

## Modellierung physikalischer Systeme

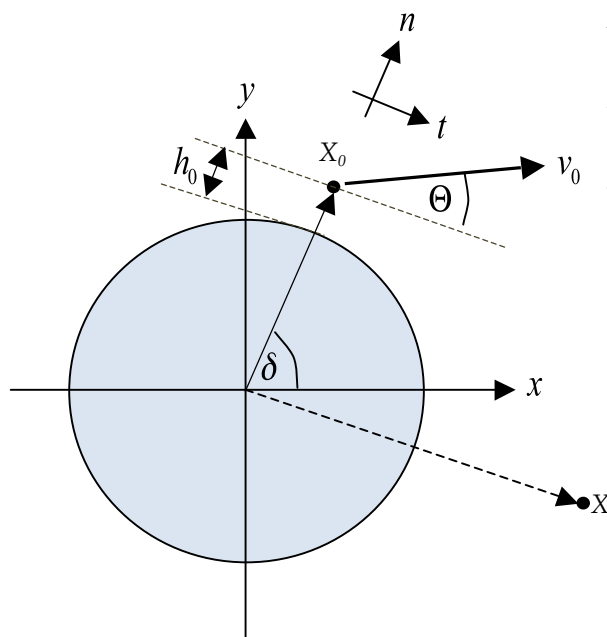
### Lernziele:

- Entwicklung der Differentialgleichungen aus dem physikalischen Modell
- Realisierung in Matlab/Simulink
- Einfache Ereignisse und Parameterumschaltung
- Grafische Ausgaben unter Matlab/Simulink

### Aufgabenstellungen:

#### 1. Weltraummissionen

##### 1.1 Erdumkreisung, Fluchtgeschwindigkeit und geostationäre Bahn



#### Aufgabenstellung:

Ein Satellit wird mit einer Trägerrakete in die Startposition  $x_0$  gebracht und fliegt von dort antriebslos weiter. Dort hat er die Geschwindigkeit  $v_0$  und den Flugwinkel  $\theta$ .

Zu simulieren ist nur die antriebslose Phase ab  $x_0$ .

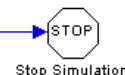
Die Flugbahn kann als ebenes Problem modelliert werden (Flug in der Äquatorebene).

Der Einfluss des Satelliten auf die Erde ist vernachlässigbar.

Satellitenposition zu einem späteren Zeitpunkt

#### Simulationsrandbedingungen:

- Als Parameter sollen vorgebar sein :
  - Startgeschwindigkeit  $v_0$  (in km/s) und Startflugwinkel  $\theta$  (in  $^\circ$ ),
  - Starthöhe  $h_0$  über Meeresspiegel (in km) und Startpositionswinkel  $\delta$  (in  $^\circ$ ).
- Es sollen Vektorintegratoren verwendet werden.  
Anm.: Ein Integrator wird automatisch zum Vektorintegrierer, wenn auf den Eingang ein Vektorsignal gegeben wird. Auch der Startwert muss dann als Vektor angegeben werden.
- Die Bahnkoordinaten sollen in den Matlab-Workspace geschrieben werden, wo sie angezeigt werden.
- Die Simulation soll abbrechen, wenn der Satellit die Erde berührt.



## Modellierung physikalischer Systeme

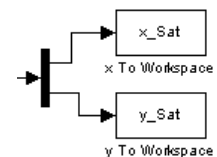
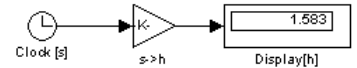
### Formeln und Konstanten:

$$\vec{F}_S = G \cdot \frac{m_E \cdot m_S}{r^2} \cdot \vec{e}_{SE}$$

Erdradius :	$r_E = 6378 \text{ km}$
Erdmasse :	$m_E = 5.9736 \cdot 10^{24} \text{ kg}$
Gravitationskonstante:	$G = 66.743 \cdot 10^{-12} \frac{\text{m}^3}{\text{kg} \cdot \text{s}^2}$

### Modellierung – Schritt für Schritt:

- Geben Sie eine Embedded-Matlab-(EM)-Funktion „**Startposition**“ an, die aus den Parametern  $\delta(^{\circ})$ , der Starthöhe  $h_0$  (km) sowie der Konstante „Erdradius“ den Startpositionsvektor  $\vec{x}_0$  berechnet ( $\rightarrow$  Anfangswert für die Position).
- Geben Sie eine EM-Funktion „**vStart**“ an, die aus den Parametern  $v_0$ ,  $\theta$  und der berechneten Startposition  $\vec{x}_0$  den Startgeschwindigkeitsvektor  $\vec{v}_{0,Welt}$  berechnet ( $\rightarrow$  Anfangswert der Geschwindkt. in Weltkoordinaten).  
Hinweis: - Einheitsvektoren in Tangential- und Normalenrichtung ( $\vec{n}$ ,  $\vec{t}$ ) aus  $\vec{x}_0$  konstruieren,  
- Tangential- und Normalkomponente der Startgeschwindigkeit ( $v_t$ ,  $v_n$ ) berechnen,  
- aus  $\vec{n}$ ,  $\vec{t}$  und ( $v_t$ ,  $v_n$ ) die Startgeschwindigkeit  $\vec{v}_{0,Welt}$  zusammenbauen.
- Geben Sie eine EM-Funktion „**Beschleunigung**“ an, die aus der aktuellen Satellitenposition  $\vec{x}$  die Satellitenbeschleunigung errechnet (Hinweis:  $\Sigma F=ma$ ).
- Geben Sie eine EM-Funktion „**Kontakt**“ an, die 0 ausgibt solange die Satellitenposition über der Erdoberfläche liegt, sonst 1.
- Zeigen Sie die Simulationszeit (in Stunden) in einem Display an.
- Geben Sie die Bahndaten (x,y) wie folgt in den Matlab-Workspace aus.



### Versuchsdurchführung:

Für die Darstellung der Flugbahn liegt ein Matlab-Skript „*Erdbahn.m*“ in meinem Pub.

- Voreinstellungen:  $\delta=30^{\circ}$ ,  $h_0=400\text{km}$  (z.B. ISS),  $\theta=0^{\circ}$ .  
Bestimmen Sie die Startgeschwindigkeit  $v_0$  so, dass der Satellit eine Kreisbahn in gleicher Höhe fliegt.  
Wie lange dauert eine Erdumkreisung (Simulationszeit variieren)?  
Tipp:  $v_0 \approx 7.5 - 8.5 \text{ km/s}$ , Simulationszeit  $\approx 1 - 2 \text{ h}$
- Voreinstellungen: wie a, Simulationszeit  $10^6 \text{ s}$ .  
Bestimmen Sie die Startgeschwindigkeit  $v_0$  so, dass der Satellit gerade der Erde entflieht.  
Tipp:  $v_0 \approx 10 - 11 \text{ km/s}$   $\rightarrow$  „*Fluchtgeschwindigkeit*“
- Voreinstellungen:  $\delta_0=30^{\circ}$ ,  $\theta=0^{\circ}$ .  
Bestimmen Sie die Starthöhe  $h_0$  und die Startgeschwindigkeit  $v_0$  so, dass eine Kreisbahn genau 1 Tag dauert.  
Tipp:  $h_0 \approx 40000 \text{ km}$ ,  $v_0 \approx 3 \text{ km/s}$   $\rightarrow$  „*geostationäre Bahn*“

Speichern Sie die Simulation unter dem Namen „*Erdorbits*“ ab.

Speichern Sie die Simulation für den nächsten Versuch erneut ab, aber jetzt unter dem Namen „*Mondmission*“.

## Modellierung physikalischer Systeme

---

### 1.2 Mondumkreisung

#### Aufgabenstellung:

Der Satellit soll jetzt von der Erde zum Mond fliegen.

Dabei wird der Mond vereinfachend als feststehend angenommen.

#### Konstanten:

Mondposition (fest):  $x_M = (0, -380.000)^T km$

Mondmasse :  $m_M = 7,3480 \cdot 10^{22} kg$

#### Modellierung:

- Ergänzen Sie die EM-Funktion „**Beschleunigung**“ entsprechend.
- Zeigen Sie die Simulationszeit (in Tagen) in einem Display an.

#### Versuchsdurchführung:

Für die Darstellung der Flugbahn liegt ein Matlab-Skript „*ErdMondBahn.m*“ in meinem Pub.

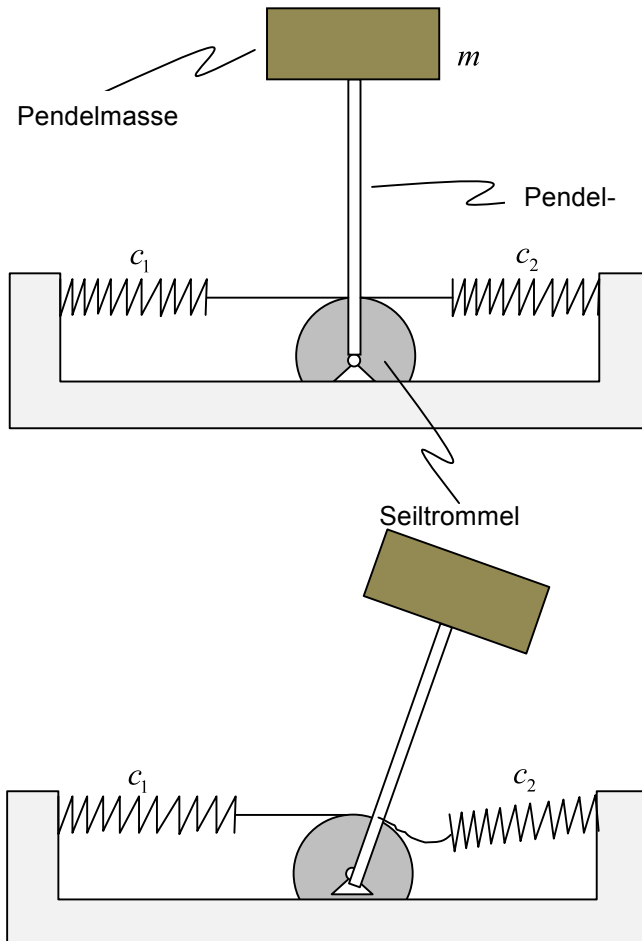
Voreinstellungen:  $\delta_0 = 30^\circ$ ,  $h_0 = 150 km$ .

Bestimmen Sie die Startgeschwindigkeit  $v_0$  und  $\theta$  so, dass der Satellit in einer 8-förmigen Schleife um den Mond und dann zur Erde zurück fliegt.

Wie lange dauert die Mondmission?

## Modellierung physikalischer Systeme

### 2. Crazy Pendulum



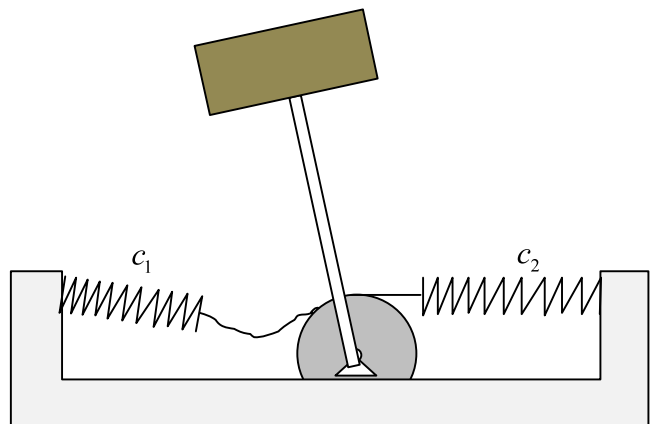
#### Aufgabenstellung:

Die Pendelmasse  $m$  ist über eine Pendelstange mit einer drehbaren Seiltrommel verbunden.

An der Seiltrommel sind zwei Federn über Seile befestigt. In der senkrechten Pendelstellung sind beide Federn gerade kraftfrei.

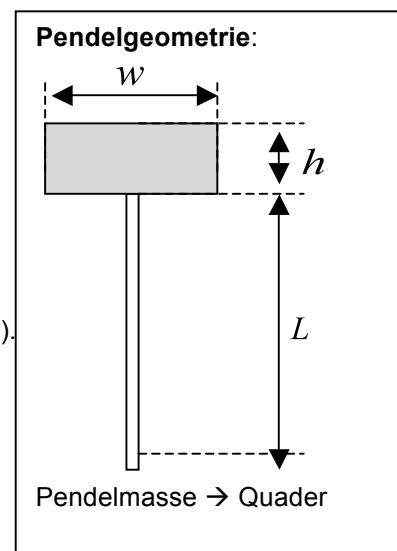
Ist das Pendel zur linken bzw. rechten Seite geneigt, so ist entweder nur Feder 2 bzw. nur Feder 1 wirksam.

Der Einfluss der Pendelstange und der Seiltrommel können bei der Simulation vernachlässigt werden.



#### Simulationsrandbedingungen:

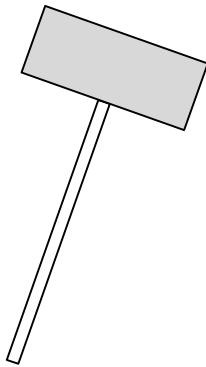
- Als Parameter sollen vorgebar sein :
  - Pendelmasse  $m$  [kg] ,
  - Abstand  $L$  [m], Quadermaße  $w$ [m] und  $h$ [m]
  - Trommelradius  $r$  [m].
  - Anfangsauslenkung  $\varphi_0$  [°] des Pendels,
  - Federkonstanten  $c_1$  [N/m] und  $c_2$  [N/m].
- Es soll  $\varphi(t)$  ausgegeben werden (Scope und virtuelles Modell).



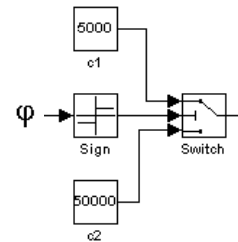
## Modellierung physikalischer Systeme

### Modellierung:

- a) Zeichnen Sie das Freikörperbild des Pendels mit allen angreifenden Kräften.  
Legen Sie die positive Richtung für  $\varphi$ ,  $\dot{\varphi}$ ,  $\ddot{\varphi}$  fest.



- b) Geben Sie die auf das Pendel wirkenden Momente bezüglich des Drehpunktes an.  
c) Geben Sie das Massenträgheitsmoment des Pendels an (Pendelmasse hat Quaderform) → Satz von Steiner.  
d) Leiten Sie die Bewegungs-DGL aus dem Drallsatz ab.  
e) Modellieren Sie das System mit Matlab-Simulink.  
Hinweis: Die Federkraftumschaltung lässt sich wie folgt realisieren:



### Versuchsdurchführung:

Beginnen Sie mit den folgenden Parametern.

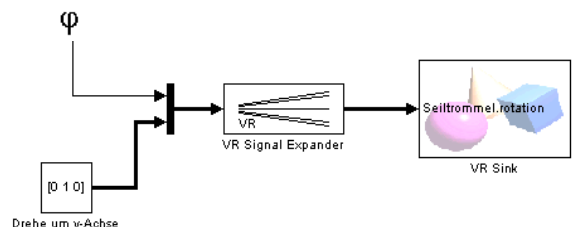
- a) Voreinstellungen:  $L=1\text{m}$ ,  $m=10\text{kg}$ , Trommelradius  $r=0.3\text{m}$ ,  $w=0.3\text{m}$ ,  $h=0.2\text{m}$ .  
 $c_1=5000\text{ N/m}$ ,  $c_2=500000\text{N/m}$   
 $\varphi_0=45^\circ$ .

Simulieren Sie das System und zeigen Sie  $\varphi(t)$  mit dem Scope an.  
Kontrollieren Sie die Genauigkeit der Schaltzeitpunkte der Federkraftumschaltung.

- b) Visualisieren Sie das pendelnde System mit einer VR-Sink (virtual reality toolbox).

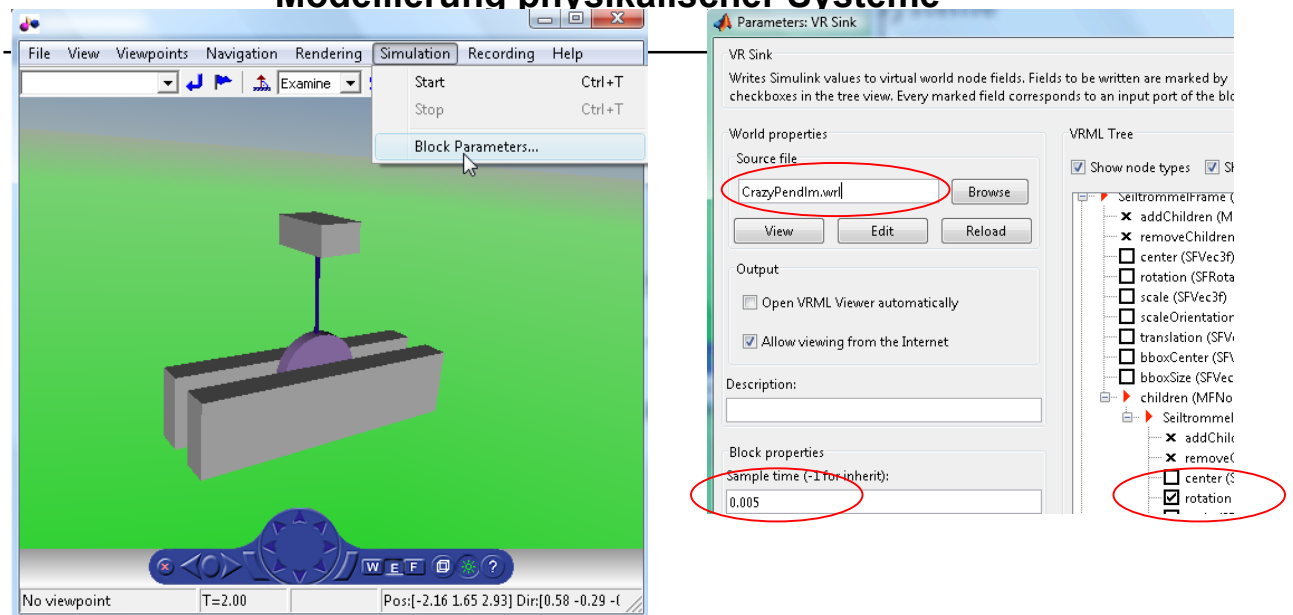
Die WRL-Datei ist bereits vorhanden und liegt im Pub-Verzeichnis (CrazyPendIm.wrl).

Über den Signalexpander wird der Drehwinkel  $\varphi$  und die Drehachse  $[0\ 1\ 0]$  (=y-Achse) auf die VR-Sink-Eingänge  $[4\ 1\ 2\ 3]$  gelegt.

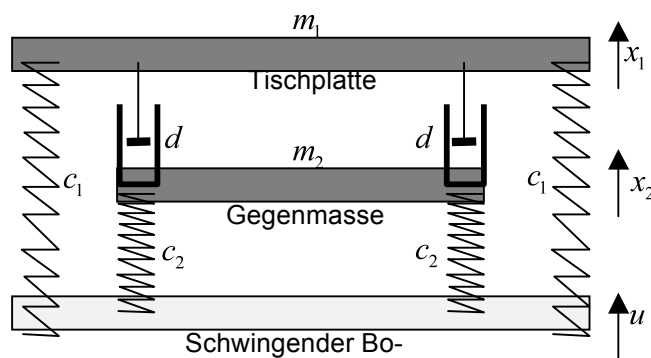


Die Sample-Time der VR-Sink wird auf 0.005 gestellt.

## Modellierung physikalischer Systeme



### 3. Schwingungsgedämpfter Tisch



#### Aufgabenstellung:

Eine Tischplatte soll schwingungsgedämpft gelagert werden. Dafür wird die nebenstehende Konstruktion verwendet.

Über den Boden kann eine Vertikalauslenkung  $u$  auf den Tisch wirken.

Die Gegenmasse  $m_2$  und die Tischplatte  $m_1$  reagieren darauf mit den Auslenkungen  $x_2$  und  $x_1$ .

Das System ist zu simulieren.

#### Simulationsrandbedingungen:

- Als Parameter sollen vorgebar sein :  $m_1$ ,  $m_2$ ,  $c_1$ ,  $c_2$ ,  $d$
- Die Auslenkung der Tischplatte  $x_1(t)$  soll mit dem Scope dargestellt werden.

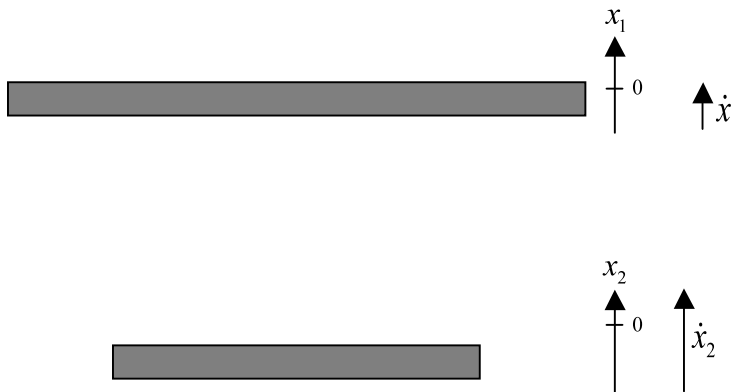
**Modellierung physikalischer Systeme**

---

**Modellierung:**

- a) Zeichnen Sie das Freikörperbild der beiden Massen mit allen angreifenden Kräften.  
Legen Sie die positive Richtung für  $x_1, \dot{x}_1, x_2, \dot{x}_2$  fest.

Hinweis: Zur Festlegung der Kraftrichtung stellen Sie sich einen bestimmten Systemzustand vor, z.B.:  $x_1 < 0$ ,  $x_2 < x_1$  und  $\dot{x}_2 > \dot{x}_1$



- b) Leiten Sie die beiden Bewegungs-DGLn für  $m_1$  und  $m_2$  aus dem 'Schwerpunktsatz ab ( $\Sigma F=ma$ ).
- c) Bestimmen Sie die Ruhelagen der beiden Massen.  
Hinweis: In der Ruhelage findet keine Bewegung mehr statt, d.h. alle Ableitungen sind 0.  
Die Ruhelagen sind die Anfangswerte der Positionen.

**Versuchsdurchführung:**

Simulieren Sie das System und zeigen Sie  $x_1(t)$  mit dem Scope an. Verwenden Sie folgende Parameter:

Voreinstellungen:  $c_1=15000$  N/m,  $c_2=10000$  N/m,  $d=600$ Ns/m  
 $m_1=60$ kg,  $m_2=450$  kg

Geben Sie einen Sprung von  $u=1$ mm auf den Tisch.

**Vorführung der Matlab / Simulink-Versuche:**

Im Praktikum während des jeweiligen Termins der Praktikumsgruppe.

**Abgabe aller Ausarbeitungen:**

23. November 2011 per E-Mail an mich. Es reicht, die pdf-Datei der Ausarbeitung zu schicken, wenn die Simulationen im Praktikum erfolgreich abgegeben wurden.