Einführung in die Numerische Mathematik

Sommersemester 2013
Prof. Dr. Sven Beuchler
Daniel Wissel

universitätbonn

Übungsblatt 1.

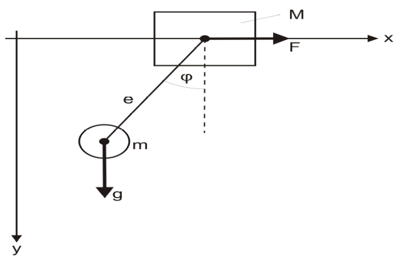
Abgabe am **Dienstag, 16.04.2013**

Aufgabe 1. (Modellierung von Optimierungsproblemen) (3 Punkte)

Vier Ziegeleien haben fünf Baustellen mit Ziegelsteinen zu beliefern. Die Kapazitäten Z_i der Ziegeleien, der Bedarf B_j der Baustellen sowie die Entfernungen d_{ij} zwischen den einzelnen Ziegeleien und Baustellen sei bekannt. Die Transportkosten seien dabei proportional zu den Entfernungen und den transportierten Mengen. Formulieren Sie ein Transportprogramm, welches die Gesamtkosten minimiert.

Aufgabe 2. (Modellierung von Optimierungsproblemen) (4 Punkte)

Eine Laufkatze ist ein entlang eines Trägers bewegliches Kranbauteil. Wir betrachten ein idealisiertes Modell (siehe Abb. rechts) einer solchen Laufkatze der Masse M, welche ein an einem Seil befestigtes Lastobjekt der Masse m transportiert. Dabei sollen für den Ort (x_M, y_M) ($y_M = \text{const.} = 0$) der Laufkatze sowie der Last (x_m, y_m) folgende **Bewegungsgleichungen** (reibungsfrei) gelten:



$$M\ddot{x}_M = F - K\sin\phi, \qquad m\ddot{x}_m = K\sin\phi, \qquad m\ddot{y}_m = mg - K\cos\phi, \tag{1}$$

wobei sich folgende geometrische Beziehungen ergeben:

$$e^2 = y_m^2 + (x_M - x_m)^2, \qquad x_m = x_M - e\sin\phi, \qquad y_m = e\cos\phi. \tag{2}$$

Dabei ist F die steuerbare Antriebskraft der Laufkatze, K die (unbekannte) Seilkraft und g die Konstante der Erdbeschleunigung.

a) Eliminieren Sie mit Hilfe der Gleichungen (2) die Lastkoordinaten x_m, y_m und die Seilkraft K aus dem dynamischen System (1), um folgendes nichtlineares ODE-System zu erhalten

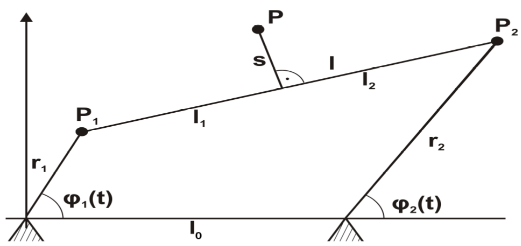
$$(m+M)\ddot{x}_M - me\left(\cos\phi\ddot{\phi} - \sin\phi(\dot{\phi})^2\right) = F \tag{3}$$

$$\cos\phi \cdot \ddot{x}_M - e\ddot{\phi} - g\sin\phi = 0 \tag{4}$$

b) Linearisieren Sie dieses System für kleine Winkel ϕ (benutzen Sie dabei: $\sin\phi \approx \phi, \cos\phi \approx 1, (\dot{\phi})^2 \approx 0$).

Aufgabe 3. (Modellierung von Optimierungsproblemen) (4 Punkte)

Ein Koppelgetriebe setzt eine Drehbewegung in eine geradlinige oder schwingende Bewegung um (oder umgekehrt). Wir betrachten ein idealisiertes Modell (siehe Abb. rechts) eines Koppelgetriebes mit folgenden Daten:



- Längen $l_0, l_1, l_2, r_1, r_2, s$, Drehgeschwindigkeit (konstant) ω_0
- vorgegebene Sollkurve $\hat{S}(t) = \begin{pmatrix} x_s(t) \\ y_s(t) \end{pmatrix}, \quad t \in \left[0, \frac{2\pi}{\omega_0}\right]$

Durch Wahl von Kontrollpunkten $t_i = \frac{i2\pi}{N\omega_0}, i = 1, \ldots, N$ wird die Sollkurve diskretisiert:

$$\hat{S} = (x_s(t_1), y_s(t_1), x_s(t_2), y_s(t_2), \ldots, x_s(t_N), y_s(t_N))^T \in \mathbb{R}^{2N}.$$

Ziel der Optimierung ist nun die Auslegung der Abmessungen, so dass die Bewegung $P(t)$ des Punktes P die Sollkurve bzgl. der Summe der Fehlerquadrate möglichst gut approximiert. Formulieren Sie das zugehörige mathematische Optimierungsproblem mit entsprechenden Nebenbedingungen. Beachten Sie hierzu die geometrischen Restriktionen des Koppelmodells (z.B. Umlauffähigkeit).

Aufgabe 4. (Satz von Carathéodory) (6 Punkte)

Beweisen Sie den *Satz von Carathéodory*: Jeder Vektor x aus der konvexen Hülle einer Menge $X \subset \mathbb{R}^d$ lässt sich als Konvexkombination von höchstens $d+1$ Elementen aus X darstellen.
Hinweis: Nach Lemma 1.5 der Vorlesung existiert eine Darstellung $x = \sum_{i=1}^m \lambda_i x_i$ für ein $m \in \mathbb{N}$. Zeigen Sie, dass sich für $m \geq d+2$ diese Darstellung um ein Element auf eine Summe von $m-1$ Elementen reduzieren lässt.

Aufgabe 5. (Jensensche Ungleichung) (4 Punkte)

Zeigen Sie die folgende Aussage: Es sei $f : X \to \mathbb{R}$ strikt konvex. Dann gilt für alle $\lambda_i \geq 0, x_i \in X, i = 1, \ldots, m$ mit $\sum_{i=1}^m \lambda_i = 1$ stets: $\quad f\left(\sum_{i=1}^m \lambda_i x_i\right) \leq \sum_{i=1}^m \lambda_i f(x_i)$.

Programmieraufgabe 1. (LAPACK-Routinen in C) (10 Punkte)

BLAS (Basic Linear Algebra Subprograms, `http://www.netlib.org/blas/`) ist eine Programmierschnittstelle (API) für Softwarebibiliotheken aus dem Bereich der linearen Algebra. **LAPACK** (Linear Algebra PACKage, `http://www.netlib.org/lapack/`, siehe auch *LAPACK Users' Guide, Third Edition, SIAM*) ist eine auf BLAS aufbauende Bibliothek, welche numerische Methoden für grundlegende Probleme der linearen Algebra bereitstellt. LAPACK ist in Fortran90 programmiert, es existiert jedoch eine einfache Sprachanbindung an C. In späteren Programmierübungen werden wir den Einsatz diverser LAPACK-Routinen aus C-Code heraus benötigen.

a) Machen Sie sich mit den Grundlagen von LAPACK vertraut, insbesondere mit dem Namensschema der Standard-Routinen und der Verwendung dieser Routinen in C-Code.

b) Machen Sie sich mit den Grundlagen von Fortran vertraut. Wie werden mehrdimensionale Arrays gespeichert? Was ist die "leading dimension"?

c) Schreiben Sie ein Programm in C oder C++, welches mittels LAPACK-Routinen das System $Ax = b$ löst und überprüfen Sie Ihr Programm anhand der folgenden Werte:

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 3 & 3 \\ 1 & 5 & 9 \end{pmatrix}, \quad b = (0, 2, 8)^T.$$

Abgabe **Mo 22.04.** und **Di 23.04.** im **CIP-Pool** (`www.iam.uni-bonn.de/pcpool/`) in der Wegelerstraße. Ab Mo 15.04. hängt dort eine Terminliste für diese beiden Tage aus; bitte tragen Sie sich alleine oder in 2er Gruppen ein.

Gesamtpunktzahl: 21 + 10 Punkte