Eingabemaske für die benutzerdefinierten Dgln. wird dargestellt. Der rechte Teil fehlt.

Da die Dgln. bereits explizit und von 1. Ordnung sind, müssen sie nicht mehr umgeschrieben werden. Die Eingabe in MECHANICUS könnte z. B. wie folgt aussehen (siehe Abb. 1):

Name der Dgln: **Gefechtsmodell**

Zahl der Dgln: **2** Zahl der Parameter: **2** Zahl der ZFk: **0**

Name von y(1) : **Zahl der Blauen** Name von y(2): **Zahl der Roten**

[4] Die Namen „Blau“ und „Rot“ gehen auf die Zeit des kalten Krieges zurück. Bei Nato-Manövern trafen zwei „Armeen“ aufeinander: Die Blauen waren die Nato-Truppen, die Roten die Truppen des Warschauer-Paktes.

1-ter Par-Name: **Feuerkraft-Blau** 2-ter Par-Name: **Feuerkraft-Rot**

Nun werden die zwei Dgln. eingegeben. Die Zahlenwerte der Parameter „Feuerkraft-Blau“ und „Feuerkraft-Rot“, die vor jeder numerischen Rechnung eingetippt werden müssen, werden in MECHANICUS an Par_Vektor(1) und Par_Vektor(2) übergeben. Da die Verwendung dieser beiden langen Bezeichnungen in den Dgln. unschön ist, werden zwei Abkürzungen eingeführt:

FK_blau = Par_Vektor(1); (1a)

FK_rot = Par_Vektor(2); (1b)

yPkt(1) = **- FK_rot * y(2)**;

yPkt(2) = **- FK_blau * y(1)**;

Hinweis: Natürlich kann man die Zuweisungen in den Gln. (1a) und (1b) ersatzlos streichen, wenn die Dgln. wie folgt lauten:

yPkt(1) = - Par_Vektor(2) * y(2);

yPkt(2) = - Par_Vektor(1) * y(1);

Man sollte aber bedenken, dass die Dgln. viel besser zu lesen sind, wenn anstelle der nichtssagenden Vektorkomponenten Par_Vektor(i) aussagekräftigere Parameternamen wie z. B. FK_rot , m , D , L , … eingeführt werden. Wir wollen daher in allen Dgln. – auch in den Zusatzfunktionen – die Parameterzuweisungen beibehalten.

Die Eingabe ist nun fertig und lässt sich mit dem Button Dgln. speichern speichern. MECHANICUS prüft die Berechenbarkeit aller Eingaben. Nach dem Wechsel ins Hauptmenü können numerische Berechnungen durchgeführt werden. Die Dgln. werden unter

C:\Mechanicus\Dgln\ Benutzerdef_Gefechtsmodell_Dgln.m

gespeichert und lassen sich unter dem Menüpunkt „Dgln. ansehen oder ändern“ jederzeit ansehen oder auch bearbeiten.

Die Parameternamen FK_blau und FK_rot sind Variablennamen und treten in den Dgln. und in den (evtl. definierten) Zusatzfunktionen auf. Daher müssen sie aus einem einzigen Wort bestehen, mit einem Buchstaben beginnen und dürfen max. nur 31 Buchstaben, Zahlen und Unterstriche enthalten – ohne ä, ü, ö, ß.

Die Dgln. des Gefechtsmodells sind bereits als benutzerdefinierte Dgln. in MECHANICUS enthalten.

Beispiel 2 Erzwungene harmonische Schwingungen

Die Dgl. zweiter Ordnung

$$m\ddot{x} + c\dot{x} + Dx = DA\sin(\Omega t)$$

muss zuerst in zwei Dgln. erster Ordnung überführt werden. Mit den Definitionen

$$y(1) := x \qquad y(2) := \dot{x}$$

ergeben sich zwei explizite Dgln. erster Ordnung:

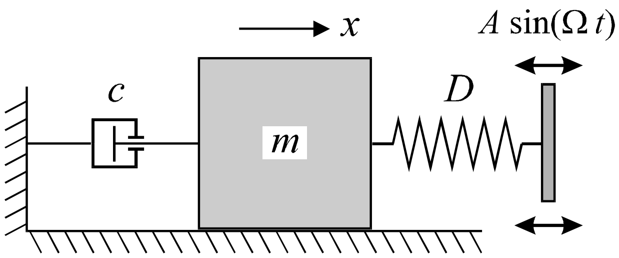

Abb. 2 Der reibungsfrei auf dem Boden gleitende Oszillator wird durch die bewegte Wand zu Schwingungen angeregt.

$$\dot{y}(1) = y(2);$$

$$\dot{y}(2) = -D/m \cdot y(1) - c/m \cdot y(2) + DA\sin(\Omega t)$$

Die Eingabe in MECHANICUS könnte z. B. wie folgt lauten:

Name der Dgln: **Erzwungene_harmon_Schwingungen**

Zahl der Dgln: **2** Zahl der Parameter: **5** Zahl der ZFk: **0**

Name von y(1) : **x** Name von y(2): **x-Pkt**

1-ter Par-Name: **Masse m** 2-ter Par-Name: **Federkonstante D**

3-ter Par-Name: **Reibung c** 4-ter Par-Name: **Amplitude A**

5-ter Par-Name: **Omega**

Nun werden die zwei Dgln. eingegeben. Die Zahlenwerte der fünf Parameter, die vor jeder numerischen Rechnung anzugeben sind, werden in MECHANICUS an Par_Vektor(1) bis Par_Vektor(5) übergeben. Da die Verwendung dieser fünf nichtssagenden Bezeichnungen in den Dgln. unschön ist, werden wie folgt fünf aussagekräftige Parameternamen eingeführt. *Dabei muss die oben vorgegebene Reihenfolge der Parameter unbedingt* (!) *eingehalten werden.*

m = Par_Vektor(1); **D** = Par_Vektor(2);

c = Par_Vektor(3); **A** = Par_Vektor(4); **Om** = Par_Vektor(5);

Die zwei expliziten Dgln. 1. Ordnung lauten nun:

yPkt(1) = y(2);

yPkt(2) = **– D/m * y(1) - c/m * y(2) + D * A * sin(Om * t)**;

Vier Hinweise: **1)** *Die Zeit muss* (!) *immer das Formelzeichen t haben.*

2) Die erste Dgl. „yPkt(1) = y(2)“ wird von Mechanicus vorgegeben, da meistens n Dgln. 2. Ordnung in $2n$ Dgln. 1. Ordnung umgewandelt werden. Bei Bedarf kann diese Vorgabe natürlich geändert werden wie in Beispiel 1.

3) Die Abkürzungen m, D, c, A, Om sind Variablennamen und treten in den Dgln. und in den (evtl. definierten) Zusatzfunktionen) auf. Daher müssen sie aus einem einzigen Wort bestehen, mit einem Buchstaben beginnen und dürfen max. nur 31 Buchstaben, Zahlen und Unterstriche enthalten – ohne ä, ü, ö, ß.

4) Die Dgln. des harmonischen Oszillators sind bereits als benutzerdefinierte Dgln. in MECHANICUS enthalten.

Beispiel 3 Kettenkarussell

Die Dgln. für die Winkel φ, ϑ eines angetriebenen reibungsfreien Kettenkarussells lauten:

$$\ddot{\varphi} = -\frac{1}{\sin\vartheta}\left[2\,(\omega+\dot{\varphi})\,\dot{\vartheta}\cos\vartheta + \frac{R}{l}\,\omega^2\sin\varphi\right]$$

$$\ddot{\vartheta} = (\omega+\dot{\varphi})^2\sin\vartheta\cos\vartheta + \frac{R}{l}\,\omega^2\cos\varphi\cos\vartheta - \frac{g}{l}\sin\vartheta$$

Zuerst müssen vier explizite Dgln. 1. Ordnung aufgestellt werden. Mit den Definitionen

$$y(1) := \varphi \qquad y(2) := \dot{\varphi}$$

$$y(3) := \vartheta \qquad y(4) := \dot{\vartheta}$$

ergeben sich vier explizite Dgln. 1. Ordnung:

$$\dot{y}(1) = y(2);$$

$$\dot{y}(2) = -\frac{1}{\sin y(3)}\left[2\,(\omega+y(2))\;y(4)\cos y(3) + \frac{R}{l}\,\omega^2\sin y(1)\right]$$

$$\dot{y}(3) = y(4);$$

$$\dot{y}(4) = (\omega+y(2))^2\sin y(3)\cos y(3) + \frac{R}{l}\,\omega^2\cos y(1)\cos y(3) - \frac{g}{l}\sin y(3)$$

Die Eingaben in MECHANICUS könnten z. B. wie folgt lauten:

Name der Dgln: **Kettenkarussell**

Zahl der Dgln: **4** Zahl der Parameter: **3** Zahl der ZFk: **0**

Name von y(1) : **Phi** Name von y(2): **Phi-Pkt**

Name von y(3) : **Theta** Name von y(4): **Theta-Pkt**

1-ter Par-Name: **Radius R** 2-ter Par-Name: **Länge l**

3-ter Par-Name: **Omega**

Als nächstes werden die vier Dgln. eingegeben. Die Zahlenwerte der drei Parameter, die vor jeder numerischen Rechnung anzugeben sind, werden in MECHANICUS an Par_Vektor(1) bis Par_Vektor(3) übergeben. Da die Verwendung dieser drei langen und Nichts sagenden Bezeichnungen in den Dgln. unschön ist, werden wie folgt drei signifikante Parameternamen eingeführt. *Dabei muss die oben vorgegebene Reihenfolge der Parameter unbedingt eingehalten werden.*

```
R = Par_Vektor(1);          l = Par_Vektor(2);          Om = Par_Vektor(3);
```

Die Funktionen $\sin\vartheta$ und $\cos\vartheta$ treten *zwei-* bzw. *dreimal* in den Dgln. auf. Daher ist es sehr ratsam, diese Funktionen durch **Abkürzungen** zu ersetzen, die jeweils nur *einmal* berechnet werden. Das erhöht die Geschwindigkeit der numerischen Berechnungen erheblich.

```
Sin_Theta  = sin( y(3) );

Cos_Theta = cos( y(3) );
```

Die vier expliziten Dgln. 1. Ordnung lauten nun:

```
yPkt(1) = y(2);

yPkt(2) = - 1 / Sin_Theta * ( 2 * ( Om + y(2) ) * y(4) * Cos_Theta + ...

              R / l * Om^2 * sin( y(1) ) ) ;

yPkt(3) = y(4);

yPkt(4) =  ( Om + y(2) )^2 * Sin_Theta * Cos_Theta + ...

              R / l * Om^2 * cos( y(1) ) * Cos_Theta – 9.81 / l * Sin_Theta ;
```

Wir kommen nun auf die **<u>Zusatzfunktionen</u>** zu sprechen. Bei vielen Systemen möchte man nicht nur die Lösungen $y_1(t),\ y_2(t),\ .....\ y_n(t)$ der Dgln. kennen, sondern auch den zeitlichen Verlauf von speziellen Funktionen

$$F\left[\ y_1(t),\ y_2(t),\ .....\ y_n(t)\ ,\ t\ \right] =:\ \hat{F}(t)$$

dieser Lösungen, die wir „Zusatzfunktionen" (abgekürzt **ZFk**) nennen wollen [5]. *Die Zusatzfunktionen hängen also nur von den Lösungen* $y_1(t),\ y_2(t),\ .....\ y_n(t)$ *der Dgln.* 1. *Ordnung und evtl. von der Zeit ab.* Zeitableitungen der Lösungen treten in den ZFk nicht auf.

Wir geben einige Beispiele von ZFk:

- In den meisten mechanischen Systemen möchte man den zeitlichen Verlauf der Energie $E(t)$ kennen. Ist die Energie konstant oder wie ändert sie sich im Laufe der Zeit?
- Wie verhalten sich Impulse und Drehimpulse?
- Welche Kräfte oder welche Drehmomente treten auf?
- Wie lauten die kartesischen Koordinaten der bewegten Körper? (Die Dgln. enthalten oft nicht kartesische, sondern andere Koordinaten – z. B. Zylinder- oder Kugelkoordinaten.)

---

[5] Bei vielen *eingebetteten* Systemen werden neben den Zusatzfunktionen auch noch Erhaltungsgrößen berechnet. Bei den *benutzerdefinierten* Systemen hingegen werden (eventuell definierte) Erhaltungsgrößen als Zusatzfunktionen betrachtet.

Die Untersuchung der Zusatzfunktionen ist in vielen Fällen entscheidend für das physikalische oder technische Verständnis eines Systems [6]. Daher kann der Anwender benutzerdef. Zusatzfunktionen zusammen mit den benutzerdef. Dgln. eintippen. Nach jeder numerischen Lösung der Dgln. werden die Zusatzfunktionen von MECHANICUS *automatisch* berechnet, zusammen mit den Lösungen $y_1(t), \ldots y_n(t)$ in einen gemeinsamen Array gepackt und dann auf der Festplatte gespeichert.

Beim Schreiben der Zusatzfunktionen gelten dieselben Regeln wie beim Schreiben der Dgln. – mit folgenden Ausnahmen:

- *Die Zeit hat jetzt* nicht mehr (wie noch bei den Dgln.) das Formelzeichen t, sondern das *Formelzeichen* T. (Beachte: MatLab unterscheidet zwischen Groß- und Kleinschreibung.)

  T ist ein Zeilenvektor, dessen Zeilenzahl gleich der Zahl der Zeitschritte ist, die bei der numerischen Berechnung durchlaufen wurden.

- *Die* i-*te Variable hat jetzt* nicht mehr (wie noch bei den Dgln.) das Formelzeichen y(i), sondern *das Formelzeichen*

  $$\mathrm{Y}(:,\mathrm{i})$$

  $\mathrm{Y}(:,\mathrm{i})$ ist ein Zeilenvektor, dessen Zeilenzahl gleich der Zahl der Zeitschritte ist, die bei der numerischen Berechnung durchlaufen wurden. Anstelle von $\sin(\mathrm{y}(3))$ muss bei den Zusatzfunktionen also $\sin(\mathrm{Y}(:,3))$ geschrieben werden.

- Auch die Zusatzfunktionen $\mathrm{ZFk}(:,\mathrm{i})$ sind Zeilenvektoren und enthalten einen Doppelpunkt.

- Multiplikation, Division und Potenzieren von Variablen $\mathrm{Y}(:,\mathrm{i})$ und von Zusatzfunktionen $\mathrm{ZFk}(:,\mathrm{i})$ haben nicht mehr (wie noch bei den Dgln.) die Zeichen * , / , ^ , sondern die Zeichen .* , ./ , .^ . Mit anderen Worten: *Unmittelbar vor das normale Operationszeichen muss unbedingt ein Punkt gesetzt werden* [7]. Beispielsweise muss anstelle von $y(1)\hat{}2 * \sin(y(3))$ geschrieben werden:

  $$\mathrm{Y}(:,1)\ .\hat{}2\ .*\ \sin(\mathrm{Y}(:,3))$$

---

[6] In vielen mechanischen Systemen sind die Zusatzfunktionen zeitlich konstant, also sog. Erhaltungsgrößen. Beispiele sind die mechanische Energie reibungsfreier, abgeschlossener mechanischer Systeme oder der Drehimpuls bei Zentralkräften. Solche konstanten Zusatzfunktionen helfen oft bei der Klärung der Frage, ob die Dgln. richtig aufgestellt wurden: Wenn man nach der numerischen Berechnung der Dgln. feststellt, dass Erhaltungsgrößen nicht konstant sind (abgesehen von kleinen Schwankungen, die auf numerische Fehler zurückzuführen sind und die durch Verringerung der relativen Toleranz RelTol und der absoluten Toleranz AbsTol verringert werden können), dann weiß man: Entweder wurden die Dgln. falsch aufgestellt oder falsch in MECHANICUS eingetippt (oder aber die Erhaltungsgrößen sind nicht richtig.)

[7] Der vorgesetzte Punkt beschreibt in MatLab eine *elementweise* Operation von Matrizen. So gilt z. B.

$$[2.5\ \ 3]\,.*[2\ \ 3] = [5\ \ 9] \qquad [2.5\ \ 3]\,./[2\ \ 3] = [1.25\ \ 1]$$

$$[2.5\ \ 3]\,.\hat{}[2\ \ 3] = [6.25\ \ 27]$$

Bei Addition und Substraktion von Matrizen und bei der Rechnung mit Skalaren wird der Punkt nicht benötigt.

- Die Zuweisungen, die der Anwender direkt oberhalb der Dgln. eingetippt hat — also z. B. m = Par_Vektor(1) und D = Par_Vektor(2) — gelten *automatisch* auch für die Zusatzfunktionen.

- Beschleunigungen können in den ZFk *nicht direkt* verwendet werden, da sie bei der numerischen Lösung der Dgln. nicht berechnet werden. Eine Beschleunigung lässt sich aber ermitteln, indem man die rechte Seite der entsprechenden expliziten Dgl. bei den Zusatzfunktionen mit den Variablen Y(:, i) erneut berechnet.

Beispiel 4 Pendel mit vertikal schwingendem Aufhängepunkt

Wir betrachten ein ebenes Pendel der Länge l, dessen Aufhängepunkt vertikale harmonische Schwingungen $A \sin \Omega t$ macht. Die Dgl. lautet

$$\ddot{\varphi} = -\frac{1}{l}\left[g - A\,\Omega^2 \sin(\Omega t)\right]\sin\varphi$$

Mit den Definitionen $y(1) := \varphi$ $y(2) := \dot{\varphi}$
ergeben sich zwei explizite Dgln. 1. Ordnung:

$$\dot{y}(1) = y(2);$$

$$\dot{y}(2) = -\frac{1}{l}\left(9.81 - A\,\Omega^2 \sin(\Omega t)\right)\sin(y(1))$$

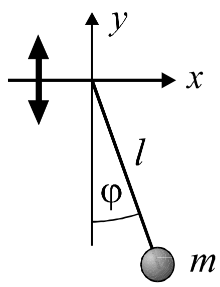

Abb. 3 Der Aufhängepunkt des Pendels wird harmonisch auf und ab bewegt.

Die Pendelenergie sowie die x- und y-Koordinate der Kugel sollen Zusatzfunktionen sein. Daher könnten die Eingaben in MECHANICUS z. B. wie folgt lauten:

Name der Dgln: **Pendel_mit_ vertikal_schwing_Aufhaeng**

Zahl der Dgln: **2** Zahl der Parameter: **4** Zahl der ZFk: **3**

Name von y(1) : **Phi** Name von y(2): **Phi-Pkt**

1-ter Par-Name: **Länge *l*** 2-ter Par-Name: **Amplitude A**
3-ter Par-Name: **Omega** 4-ter Par-Name: **Masse m**

1-ter ZFk-Name: **Energie** 2-ter ZFk-Name: **x** 3-ter ZFk-Name: **y**

l = Par_Vektor(1); **A** = Par_Vektor(2);

Om = Par_Vektor(3); **m** = Par_Vektor(4);

Die zwei expliziten Dgln. 1. Ordnung lauten nun:

yPkt(1) = y(2);

yPkt(2) = **- 1 / l * (9.81 – A * Om^2 * sin(Om * t)) * sin(y(1))**;

Als Zusatzfunktionen wählen wir die Energie

$$E = \frac{m}{2} m\, l^2\, \dot{\varphi}^2 + 2\, l\, \Omega\, A\, \dot{\varphi} \cos(\Omega\, t) \sin\varphi + A^2 \cos^2(\Omega\, t) + 9.81 \cdot m \left[A \sin(\Omega\, t) + l\, (1 - \cos\varphi) \right]$$

und die x- und y-Koordinate der Kugel:

$$x = l \sin\varphi \qquad y = A \sin(\Omega\, t) - l \cos\varphi$$

Die Energie wurde so normiert, dass sie Null ist für $A = 0$ und $\varphi(t) = 0$. Zur Verkürzung der Rechenzeit definieren wir folgende **Abkürzungen**, die in den ZFk mehrmals auftreten:

```
sin_Om_t  =  sin(Om * T);
cos_Om_t  =  cos(Om * T);
sin_Phi   =  sin( Y(:,1) );
cos_Phi   =  cos( Y(:,1) );
```

Die vier Abkürzungen sind Zeilenvektoren. Damit lauten die ZFK:

```
ZFk(: , 1)  =   0.5 * m * l^2 * ( Y(:,2)  .*  Y(:,2) )  +  ...
                2 * l * Om * A * Y(:,2) .* cos_Om_t  .*  sin_Phi  +  ...
                A^2 * cos_Om_t .* cos_Om_t  +  ...
                9.81 * m * ( A * sin_Om_t  +  l * ( 1 - cos_Phi ) );

ZFk(: , 2)  = l * sin_Phi ;
ZFk(: , 3)  = A * sin_Om_t  -  l * cos_Phi ;
```

Die Dgln. des Pendels mit vertikal schwingendem Aufhängepunkt sind bereits als benutzerdefinierte Dgln. in MECHANICUS enthalten.

In der folgenden Tabelle werden die wichtigsten **MatLab-Funktionen** aufgezählt.

$x * y$	x mal y
$x\,/\,y$	x dividiert durch y
x^y	x hoch y
abs(x)	Betrag von x
acos(x)	Arkuskosinus von x
acosh(x)	Areacosinus hyperbolicus von x
asin(x)	Arkussinus von x
asinh(x)	Areasinus hyperbolicus von x
atan(x)	Arkustangens $y = \mathrm{atan}(x)$ mit $y \in [-\pi/2, \pi/2]$

atanh(*x*)	Areatangens hyperbolicus von *x*
ceil(*x*)	Rundet *x* auf die nächste größere ganze Zahl
cos(*x*)	Kosinus von *x*
cosh(*x*)	Kosinus hyperbolicus von *x*
exp(*x*)	Exponentialfunktion
fix(*x*)	Ganzzahliger Anteil von *x*
floor(*x*)	Rundet *x* auf die nächste kleinere ganze Zahl
log(*x*)	Natürlicher Logarithmus von *x*
log10(*x*)	Dekadischer Logarithmus von
mod(*x*,*y*)	Modulo-Funktion
rem(*x*,*y*)	Rest der Division von *x* durch *y*
round(*x*)	Rundet *x* auf die nächste ganze Zahl
sign(*x*)	Vorzeichen von *x* mit den Werten -1,+1 oder 0.
sin(*x*)	Sinus von *x*
sinh(*x*)	Sinus hyperbolicus von *x*
sqrt(*x*)	Quadratwurzel von *x*
tan(*x*)	Tangens von *x*
tanh(*x*)	Tangens hyperbolicus von *x*

Zwei wichtige **Konstanten** lauten:

9.81 = Wert der Erdbeschleunigung

pi = Wert von π = 3.141592 ...