



Technische Universität München

Fakultät für Mathematik



Bachelor-Arbeit

Qualitativer Vergleich diverser Methoden zur Matrixrekonstruktion im Kontext der optimalen Stabilisierung des inversen Pendels

Markus Stachl

Aufgabensteller: Prof. Dr. Oliver Junge

Betreuer: Prof. Dr. Oliver Junge

Abgabetermin: ...

Ich erkläre hiermit, dass ich die Bachelor-Arbeit selbständige und nur mit den angegebenen Hilfsmitteln angefertigt habe.

Garching, den

Summary

Bei einer in deutscher Sprache verfassten Arbeit muss eine Zusammenfassung in englischer Sprache vorangestellt werden. Dafür ist hier Platz.

Inhaltsverzeichnis

1 Einführung	1
2 Hintergrund und Motivation	2
2.1 Optimale Kontrolle	2
2.2 Berechnung der Wertefunktion	2
2.3 Beispiel: das invertierte Pendel	3
3 Niedring-Rang-Approximation einer Matrix	5
3.1 Singulärwertzerlegung (SVD)	5
3.1.1 Definition	5
3.1.2 Fehlerapproximation	6
3.2 CUR-Zerlegung	7
3.2.1 Definition	7
3.2.2 Zufällige Spaltenselektion auf Basis der Gleichverteilung	8
3.2.3 Spaltenselektion anhand des Leverage Scores	9
4 Berechnung der Residuen	12
4.1 Definition	12
4.2 Beispiel: Residuum der exakten Wertefunktion	12
5 Qualitativer Vergleich der Approximationsmethoden	14
5.1 Anwendung der Rang-k-Singulärwertzerlegung	14
5.2 Anwendung der CUR-Zerlegung auf Basis der Gleichverteilung	17

1 Einführung

Seit einigen Jahrzehnten befinden wir uns in einer Gesellschaft, in der wir freien Zugang zu aller Arten von Informationen haben. Durch Wissens-Datenbanken wie Wikipedia wird jedem der Zugriff zu Spezialwissen verschiedenster Themen ermöglicht und jeder Experte kann sein eigenes Spezialwissen inserieren und somit verbreiten.

In dem Informationszeitalter, in dem sich unsere Gesellschaft derzeit befindet, liegt das Augenmerk nicht mehr in der Informationsbeschaffung, sondern in der Speicherung dieser Informationen und der Erleichterung des Zugriffs auf diese.

Leider nimmt die Leistungsfähigkeit der Rechner nicht in dem selben Maße zu wie die Menge an Daten. Aus diesem Grund ist es notwendig, durch die Untersuchung einer finiten Teilmenge des Ganzen genügend Informationen zu sammeln, um damit Rückschlüsse auf die Gesamtheit ziehen zu können.

An dieser Stelle setzt diese Arbeit ein. Ist Information, beispielsweise zu den optimalen Kosten zur Stabilisierung eines instabilen Systems, in Form einer Matrix gegeben, so soll aus einer finiten Anzahl an Vektoren, die idealerweise viel kleiner ist als der Rang der Zielmatrix, in Kombination mit einem Algorithmus die ursprüngliche Matrix möglichst genau wiederhergestellt werden. Bevorzugt wird dabei ein Algorithmus mit geringen Berechnungskosten. Neben der gängigen Methode der Singulärwertzerlegung (kurz SVD, von engl. *singular value decomposition*), welche leider einige Limitierungen durch die hohen Berechnungskosten besitzt, wird in der vorliegenden Arbeit ein relativ junges Verfahren vorgestellt, die sogenannte CUR-Zerlegung.

Bei dieser Methode zur Rekonstruktion einer Matrix wird versucht, die Zielmatrix mithilfe einer geringen Anzahl an Zeilen und Spalten derselben Matrix wiederherzustellen. Die Selektion der Zeilen bzw. Spalten wird anhand einer Wahrscheinlichkeitsverteilung vorgenommen. Ergebnisse zu zwei Arten von Verteilungen werden hier verglichen: zum Einen bei einer Gleichverteilung der einzelnen Selektionswahrscheinlichkeiten, zum Anderen die Selektion auf Basis des Leverage Scores. Abschließend werden die einzelnen Simulationsergebnisse miteinander verglichen und ein aussagekräftiges Fazit abgegeben.

2 Hintergrund und Motivation

2.1 Optimale Kontrolle

Gegeben sei das Problem der optimalen Stabilisierung eines (instabilen) diskreten Systems

$$x_{k+1} = f(x_k, u_k), \quad k = 0 \dots n \quad (1)$$

um einen Fixpunkt $0 \in X$, also $f(0, 0) = 0$. Der Fixpunkt kann o.B.d.A gewählt werden, ein anderer Fixpunkt kann gegebenenfalls in den Nullpunkt geshiftet werden. Die x_k sind aus dem Zustandsphasenraum $X \subset \mathbb{R}^n$, die u_k aus der Feedback-Menge $U \subsetneq \mathbb{R}^m$. X und U sind jeweils kompakte Teilmengen der Grundräume. Die Kosten pro Schritt sind dabei gegeben durch eine (stetige) Kostenfunktion $g : X \times U \rightarrow [0, \infty]$ mit $g(x, u) \geq 0 \ \forall x \in X, u \in U$ und $g(x, u) = 0 \Leftrightarrow x = 0, u = 0$.

Das Feedback u soll nun so gewählt werden, dass die entstehenden Kosten minimiert werden und der Fixpunkt 0 zu einem asymptotisch stabilen Fixpunkt wird. Anders ausgedrückt: ausgehend von einem Punkt $x \in X$ soll ein Kontrollvektor $u = (u_1, u_2 \dots)$ gewählt werden, sodass die Trajektorie

$$x_0(x, u) = x, \quad x_{k+1} = f(x_k(x, u), u_k), \quad k = 0, 1, \dots$$

für alle k in X verbleibt und $x_k(x, u) \rightarrow 0$ für $k \rightarrow \infty$ (vgl. (9)). Alle Punkte x , für die eine Stabilisierung möglich ist werden in der Menge

$$S = \{x \in X \mid \exists u \in U^\mathbb{N} : x_k(x, u) \rightarrow 0\}$$

zusammengefasst.

Somit ergibt sich für die gesamten Kosten zur Stabilisierung eines Punktes $x \in X$ entlang einer Trajektorie

$$J(x, u) = \sum_{k=0}^{\infty} q(x_k(x, u), u_k) \quad \in [0, \infty] \quad (2)$$

Ist x nicht stabilisierbar, d.h. es existiert kein $u \in U^\mathbb{N}$ sodass $x_k(x, u) \rightarrow 0$, so ist $J(x, u) = \infty$ und $x \in X \setminus S$.

Gesucht sind nun die minimalen Kosten zur Stabilisierung (vgl. (11)) eines Punktes x , die Kostenfunktion

$$V(x) = \inf_{u \in U^\mathbb{N}} J(x, u) \quad (3)$$

2.2 Berechnung der Wertefunktion

Im Folgenden soll die zuvor definierte optimale Wertefunktion 3 numerisch berechnet werden. Hierzu wird das *Optimalitätsprinzip nach Bellman* (vgl. (3)) zu Hilfe genommen:

$$V(x) = \inf_{u \in U^\mathbb{N}} \{q(x, u) + V(f(x, u))\} \quad (4)$$

wobei sich die optimale Steuerungssequenz ergibt aus

$$u(x) = \operatorname{argmin}_{u \in U^\mathbb{N}} \{q(x, u) + V(f(x, u))\} \quad (5)$$

Zur numerischen Lösung dieses Optimierungsproblems wird ein Ansatz aus der Graphentheorie gewählt:

Die (kompakte) Menge X wird partitioniert, d.h. es entsteht eine Menge \mathcal{P} von Teilmengen $P_i \subset X, i = 1, \dots, l$, wobei $\cup_{i=1}^l P_i = X$ und $m(P_i \cap P_j) = 0$ für $i \neq j$ (m bezeichnet hier das Lebesgue-Maß).

Auf dieser Partition wird nun ein gerichteter Graph

$$G_{\mathcal{P}} = (\mathcal{P}, E_{\mathcal{P}}), \quad E_{\mathcal{P}} = \{(P_i, P_j) \in \mathcal{P} \times \mathcal{P} \mid f(P_i, U) \cap P_j \neq \emptyset\}$$

definiert. Hierbei sind die Gewichte der Kanten $e = (P_i, P_j)$ (vgl. (11)) alle nichtnegativ und gegeben durch

$$w(e) = \min_{x \in P_i, u \in U} \{q(x, u) \mid f(x, u) \in P_j\}.$$

Bild 1 zeigt schematisch solch einen Graphen ausgehend von einem Punkt x .

Bild des Graphen

Anstatt der exakten Wertefunktion $V(x)$ (4) wird nun die approximative Wertefunktion

$V_{\mathcal{P}}(x) = \min\{w(p(x)) \mid \text{Pfad } p(x) = (e_1, \dots, e_m), e_k \in E_{\mathcal{P}}$ verbindet x mit dem Ursprung}

berechnet.

Um, ausgehend von dem bestimmten Graphen, den kürzesten Weg und damit die bestmögliche Stabilisierung bezüglich der Kosten zwischen einem Punkt x und einem Fixpunkt 0 zu bestimmen, können Standard-*shortest-path*-Algorithmen wie beispielsweise der Algorithmus von Dijkstra (vgl. (4)) verwendet werden.

Somit kann für jedes Element der Partition \mathcal{P} eine optimale Stabilisierung, d.h. bezüglich der geringsten Kosten, berechnet werden.

Konvergenz: Für alle Punkte $x \in S$, für die eine Stabilisierung möglich ist, und für den Partitionsdurchmesser $diam(\mathcal{P}) := \max_i \{diam(P_i)\} \rightarrow 0$ gilt:

$$V_{\mathcal{P}}(x) \rightarrow V(x)$$

Für einen Beweis siehe (11).

Schreibt man den resultierenden Boxplot in Matrixform, erhält man Folgendes:

Da diese Matrizen je nach Parameter ziemlich groß werden können, sollen sie approximiert werden. Nach der Darstellung eines Beispiels werden im nächsten Kapitel diverse Approximationsmethoden vorgestellt.

2.3 Beispiel: das invertierte Pendel

Als Beispiel dient das Modell des invertierten Pendels auf einem Wagen. Bei diesem Modell wird ein starres invertiertes Pendel rotationsfähig auf einem Wagen montiert. Beide

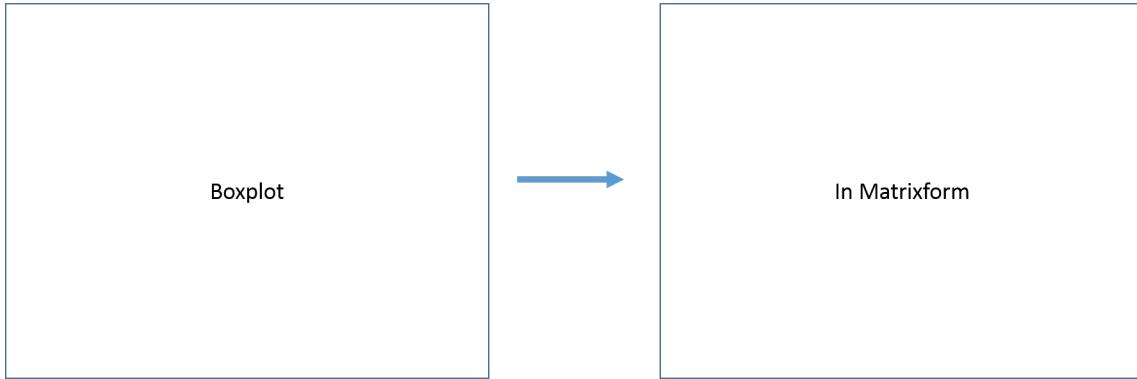


Abbildung 1: Umformung des Boxplots in Matrixform

Körper, Pendelmasse und Wagen, werden als konzentrierte Punktmassen angenommen. Zur Balancierung des Pendels können Kräfte u auf den Wagen angewandt werden um ihn in der Horizontalen zu stabilisieren. Die Dynamik des Wagens wird in diesem Modell vernachlässigt, stattdessen konzentriert man sich auf den Auslenkungswinkel φ des Pendels von der Vertikalen und auf die Winkelgeschwindigkeit $\dot{\varphi}$ im Zustandsraum $(\varphi, \dot{\varphi}) \in \mathbb{R}^2$. Das Pendel befindet sich im höchsten Punkt in einer instabilen Ruhelage. Verlässt es diese Ruhelage um einen Winkel φ , so muss dem Pendel wieder Energie hinzugefügt werden um es wieder in den Zustand der instabilen Ruhelage zurück zu versetzen. Das mathematische System zu dieser Zustandsrückführung definiert sich wie folgt (vgl. (9)):

$$\dot{x}_1 = x_2 \quad (6)$$

$$\dot{x}_2 = \frac{\frac{g}{l} \sin(x_1) - \frac{1}{2} m_r x_2^2 \sin(2x_1) - \frac{m_r}{ml} \cos(x_1) u}{\frac{4}{3} - m_r \cos^2(x_1)} \quad (7)$$

wobei $x_1 = \varphi$, $x_2 = \dot{\varphi}$, $m_r = \frac{m}{m+M}$ das Massenverhältniss zwischen Pendelmasse und Wagenmasse, $m = 2\text{kg}$, $M = 8\text{kg}$, $l = 0.5\text{m}$ Länge des Pendels und $g = 9.8\frac{\text{m}}{\text{s}^2}$ die Gravitationskonstante.

Die schrittweisen Stabilisierungskosten des inversen Pendels ergeben sich aus

$$q(x, u) = \frac{1}{2}(0.1x_1^2 + 0.05x_2^2 + 0.01u^2). \quad (8)$$

Berechnet man auf Basis dieses mathematischen Modells nun die Wertefunktion V_P wie in 2.2 definiert, so ergibt sich folgendes Bild:

Ausgehend von der Ruhelage des Pendels im Punkt $(0, 0)$ im Phasendiagramm nehmen die Kosten zur Stabilisierung des Pendelsystems zu. [hier noch etwas aus dem gaio paper eintragen](#)

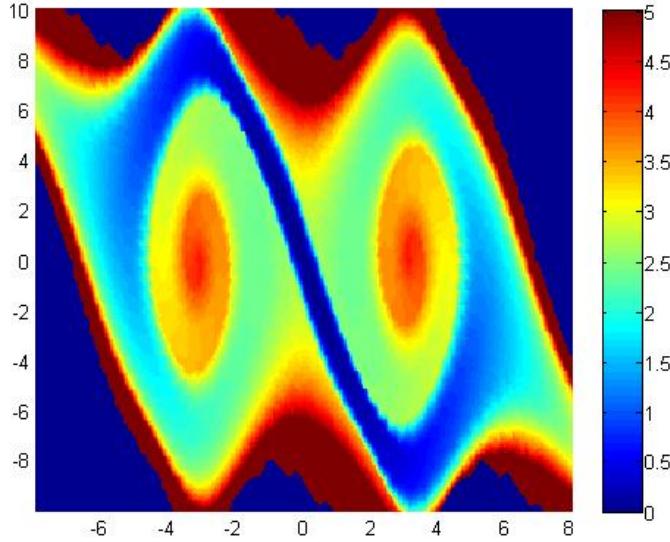


Abbildung 2: Approximierte Wertefunktion V_P bei Simulationstiefe 16

3 Niedrig-Rang-Approximation einer Matrix

Je nach Tiefe der Simulation bzw. der Dimension des zugrunde liegenden Raumes (vgl. **Fluch der Dimension** nach Bellman (1)) können gigantische Matrizen entstehen. Da die gängigen Rechner bei der Speicherung solcher Matrizen schnell an ihre Grenzen stoßen, soll nun für die Wertematrix V eine Niedrig-Rang- Approximation der Form

$$V(x) \approx \sum_{i=1}^k \sigma_i u_i(x) v_i^T(x) \quad (9)$$

gefunden werden, wobei $\sigma_i \in \mathbb{R}$ Konstanten und u_i, v_i Vektoren im \mathbb{R}^n sind. Im Folgenden werden einige Näherungen vorgestellt.

3.1 Singulärwertzerlegung (SVD)

3.1.1 Definition

Die klassische Singulärwertzerlegung ist eine Erweiterung der Diagonalisierung von reellen quadratischen Matrizen auf komplexe $\mathbb{R}^{m \times n}$ -Matrizen und gerade in der numerischen Mathematik von großem Nutzen. Sie ist folgendermaßen definiert.

Definition 3.1 Sei A eine komplexe $m \times n$ -Matrix von Rang r . Die Singulärwertzerlegung von A ist gegeben durch (vgl. (3))

$$A = U \Sigma V^*$$

wobei

- U eine unitäre $m \times m$ -Matrix ist,
- V^* die Adjungierte einer $n \times n$ -Matrix V ist und

- Σ eine reelle $m \times n$ -Diagonalmatrix der Form

$$\Sigma = \begin{pmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_r & & & \\ \hline & & & \vdots & & \\ & \vdots & & & \ddots & \\ & & 0 & \cdots & & \\ & & & \cdots & 0 & \cdots \\ & & & & & \vdots \end{pmatrix}$$

Die Einträge $\sigma_i > 0$, $i = 1 \dots r$, $\sigma_1 > \dots > \sigma_r$ heißen Singulärwerte von A . Die Spalten von U , u_i , heißen linke Singulärvektoren, die Spalten von V , v_i , heißen rechte Singulärvektoren.

Bemerkung 1 A kann auch geschrieben werden als die Summe von Rang-1-Matrizen:

$$A = \sum_{i=1}^r \sigma_i u_i v_i^T$$

Die σ_i sind dabei die Singulärwerte von A und u_i und v_i die Zeilen von U bzw. V .

Die Berechnungskosten, die für das Ausführen der Matrixzerlegung benötigt werden, betragen $\mathcal{O}(\min(mn^2, m^2n))$ (vgl. (12)).

3.1.2 Fehlerapproximation

Im Folgenden wollen wir die Wertematrix V so approximieren, dass der Fehler

$$e(D) = \|V - D\|_\xi$$

bezüglich einer Norm ξ möglichst klein wird für Matrizen D von Rang $\text{rank}(D) \leq k$. $\xi = 2$ bezeichnet hierbei die 2-Norm ($\|A\|_2 = \left(\sum_{i,j} |a_{ij}|^2\right)^{\frac{1}{2}}$), $\xi = \max$ bezeichnet die Maximumsnorm ($\|A\|_{\max} = \max_{i,j} |a_{ij}|$) und $\xi = F$ verwendet die Frobeniusnorm ($\|A\|_F = (\sigma_1^2 + \dots + \sigma_r^2)^{\frac{1}{2}}$).

Die erste Frage, die es dabei zu klären gilt, ist die nach der Existenz einer "besten" Rang-k-Approximation.

Satz 3.2 (Eckart-Young) Sei eine Matrix V von Rang r gegeben durch 3.1 und $k < r$. Die beste Rang-k-Approximation von V , also eine Matrix, die

$$\min_{\text{rank}(B) \leq k} \|V - B\|_2$$

erfüllt, ist gegeben durch

$$V_k = \sum_{i=1}^k \sigma_i u_i v_i^T$$

und

$$\|V - V_k\|_2 = \sigma_{k+1}$$

wobei σ_{k+1} der $(k+1)$ -te Singulärwert von V ist. Dies impliziert, dass für Singulärwertzerlegungen von Rang $k \geq r$ gilt: $\|V - V_k\|_2 = 0$.

Beweis 1 Für einen Beweis siehe (8).

Zusammenfassend kann über die Matrixzerlegung mithilfe der Singulärwertzerlegung folgendes gesagt werden:

Vorteil:

- Die Singulärwertzerlegung ist die bezüglich der Maximumsnorm "beste" Rang-k-Approximation einer Matrix.

Nachteile:

- Sehr hohe Kosten zum Ausführen der Singulärwertzerlegung. Die Kosten der Rang-k-Singulärwertzerlegung betragen $\mathcal{O}(kmn)$ (siehe (6)).
- Zur Approximation werden keine Einträge der Zielmatrix verwendet.

Im nächsten Abschnitt werden zwei Verfahren vorgestellt, welche zur Approximation Spalten und Zeilen der Eingabematrix verwenden.

3.2 CUR-Zerlegung

3.2.1 Definition

Bei der CUR-Zerlegung werden im Gegensatz zur CX-Zerlegung Spalten und Zeilen der zu approximierenden Matrix verwendet. Die Matrix C enthält dabei Spalten von V und Zeilen werden in R gefasst. Die Matrix U wird dann so gewählt, dass der Fehler zwischen V und $\tilde{V} = CUR$ bezüglich einer Norm (i.d.R der Frobenius- oder Spektralnorm) minimal wird.

Die Struktur der CUR-Zerlegung stellt sich wie folgt dar:

$$\underbrace{\begin{pmatrix} & \\ & V \\ & \end{pmatrix}}_{m \times n} \approx \underbrace{\begin{pmatrix} & \\ & C \\ & \end{pmatrix}}_{m \times c} \cdot \underbrace{\begin{pmatrix} & \\ & U \\ & \end{pmatrix}}_{c \times r} \cdot \underbrace{\begin{pmatrix} & \\ & R \\ & \end{pmatrix}}_{r \times n}$$

Da wir nun die Idee hinter der CUR-Zerlegung sowie deren Struktur festgelegt haben, können wir die Approximation mathematisch definieren.

Definition 3.3 (vgl. (7)) Gegeben sei eine Matrix $V \in \mathbb{R}^{m \times n}$. Sei $C \in \mathbb{R}^{m \times c}$ eine Matrix, welche aus c Spalten von V besteht, und $R \in \mathbb{R}^{r \times n}$ eine Matrix, deren r Zeilen aus Zeilen von V bestehen. Dann ist die Matrix $V' = CUR$ für jede beliebige Matrix $U \in \mathbb{R}^{c \times r}$ eine Spalten-Zeilen-basierte Approximation von V , beziehungsweise eine CUR-Zerlegung von V .

Besonders kritisch bei der Berechnung der Zerlegung ist, welche Zeilen bzw. Spalten für die Approximation verwendet werden. Dafür werden im Folgenden zwei verschiedene Verfahren und die resultierenden Approximationen dargestellt.

3.2.2 Zufällige Spaltenselektion auf Basis der Gleichverteilung

Der erste intuitive Ansatz zur Selektion von Matrixindizes ist der, der auf der **Gleichverteilung** der Indexwahrscheinlichkeiten beruht. Das heißt, die Wahrscheinlichkeit, dass eine beliebige Zeile oder Spalte einer Matrix ausgewählt wird, ist für alle relevanten Matrixbestandteile identisch.

Im Falle einer $m \times n$ -Matrix ergibt sich explizit $\sigma_i = \frac{1}{m}$ für die Wahrscheinlichkeit zur Selektion der i -ten Zeile respektive $\sigma_j = \frac{1}{n}$ für die der j -ten Spalte. Man sieht sehr schön, dass die Selektions-Wahrscheinlichkeit einer Zeile oder Spalte unabhängig von dem ausgewählten Index ist.

Vorteile:

CUR-Zerlegung bei Gleichverteilung

- Input:** Matrix $V \in \mathbb{R}^{m \times n}$, Reihenanzahl $r \leq m$, Spaltenanzahl $c \leq n$
- (i) für $t = 1 \dots r$ wähle die i -te Zeile mit Wahrscheinlichkeit $\sigma_i = \frac{1}{m}$ und füge sie als Zeile t der Matrix R hinzu. Wurde diese Zeile bereits ausgewählt, wiederhole den Selektionsschritt. Erhalte so die Reihenmatrix R .
 - (ii) für $t = 1 \dots c$ wähle die j -te Spalte mit Wahrscheinlichkeit $\sigma_j = \frac{1}{n}$ und füge sie als Spalte t der Matrix C hinzu. Wurde diese Spalte bereits ausgewählt, wiederhole den Selektionsschritt. Erhalte so die Zeilenmatrix C .
 - (iii) Definiere die Matrix U als $U = C^+ A R^+$, wobei X^+ die Moore-Penrose-Inverse (13) von X ist.
- Output:** Spaltenmatrix $C \in \mathbb{R}^{m \times c}$, Zeilenmatrix $R \in \mathbb{R}^{r \times n}$ und Matrix $U \in \mathbb{R}^{c \times r}$

Tabelle 1: Der Algorithmus *randCUR* zur Bestimmung der CUR-Zerlegung auf Grundlage der gleichverteilten Wahrscheinlichkeiten für die Selektion von Spalten- bzw. Zeilen-Indizes

- Geringe Berechnungskosten $\mathcal{O}(l + r + m + n)$, da die Singulärwertzerlegung zur Indexauswahl nicht benötigt wird.
- Trotzdem relativ gute Approximationsergebnisse

Nachteile:

- Zufällige Selektion lässt keine Rückschlüsse auf den größtmöglichen Fehler zu.
- Langsamere Konvergenz (vgl. 5.2) im Vergleich zur Singulärwertzerlegung

3.2.3 Spaltenselektion anhand des Leverage Scores

Im Folgenden wird das Verfahren der Spaltenselektion anhand eines statistischen Wertes (engl.: *leverage score*) nach Mahoney und Drineas (12) dargestellt. Dabei wird ein Rating erzeugt, welche den Einfluss einer Spalte/Zeile auf die "beste" Rang-k-Approximation, also die Singulärwertzerlegung von Rank k , der Matrix beschreibt. Dieser spezifische Wert beschreibt, mit welcher Wahrscheinlichkeit eine Zeile/Spalte selektiert und zur Approximationsmatrix hinzugefügt wird.

Bestimmung des statistischen Einflusswertes einer Spalte/Zeile:

Aus 1 wissen wir, dass sich die j -te Spalte einer Zielformatrix V von Rang r darstellen lassen kann als

$$V^j = \sum_{\xi=1}^r (\sigma_\xi u^\xi) v_j^\xi$$

bzw. deren "beste" Rang-k-Approximation durch

$$V^j = \sum_{\xi=1}^k (\sigma_\xi u^\xi) v_j^\xi$$

wobei u^ξ der ξ -te linke Singulärvektor und v_j^ξ die j -te Koordinate des ξ -ten rechten Singulärvektors ist. Der **statistische Einflusswert** einer Spalte von V kann nun gemessen werden durch

$$\pi_j = \frac{1}{k} \sum_{\xi=1}^k (v_j^\xi)^2 \quad j \in [1, n] \quad (10)$$

Er gibt an, wie groß der Einfluss einer Spalte auf die durch die Singulärwertzerlegung definierte "beste" Rang-k-Approximation V_k ist.

Diese Einflusswerte bilden eine Wahrscheinlichkeitsverteilung auf den n Spalten und es gilt: $\sum_{j=1}^n \pi_j = 1$ und $\pi_j \geq 0$.

Das Selektieren der Spalten wird anschließend mithilfe des Algorithmus' COLUMNSELECT (vgl. (12)) durchgeführt:

- (i) **Input:** Matrix $V \in \mathbb{R}^{m \times n}$ und Fehlerparameter ϵ
- (ii) Berechne die ersten k rechten Singulärvektoren (v_1, \dots, v_k) von V und die statistischen Einflusswerte π_j der Spalten von V .
- (iii) Verwende die j -te Spalte von V mit Wahrscheinlichkeit $p_j = \min\{1, c\pi_j\}$, wobei $c = \mathcal{O}(k \log \frac{k}{\epsilon^2})$
- (iv) **Output:** Matrix C , welche aus den selektierten Spalten von A besteht.

Mithilfe dieses Algorithmus' entsteht eine Matrix C , die $c' \leq c$ der selektierten Spalten verwendet. Die Kosten zur Ausführung des Algorithmus' beruhen hauptsächlich auf der Berechnung der ersten k Singulärvektoren und sind daher $\mathcal{O}(mnk)$ (12).

Zur Berechnung der Matrizen C, U und R wird abschließend folgender Algorithmus ALGORITHMCUR (vgl. (12)) verwendet:

ALGORITHM CUR

- Input:** Matrix $V \in \mathbb{R}^{m \times n}$, Zielrang k , Fehlerparameter ϵ
- (i) Wende COLUMNSELECT auf A an mit $c = \mathcal{O}(k \log \frac{k}{\epsilon^2})$ und berechne somit die Matrix C
 - (ii) Wende COLUMNSELECT auf A^T an mit $r = \mathcal{O}(k \log \frac{k}{\epsilon^2})$ und berechne somit die Matrix R
 - (iii) Definiere die Matrix U als $U = C^+ A R^+$, wobei X^+ die Moore-Penrose-Inverse (13) von X ist
- Output:** Spaltenmatrix C , Zeilenmatrix R und Matrix U

Tabelle 2: Der Algorithmus ALGORITHM CUR zur Bestimmung der CUR-Zerlegung mithilfe des Leverage Scores

Die Wahrscheinlichkeit für die Auswahl einer der c Zeilen bzw. r Reihen erfolgt dabei proportional zur Einflussrate der entsprechenden Matrixeinträge.

Die mithilfe des Algorithmus' konstruierte approximative Matrix $V' = CUR$ erfüllt nun, dass

$$\|V - V'\|_F \leq (2 + \epsilon)\|V - V_k\|_F$$

mit hoher Wahrscheinlichkeit (m.h.W.) (vgl. (12)).

Die Rate $(2 + \epsilon)$ kann mit einer Variation des Algorithmus' noch weiter verbessert werden (7). Bei dieser Methode wird die Matrix R abhängig von den für die Matrix C gewählten Zeilen berechnet. Durch diese Abwandlung verringert sich der Fehler m.h.W. zu

$$\|V - CUR\|_F \leq (1 + \epsilon)\|V - V_k\|_F$$

Dies ist nur eine Möglichkeit eine approximative CUR-Zerlegung der Matrix V zu finden. Weitere werden in folgender Tabelle gezeigt:

	Anzahl Spalten c	Anzahl Reihen r	rank(U)	$\ V - CUR\ _F^2 \leq$
Drineas et al. 2003 (5)	$\frac{k}{\epsilon^2}$	$\frac{k}{\epsilon^2}$	k	$\ V - V_k\ _F^2 + \epsilon\ A\ _F^2$
Drineas et al. 2008 (7)	$\frac{k \log k}{\epsilon^2}$	$\frac{k \log k}{\epsilon^4}$	$\frac{k \log k}{\epsilon^2}$	$(1 + \epsilon)\ V - V_k\ _F^2$
Wang et al. 2013 (14)	$\frac{k}{\epsilon}$	$\frac{k}{\epsilon^2}$	$\frac{k}{\epsilon}$	$(1 + \epsilon)\ V - V_k\ _F^2$

Abbildung 3: Tabelle über verschiedene CUR-Approximationen

Vorteile:

- Zeilen und Spalten der ursprünglichen Matrix werden verwendet.
- Auch bei der Selektion von nur einer geringen Anzahl an Spalten/Zeilen ($\approx \frac{1}{10}$ aller Spalten) wird bereits annähernd ein voller Rang der Matrix \tilde{V} erreicht.

Nachteile:

- Die approximierende Matrix ist nicht mehr von der Form 9
- Die Hauptkosten dieses Verfahrens zur Auswahl der Spalten bzw. Zeilen resultieren aus der Berechnung der relativ teuren Singulärwertzerlegung → hohe Berechnungskosten

4 Berechnung der Residuen

4.1 Definition

Für die exakte Wertefunktion V gilt nach Bellmann, dass

$$\begin{aligned} V(x_k) &= \inf_{u \in U^{\mathbb{N}}} \{q(x_k, u) + V(f(x_k, u))\} \\ &= q(x_k, u_k^*) + V(f(x_k, u_k^*)) \end{aligned}$$

mit dem minimierenden u_k^* , falls das Optimum angenommen wird, wobei $V(f(x_k, u)) =: V(x_{k+1})$. Da $q(x, u) > 0$ für $x \neq 0$ gilt also insbesondere, dass $V(x_k) > V(x_{k+1})$.

Mit dieser Eigenschaft wird die Wertefunktion zu einer **Lyapunov-Funktion** (siehe (15)), da gilt:

- (i) $V(x) \geq 0$
- (ii) $V(x) = 0 \Leftrightarrow x = 0$
- (iii) $V(x_k) > V(x_{k+1})$

Leider ergibt sich bei der approximierenden CUR-Zerlegung \tilde{V} das Problem, dass die Bedingung (iii) nicht mehr notwendigerweise erfüllt ist.

Für den resultierenden Fehler berechnet man dazu das **Residuum** der Bellman-Gleichung. Dieses ist gegeben durch

$$R(x) = \tilde{V}(x) - \inf_u \{q(x, u) + \tilde{V}(f(x, u))\}. \quad (11)$$

mit

$$\tilde{u}(x) = \operatorname{argmin}_{u \in U^{\mathbb{N}}} \{q(x, u) + \tilde{V}(f(x, u))\} \quad (12)$$

Im Falle des Erfüllens der Bedingung (iii) ergibt sich

$$R(x) \geq 0 \quad \forall x \in X \quad (13)$$

da $q(x, u) \geq 0$ für alle $x \in X$.

Typischerweise ist diese Bedingung aber nicht für alle $x \in X$ erfüllt. Als alternative Formulierung soll deshalb die Menge $\bar{S} = \{x \in X | R(x) < 0\}$ bezüglich des Lebesgue-Maßes möglichst geringes Volumen besitzen.

4.2 Beispiel: Residuum der exakten Wertefunktion

Die Berechnung des Residuums soll anhand der exakten Wertefunktion verdeutlicht werden. Diese exakte Wertematrix relativ zur Partition \mathcal{P} entsteht, wie in 4 beschrieben, durch Anwendung des Optimalitätsprinzips nach Bellman. Vergleicht man dieses nun mit der Definition des Residuums (siehe 11), so liegt der Rückschluss nahe, dass das Residuum für die Wertefunktion trivial für alle Werte von x ist (also $R(x) = 0 \forall x$). Wie Abbildung 4 aber verdeutlicht, gilt dies nicht für die gesamte Wertefunktion, sondern nur

für den stabilisierbaren Bereich. Der Grund dafür liegt in der Konstruktion von V nach dem Optimalitätsprinzip:

$$V(x) = \inf_u \{q(x, u) + V(f(x, u))\}$$

Sei $x \in X \setminus S$ nun nicht stabilisierbar und $f(x, u)$ weiterhin Element in $X \setminus S$. Auch wenn $V(f(x, u)) = -1$ nach Definition wegen der nicht vorhandenen Stabilisierbarkeit,

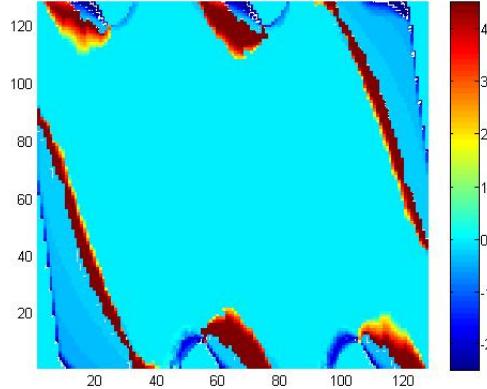


Abbildung 4: Residuum der exakten Wertefunktion bei Simulationstiefe 14

so ist aber $q(x, u)$ definiert und nicht-negativ. Aus diesem Grund ergeben sich am Rand und im nichtstabilisierbaren Bereich von X Abweichungen beim Residuum. Da in der Anwendung aber hauptsächlich der stabilisierbare Bereich von Relevanz ist, nehmen wir dieses Phänomen zur Kenntnis, ignorieren es aber in den folgenden Experimenten und konzentrieren uns auf den stabilisierbaren Bereich S .

5 Qualitativer Vergleich der Approximationsmethoden

Im Folgenden werden einige Experimente zu den verschiedenen Methoden zur Rekonstruktion der Zielmatrix durchgeführt. Als Vergleichsobjekt dient dabei die Matrix der nötigen Kosten zur Stabilisierung eines inversen Pendels. Die Ergebnisse wurden mithilfe des graphentheoretischen Ansatzes von Grüne und Junge (9) in Matlab¹ bei Simulationstiefe 14 - entspricht einer resultierenden Kostenmatrix $V \in \mathbb{R}^{128 \times 128}$ - erstellt. Die Verfahren werden in Hinblick auf Konvergenzgeschwindigkeit, Akkurate und Standort der größten Fehler verglichen.

5.1 Anwendung der Rang-k-Singulärwertzerlegung

In Kapitel 3.1 wurde bereits festgestellt, dass die Singulärwertzerlegung die beste Rang- k -Annäherung einer Matrix V bezüglich der Frobenius- bzw. Spektralnorm liefert. Leider beinhaltet dieses Faktum keine a-priori-Informationen, wie sich die Fehler in einem bestimmten Bereich der Matrix verhalten.

Aus diesem Grund werden folgend einige experimentelle Beobachtungen geliefert, die Rückschlüsse auf diese fehlenden Informationen geben sollen.

Akkurate und Konvergenz

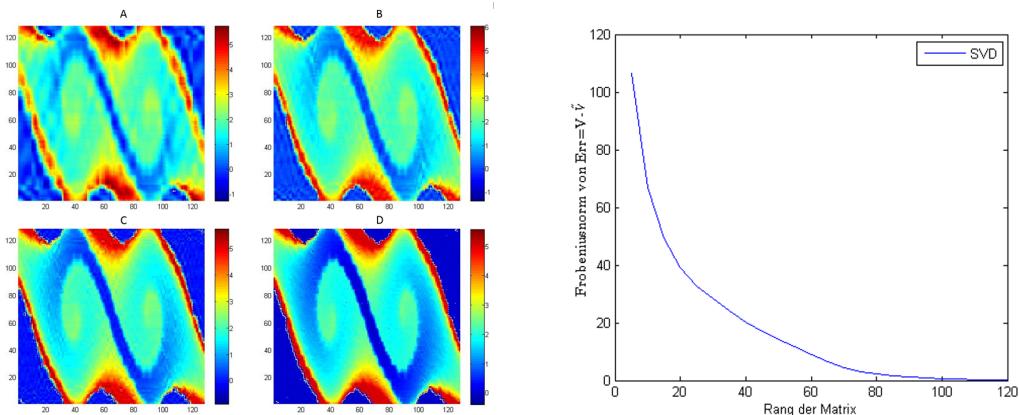


Abbildung 5: links: Boxplots der resultierenden Approximationsmatrizen bei Auswahl der 10 (A), 20 (B), 30 (C) und 60 (D) oberen Singulärvektoren. rechts: resultierender Fehler der Rang-k-Singulärwertzerlegung bezüglich der Frobeniusnorm

Wie in der oberen linken Abbildung ersichtlich, ist schon bei der Auswahl der 20 - also ca. $\frac{1}{6}$ aller - oberen Singulärwerten klar die Struktur der ursprünglichen Matrix zu erkennen. Werden weitere Singulärwerte hinzugefügt, so verbessert sich das Bild zwar noch stetig, allerdings nur noch in geringerem Maße. Zieht man die 60 größten Singulärwerte in Betracht, also in etwa die Hälfte, so unterscheidet sich das Ergebnis kaum mehr von der Originalmatrix (vgl. Abbildung 2).

¹Verwendet wurde das Software-Paket GAIO[©] - Global Analysis of Invariant Objects für Matlab

Verdeutlicht wird dies auch durch den Plot des resultierenden Fehlers bezüglich der Frobeniusnorm (siehe Abbildung 5 rechts). Die mit Hilfe der Singulärwertzerlegung erhaltene Approximationsmatrix konvergiert exponentiell in der Anzahl an gewählten Singulärwerten bzw. im Rang k der Approximation gegen die Zielmatrix V .

Fehlerlokalisierung

Auch wenn wir nun wissen, dass der Fehler relativ schnell gegen Null konvergiert, so beinhaltet dies keine Informationen darüber, wo sich bei einer Rang- k -Approximation die größten Fehler befinden. In dem vorliegenden Fall sind wir vor allem an einer gewissen Genauigkeit innerhalb des stabilisierbaren Bereiches interessiert. Größere Fehler außerhalb dieses Bereiches können deshalb gegebenenfalls ignoriert werden. Wie in Abbildung

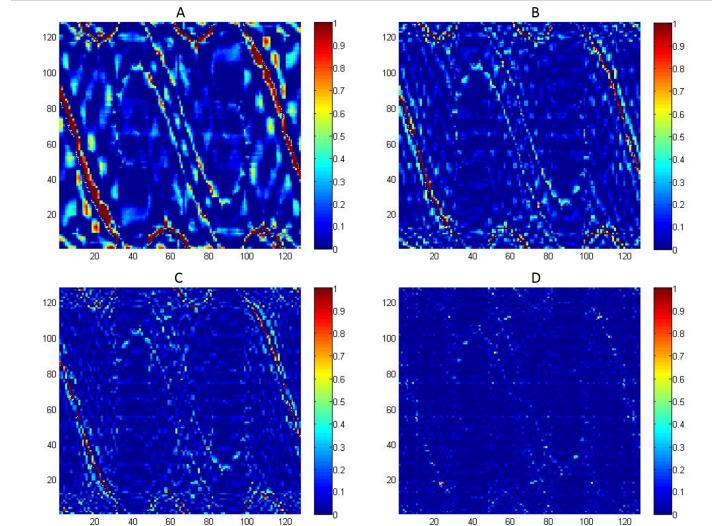


Abbildung 6: Lokalisierung der größten Fehler bei der Approximation durch die Singulärwertzerlegung von Rang 10 (A), 20 (B), 40 (C) und 60 (D)

6 zu sehen ist, befinden sich die größten Abweichungen der Approximation verglichen mit dem Original größtenteils am Rand des stabilisierbaren Bereiches. Während es bei der Singulärwertzerlegung von Rang 10 noch vereinzelt zu Abweichungen im Inneren kommt, stimmen die SVD von Rang 20 und höher im stabilisierbaren Bereich fast komplett überein und unterscheiden sich nur noch in der Größe der Abweichung am Rand des stabilisierbaren Bereiches.

Residuum

Für die positive Bewertung eines Verfahrens bezüglich des Residuums ist ausschlaggebend, wie viele Einträge der Residuenmatrix R nicht-negativ sind. Wird diese Matrix als flache Oberfläche graphisch dargestellt, so lässt sich anhand der Farben der Wert der Matrixeinträge erkennen. Bei der SVD von Rang 10 und Rang 20 sind noch sehr deutlich rote und gelbe "Flecken" sichtbar, welche in der Graphik die Stellen mit negativem Residuum symbolisieren. Erst bei Einbeziehung von mindestens 40 oberen Singulärwerten treten diese Punkte nur noch vereinzelt und mit kleinerem Volumen auf.

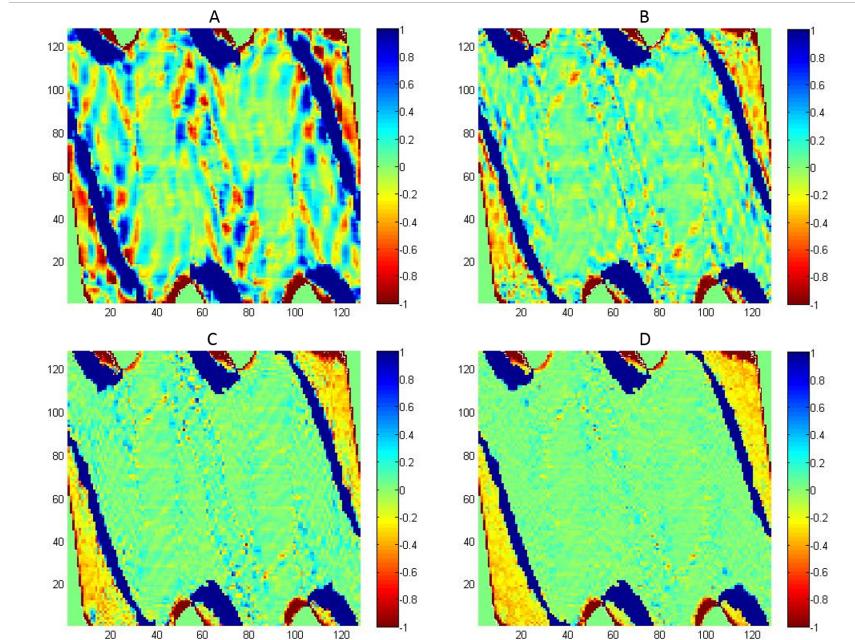


Abbildung 7: Darstellung der Residuen bei Anwendung der Singulärwertzerlegung von Rang 10 (A), 20 (B), 40 (C) und 60 (D). Zur besseren Übersicht wird das Ergebnis auf den Bereich $[-1, 1]$ eingeschränkt.

Zusammenfassung

- + schnelle Konvergenz
- + innere Struktur der Matrix bleibt auch bei geringem Approximations-Rang erhalten
- + auch bei geringem Rang befinden sich die Fehler größtenteils am Rand des stabilisierbaren Bereiches
- + geringes Volumen des negativen Residuums
- vergleichsweise sehr hohe Berechnungskosten

5.2 Anwendung der CUR-Zerlegung auf Basis der Gleichverteilung

Abkürzungen

- i.i.d unabhängig identisch verteilt
m.h.W mit hoher Wahrscheinlichkeit
SVD Singulärwertzerlegung

Literatur

- [1] BELLMAN, Richard E.: *Adaptive Control Processes*. Princeton : Princeton University Press, 1961
- [2] BODOR, Andras ; SOLYMOSI, Norbert: *rCUR: an R package for CUR matrix decomposition*. 2012
- [3] DEUFLHARD, Peter ; HOHMANN, Andreas: *Numerische Mathematik 1: Eine algorithmisch orientierte Einführung*. 4. überarb. u. erw. Aufl. Berlin u.a. : de Gruyter, 2008 (de Gruyter Lehrbuch). <http://www.reference-global.com/action/showBook?doi=10.1515/9783110203554>. – ISBN 9783110203554
- [4] DIJKSTRA, E.W.: *A Note on Two Problems in Connection with Graphs*. 1959
- [5] DRINEAS, Petros ; KANNAN, R.: Pass Efficient Algorithms for Approximating Large Matrices. In: *Proc. of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* (2003), S. 223–232
- [6] DRINEAS, Petros ; KANNAN, Ravi ; MAHONEY, Michael W.: Fast Monte Carlo algorithms for matrices III: Computing a compressed approximate matrix decomposition. In: *Society for Industrial and Applied Mathematics (SIAM)* (2006), S. 184–206
- [7] DRINEAS, Petros ; MAHONEY, Michael W. ; MUTHUKRISHNAN, S.: Relative-Error CUR Matrix Decompositions. In: *SIAM journal on matrix analysis and applications* 30 (2009), Nr. 3, S. 844–881
- [8] GOLUB, Gene H. ; VAN LOAN, Charles F.: *Matrix computations*. 4. ed. Baltimore : Johns Hopkins Univ Press, 2013 (Johns Hopkins studies in the mathematical sciences). – ISBN 9781421408590
- [9] GRÜNE, Lars ; JUNGE, Oliver: A set oriented approach to optimal feedback stabilization. In: *Systems & control letters* 54 (2005), Nr. 2, S. 169
- [10] HYVÖNEN, Saara ; MIETTINEN, Pauli ; TERZI, Evimaria: Interpretable nonnegative matrix decompositions. In: *Knowledge discovery and data mining (KDD)* (2008)
- [11] JUNGE, O. ; OSINGA, H. M.: A set oriented approach to global optimal control. In: *Control, optimisation and calculus of variations : COCV* 10 (2004) (2004), S. 259–270
- [12] MAHONEY, Michael W. ; DRINEAS, Petros: CUR matrix decompositions for improved data analysis. In: *PNAS* 2009 (2008), Nr. vol. 106, S. 697–702
- [13] PENROSE, Roger: A generalized inverse for matrices. In: *Proceedings of the Cambridge Philosophical Society* 51 (1955), S. 406–4013
- [14] WANG, Shusen ; ZANG, Zhihua: Improving CUR Matrix Decomposition and the Nyström Approximation via Adaptive Sampling. In: *Journal of Machine Learning Research* 2013 (2013), Nr. 14, S. 2729–2769

- [15] WIGGINS, Stephen: *Introduction to applied nonlinear dynamical systems and chaos.* New York : Springer-Verlag, 1990