

Projektarbeit: Doppelt inverses Pendel

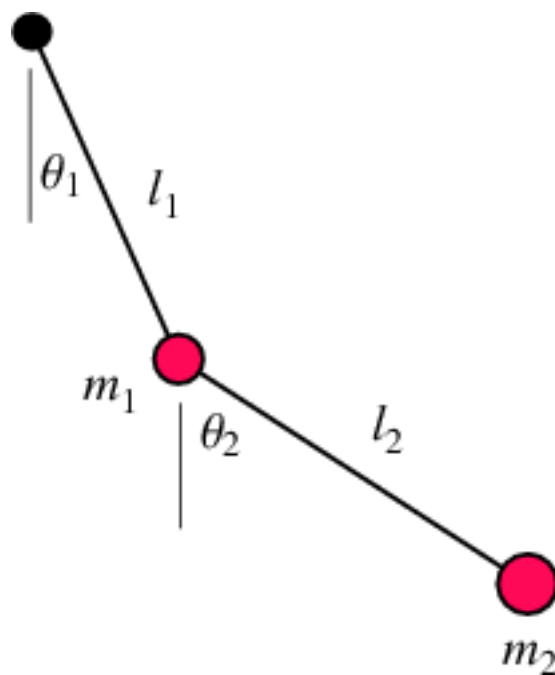


Bild: scienceworld.wolfram

Michael Höhn
hoehnmic@students.zhaw.ch

Stefan Hauenstein
hauenste@students.zhaw.ch

30. Mai 2012

Zusammenfassung

Bei dieser Projektarbeit handelt es sich um die programmiertechnische Umsetzung des Chaotischen Systems eines inversen Doppelpendels in ActionScript 3.

Inhaltsverzeichnis

1	Projektbeschreibung	4
1.1	Danksagung	4
1.2	Aufgabenstellung durch den Experten	4
1.3	Motivation	4
1.4	Iterationsplan	5
1.5	Beteiligte Personen	5
2	Anforderungsanalyse	6
2.1	Zielbestimmung	6
2.2	Zielbestimmung	6
2.3	Produktfunktionen	6
3	Umsetzung	7
3.1	Verwendete Technologien	7
3.1.1	ActionScript 3	7
3.1.2	Numerisches Verfahren	7
3.1.2.1	Runge-Kutta Verfahren 4. Ordnung	7
4	Physikalische und Mathematische Zusammenhänge	9
4.1	Physikalische Zusammenhänge	9
4.1.1	Kinematik des Doppelpendels	9
4.1.2	Kräfte im Doppelpendel	10
4.1.3	Umformen der Gleichungen	12
4.2	Numerische Differentialgleichung	12
4.3	Numerisches Verfahren zur Berechnung der Lösung	13
4.3.1	Runge-Kutta Algorithmus	13
5	Software	14
5.1	Grafische Oberfläche	14

1 Projektbeschreibung

1.1 Danksagung

Wir möchten an dieser Stelle Herrn Georg Brügger und Herrn Lukas Eppler danken, ohne ihre wertvolle Unterstützung und Anregungen wäre dieses Software Projekt nur schwer umsetzbar gewesen.

1.2 Aufgabenstellung durch den Experten

Das Schwingverhalten eines idealen „Doppelt Inversen Pendel“ soll untersucht und durch geeignete Physikalische und Mathematische Beziehungen nach gebildet werden. Physikalische Effekte wie Reibung und Luftwiderstand werden vernachlässigt, Abmessungen und Materialien müssen als Eingabegrößen variabel gestaltet werden.

Der Vorgang wird in „Echtzeit“ am Bildschirm des PC's Graphisch dargestellt visualisiert.

Zusätzliche Angaben des Verhaltens können auf der gleichen Bildschirmseite eingeblendet werden.

1.3 Motivation

Physikalische Eigenschaften mit einer Objekt Orientierten Programmiersprache umzusetzen reizte uns schon immer. Die Aufgabenstellung ermöglichte uns das theoretisch gelernte aus dem Studienfach Numerik in Verbindung mit realen Physikalischen Grundlagen in einer praktischen Anwendung umzusetzen.

Auch ermöglichte es die algebraischen Zusammenhänge aus vergangenen Vorlesungen besser zu verstehen und Lücken aufzufüllen.

1.4 Iterationsplan

Iteration	Termin	Task
1	Mi 04.04.2012	<ul style="list-style-type: none">• Physikalische Eigenschaften des Doppelpendels aufstellen• Evaluierung der Programmiersprache• Gleichungssystem aufstellen
2	Fr 27.04.2012	<ul style="list-style-type: none">• Grafische Benutzeroberfläche• Programmierung des ersten Pendels• Programmierung des zweiten Pendels• Runge Kutta definieren
3	Mi 23.05.2012	<ul style="list-style-type: none">• Pendel einstellen• Projektdokumentation• Präsentation
4	Präsentationstermin	<ul style="list-style-type: none">• Präsentation• Projektdokumentation

1.5 Beteiligte Personen

Funktion	Name
Projekt Team	Stefan Hauenstein Michael Höhn
Experte	Georg Brügger
Scrum Master	Lukas Eppler

2 Anforderungsanalyse

2.1 Zielbestimmung

Es soll ein Programm entwickelt werden das die physikalischen und chaotischen Eigenschaften des inversen Doppelpendels grafisch darstellt.

Die einzelnen Segmente des Pendels sollen durch Benutzereingaben über Tastatur und/oder Maus einstellbar sein.

2.2 Zielbestimmung

- Die Umgebung des Doppelpendels soll als ideal gelten.
- Das Ausgangsmaterial wird als Aluminium Stab definiert (Dichte $2,7[g/cm^3]$)
- Die Gravitation wird mit $9.81[m/s^2]$ definiert
- Wegen der chaotischen Eigenschaft des Pendels können keine aussagenkräftigen Tests implementiert werden.

2.3 Produktfunktionen

- Der Benutzer kann eine Konfiguration als XML-File mit der Erweiterung DIP einlesen.
- Der Benutzer kann die einzelnen Glieder des Pendels mit der Maus positionieren.
- Während das Pendel in Bewegung ist werden Winkel und Geschwindigkeit angezeigt.
- Der Benutzer kann den Ablauf starten, jederzeit stoppen und zurückstellen.

3 Umsetzung

3.1 Verwendete Technologien

3.1.1 ActionScript 3

Das Programm wird mittels der Objektorientierten Sprache ActionScript 3 (AS3) und dem Flex SDK in der Eclipse Entwicklungsumgebung programmiert. Das Programm ist dank dem Flashplayer auf fast jedem Betriebssystem und Browser lauffähig. Die Implementierung mittels Flex SDK und AS3 ist speziell empfehlenswert, da das Flex Framework speziell für die Animation ausgelegt ist und es dadurch zu einer grossen Zeiteinsparung bei der Entwicklung kommt.

3.1.2 Numerisches Verfahren

Um die Fehlerkomultierten Abweichungen der Berechnung auf möglichst lange Zeit klein zu halten wird das Numerische Verfahren von Runge-Kutta in der 4. Ordnung verwendet. Der Rechenaufwand für dieses Verfahren ist höher als bei anderen Verfahren wie z.B. dem Euler-Verfahren. Seine Genauigkeit auf grosse Zeitintervalle gleichen diesen Mehraufwand wieder aus.

3.1.2.1 Runge-Kutta Verfahren 4. Ordnung

Das klassische Runge-Kutta-Verfahren (nach Carl Runge und Martin Wilhelm Kutta) ist ein spezielles explizites 4-stufiges Runge-Kutta-Verfahren zur numerischen Lösung von Anfangswertproblemen. Runge hat als erster (1895) ein mehrstufiges Verfahren angegeben und Kutta die allgemeine Form expliziter s-stufiger Verfahren.

Das klassische Runge-Kutta-Verfahren verwendet den Ansatz, Ableitungen durch Differenzenquotienten zu approximieren. Die dabei bei nichtlinearen Funktionen notwendigerweise auftretenden können durch geeignete Kombinationen verschiedener Differenzenquotienten reduziert werden. Das Runge-Kutta-Verfahren 4. Ordnung ist eine solche Kombination, die Diskretisierungsfehler bis zur dritten Ableitung kompensiert.

$$y(t)' = f(t, y(t)) \quad \text{mit} \quad y(0) = y_0$$

$$y^{k+1} = y^k + h \left(\frac{1}{6}k_1 + \frac{2}{6}k_2 + \frac{2}{6}k_3 + \frac{1}{6}k_4 \right)$$

$$\text{mit} \quad k_1 = f(t_k, y^k)$$

$$k_2 = f\left(t_k + \frac{h}{2}, y^k + \frac{h}{2}k_1\right)$$

$$k_3 = f\left(t_k + \frac{h}{2}, y^k + \frac{h}{2}k_2\right)$$

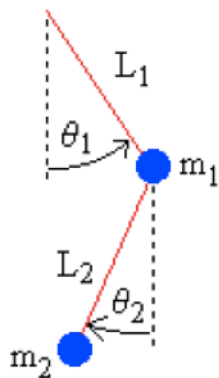
$$k_4 = f(t_k + h, y^k + hk_3)$$

4 Physikalische und Mathematische Zusammenhänge

4.1 Physikalische Zusammenhänge

Um die Gleichungen des Ausgangsproblems zu definieren, muss man sich erst mit den physikalischen Eigenschaften eines Pendels beschäftigen. Kräfte welche aus der Bewegung entstehen kombinieren sich mit Statischen. Kräfte des ersten Pendels beeinflussen die des Zweiten. Im Folgenden werden die Kinematik und die Kräfte des Doppelpendels zum gesuchten Ausgangsproblem kombiniert.

4.1.1 Kinematik des Doppelpendels



x = Horizontale Position des Massenschwerpunkts
 y = Vertikale Position des Massenschwerpunkts
 θ = Winkel
 L = Pendellänge

Abbildung 1: Energie im Pendel

Ausgangslage für die Berechnung ist das obere Pendel. Das erste Pendel wird als Subskript 1 definiert mit x_1 und y_1 . Äquivalent dazu wird das zweite Pendel als Subskript 2 definiert mit x_2 und y_2 .

Als erstes werden die einfachen trigonometrischen Gleichungen aufgestellt:

$$x_1 = L_1 \sin \theta_1 \quad (4.1.1)$$

$$y_1 = -L_1 \cos \theta_1 \quad (4.1.2)$$

$$x_2 = x_1 + L_2 \sin \theta_2 \quad (4.1.3)$$

$$y_2 = y_1 - L_2 \cos \theta_2 \quad (4.1.4)$$

Aus den trigonometrischen Gleichungen werden die erste und zweite Ableitung gebildet.

Erste Ableitung (Geschwindigkeit):

$$\dot{x}_1 = \dot{\theta}_1 L_1 \cos \theta_1 \quad (4.1.5)$$

$$\dot{y}_1 = \dot{\theta}_1 L_1 \sin \theta_1 \quad (4.1.6)$$

$$\dot{x}_2 = \dot{x}_1 + \dot{\theta}_2 L_2 \cos \theta_2 \quad (4.1.7)$$

$$\dot{y}_2 = \dot{y}_1 + \dot{\theta}_2 L_2 \sin \theta_2 \quad (4.1.8)$$

Zweite Ableitung (Beschleunigung):

$$\ddot{x}_1 = -\dot{\theta}_1^2 L_1 \sin \theta_1 + \ddot{\theta}_1 L_1 \cos \theta_1 \quad (4.1.9)$$

$$\ddot{y}_1 = \dot{\theta}_1^2 L_1 \cos \theta_1 + \ddot{\theta}_1 L_1 \sin \theta_1 \quad (4.1.10)$$

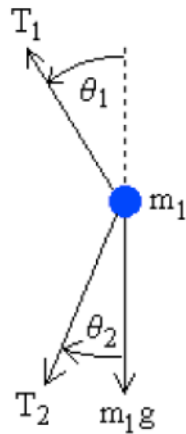
$$\ddot{x}_2 = \ddot{x}_1 - \dot{\theta}_2^2 L_2 \sin \theta_2 + \ddot{\theta}_2 L_2 \cos \theta_2 \quad (4.1.11)$$

$$\ddot{y}_2 = \ddot{y}_1 + \dot{\theta}_2^2 L_2 \cos \theta_2 + \ddot{\theta}_2 L_2 \sin \theta_2 \quad (4.1.12)$$

4.1.2 Kräfte im Doppelpendel

Die Kräfte werden pro Pendel berechnet und in einzelne Gleichungen gesetzt. Auf Grundlage des zweiten newtonschen Gesetzes ($F = m \cdot a$) wurden folgende Gleichungen für das Pendel erstellt:

Oberes Pendel:



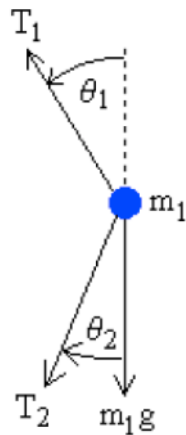
T_1 = Kraft im oberen Pendel
 m_1 = Masse des oberen Pendels
 g = Gravitations Konstante

Abbildung 2: Kräfte im oberen Pendel

$$m_1 x_1'' = -T_1 \sin \theta_1 + T_2 \sin \theta_2 \quad (4.1.13)$$

$$m_1 y_1'' = T_1 \cos \theta_1 + T_2 \cos \theta_2 - m_1 g \quad (4.1.14)$$

Unteres Pendel:



T_2 = Kraft im unteren Pendel
 m_2 = Masse des unteren Pendels
 g = Gravitations Konstante

Abbildung 3: Kräfte im unteren Pendel

$$m_2 x_2'' = -T_2 \sin \theta_2 \quad (4.1.15)$$

$$m_2 y_2'' = T_2 \cos \theta_2 - m_2 g \quad (4.1.16)$$

4.1.3 Umformen der Gleichungen

Um zu der entgültigen Differenziellen Gleichung für die Berechnung zu gelangen müssen die einzelnen Gleichungen algebraisch umgeformt werden. Angefangen bei den Kräftegleichungen. Gleichung (4.1.15) und (4.1.16) werden nach $T_2 \sin \theta_2$ bzw. nach $T_2 \cos \theta_2$ aufgelöst und in die Gleichungen (4.1.13) und (4.1.14) eingesetzt.

(4.17) wird mit $\cos \theta_1$ und (4.18) mit $\sin \theta_1$ erweitert und nach $T_1 \sin \theta_1 \cos \theta_1$ aufgelöst.

Daraus ergibt sich die erste Teil der Bewegungsgleichung.

Um den zweiten Teil der Gleichung zu erhalten wird die Gleichung (4.15) mit $\cos \theta_2$ und die (4.16) mit $\sin \theta_2$ erweitert.

Der zweite Teil der Gleichung heisst somit

x_2'', x_2'', y_1'' und y_2'' werden durch die Gleichungen (4.9 – 4.12) ersetzt und mit Hilfe des Lagrange-Formalismus nach θ_1'' und θ_2'' aufgelöst.

Die endgültige Bewegungsgleichung für das Doppelpendel lautet somit:

$$\theta_1'' = \frac{-g(2m_1 + m_2) \sin \theta_1 - m_2 g \sin(\theta_1 - 2\theta_2) - 2 \sin(\theta_1 - \theta_2) m_2 (\theta_2'^2 L_2 + \theta_1'^2 L_1 \cos(\theta_1 - \theta_2))}{L_1(2m_1 + m_2 - m_2 \cos(2\theta_1 - 2\theta_2))} \quad (4.1.17)$$

$$\theta_2'' = \frac{2 \sin(\theta_1 - \theta_2) (\theta_1'^2 L_1 (m_1 + m_2) + g(m_1 + m_2) \cos \theta_1 + \theta_2'^2 L_2 m_2 \cos(\theta_1 - \theta_2))}{L_2(2m_1 + m_2 - m_2 \cos(2\theta_1 - 2\theta_2))} \quad (4.1.18)$$

4.2 Numerische Differentialgleichung

Die obige Bewegungsgleichung lässt sich aber noch nicht mittels des Runge-Kutta Algorithmus numerisch annähern. Um dies zu ermöglichen muss bedacht werden, dass die erste Ableitung von θ die Winkelgeschwindigkeit ω ist.

Somit ist: $\theta_1' = \omega_1$ und $\theta_2' = \omega_2$

und damit erhält man für den Runge-Kutta Algorithmus nun folgende Schlussgleichun-

gen:

$$\theta'_1 = \omega_1 \quad (4.2.1)$$

$$\theta'_2 = \omega_2 \quad (4.2.2)$$

$$\omega'_1 = \frac{-g(2m_1 + m_2) \sin \theta_1 - m_2 g \sin(\theta_1 - 2\theta_2) - 2 \sin(\theta_1 - \theta_2) m_2 (\omega_2^2 L_2 + \omega_1^2 L_1 \cos(\theta_1 - \theta_2))}{L_1(2m_1 + m_2 - m_2 \cos(2\theta_1 - 2\theta_2))} \quad (4.2.3)$$

$$\omega'_2 = \frac{2 \sin(\theta_1 - \theta_2) (\omega_1^2 L_1 (m_1 + m_2) + g(m_1 + m_2) \cos \theta_1 + \omega_2^2 L_2 m_2 \cos(\theta_1 - \theta_2))}{L_2(2m_1 + m_2 - m_2 \cos(2\theta_1 - 2\theta_2))} \quad (4.2.4)$$

4.3 Numerisches Verfahren zur Berechnung der Lösung

4.3.1 Runge-Kutta Algorithmus

Mittels des Runge-Kutta Verfahren kann eine Differenzialgleichung Numerisch approximiert werden. Das Verfahren eignet sich besonders gut für Berechnungen in einem Programm, da es sich um ein Iteratives Verfahren handelt und somit sehr einfach und Ressourcenschonend implementiert werden kann. Verwendet wird das Runge-Kutta Verfahren 4. Ordnung auch RK4 genannt.

5 Software

5.1 Grafische Oberfläche

Die Grafische Oberfläche wird als Skalierbares Modell aufgebaut. Der Vorteil in diesem Aufbau besteht darin, dass die Applikation auf jeder Bildschirmauflösung optimal dargestellt wird. Die Simulation des Pendels passt sich anhand der Fenstergrösse automatisch an.



Abbildungsverzeichnis

1	Energie im Pendel	9
2	Kräfte im oberen Pendel	11
3	Kräfte im unteren Pendel	11

Tabellenverzeichnis

Literatur