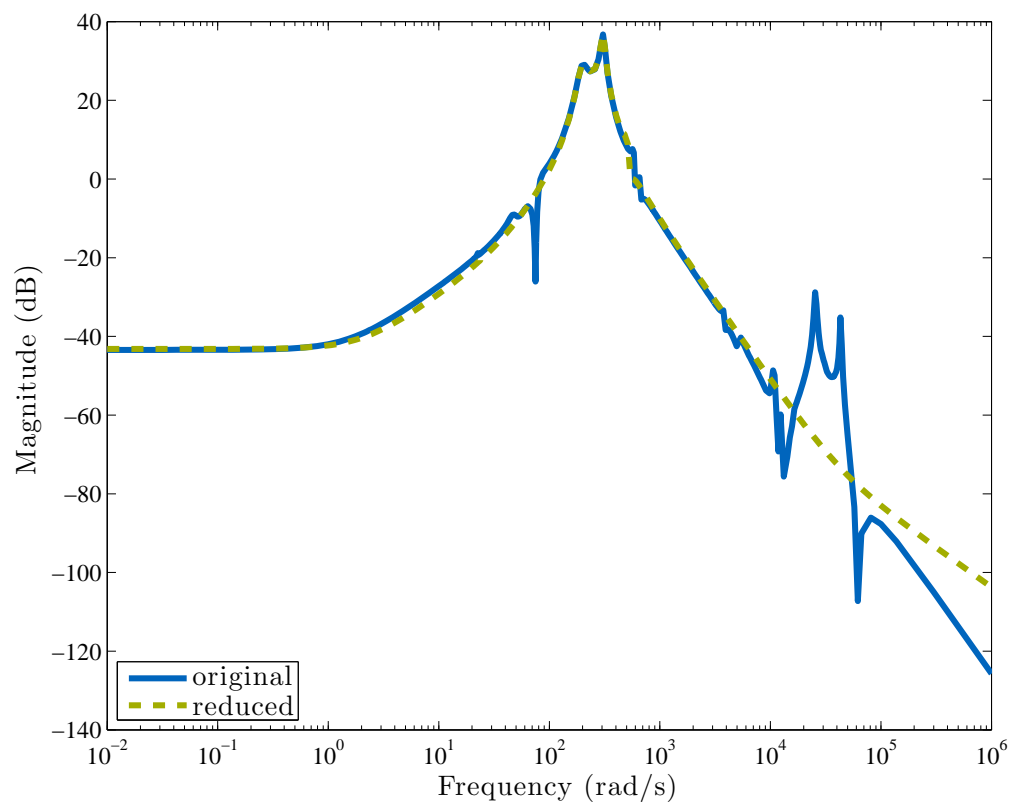


Moderne Methoden der Regelungstechnik 3

Teil B: Einführung in die Modellreduktion

Stand: 18. Oktober 2017

Dr.-Ing. Thomas Wolf
Alessandro Castagnotto, M.Sc.
Dr.-Ing. Rudy Eid
{a.castagnotto@tum.de}



*Das bewußte Reduzieren,
das Weglassen, das Vereinfachen
hat eine tiefe ethische Grundlage:
Nie kann etwas zuwider sein,
was einfach ist.*

Egon Eiermann

Kapitel 1

Einführung

1.1 Was ist Modellreduktion?

Durch *Modellreduktion* bzw. Modellordnungsreduktion, Engl.: *Model Order Reduction* (MOR), soll die Anzahl systembeschreibender Gleichungen eines dynamischen Systems signifikant reduziert werden, *ohne* das charakteristische Verhalten wesentlich zu ändern. Es handelt sich dabei um eine *Approximation*, d. h. es entsteht ein gewisser Fehler.

1.2 Welche Modelle werden reduziert?

Es gibt verschiedene Klassen mathematischer Modelle zur Beschreibung dynamischer Systeme (wodurch auch unterschiedliche Methoden der Modellreduktion nötig werden). Hier werden ein paar Klassen aufgeführt.

Lineare zeitinvariante (LZI) Systeme erster Ordnung: Im Allgemeinen resultiert die Modellierung solcher Systeme in eine *implizite* Zustandsraumdarstellung der Form

$$\begin{aligned}\mathbf{E}\dot{\mathbf{x}}(t) &= \tilde{\mathbf{A}}\mathbf{x}(t) + \tilde{\mathbf{B}}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t).\end{aligned}\tag{1.1}$$

Hier gilt: $\mathbf{E}, \tilde{\mathbf{A}} \in \mathbb{R}^{n \times n}$; $\mathbf{x}(t) \in \mathbb{R}^n$; $\tilde{\mathbf{B}} \in \mathbb{R}^{n \times m}$; $\mathbf{C} \in \mathbb{R}^{p \times n}$; $\mathbf{D} \in \mathbb{R}^{p \times m}$. Dabei ist $\mathbf{x}(t)$ der Zustandsvektor des Modells mit m Eingängen und p Ausgängen. Unter Voraussetzung, dass die *Deskriptormatrix* \mathbf{E} regulär¹ ist, kann die Zustandsdifferentialgleichung durch

¹Modelle mit singulärer \mathbf{E} Matrix werden allgemein *Deskriptorsysteme* oder *Differential-Algebraic Equations* (DAE) genannt. Diese beinhalten zusätzliche algebraische Gleichungen, wodurch die Analyse und Reduktion etwas erschwert wird. Im Literaturverzeichnis am Ende des Skripts sind einige Literaturhinweise zum Thema zu finden.

Multiplikation mit \mathbf{E}^{-1} in die bekannte *explizite* Form gebracht werden

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A} \mathbf{x}(t) + \mathbf{B} \mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C} \mathbf{x}(t) + \mathbf{D} \mathbf{u}(t).\end{aligned}\tag{1.2}$$

Diese ist theoretisch äquivalent zu (1.1) (insb.: weder das Übertragungsverhalten noch die Lösung $\mathbf{x}(t)$ werden dadurch verändert), sodass wir im Folgenden nur die explizite Darstellung untersuchen werden. Es sei an dieser Stelle jedoch bereits darauf hingewiesen, dass die implizite Darstellung (1.1) erhebliche Vorteile für die Numerik aufweist². Das „charakteristische Verhalten“ ist das Ein-/Ausgangsverhalten, sprich die Übertragungsmatrix $\mathbf{G}(s) = \mathbf{C} (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D}$ vom Eingang $\mathbf{u}(t)$ zum Ausgang $\mathbf{y}(t)$. Es gilt die Anzahl der Zustandsgleichungen zu reduzieren.

LZI-Systeme zweiter Ordnung:

$$\begin{aligned}\mathbf{M} \ddot{\mathbf{z}}(t) + \mathbf{D} \dot{\mathbf{z}}(t) + \mathbf{K} \mathbf{z}(t) &= \mathbf{g} u(t), \\ y(t) &= \mathbf{l} \mathbf{z}(t),\end{aligned}\tag{1.3}$$

mit der Massenmatrix \mathbf{M} , der Steifigkeit \mathbf{K} und der Dämpfung \mathbf{D} . Lässt sich auch als System erster Ordnung formulieren mit $\mathbf{x}(t) = [\mathbf{z}^T, \dot{\mathbf{z}}^T]^T$. Durch Reduktion direkt aus der Darstellung (1.3) lassen sich Definitheitseigenschaften der Matrizen $\mathbf{M}, \mathbf{D}, \mathbf{K}$ jedoch einfacher erhalten.

Port-Hamilton Systeme:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= (\mathbf{J} - \mathbf{R}) \nabla H(\mathbf{x}(t)) + \mathbf{b} u(t), \\ y(t) &= \mathbf{b}^T \nabla H(\mathbf{x}(t)).\end{aligned}\tag{1.4}$$

Ziel der MOR ist hier die Reduktion der Anzahl der Zustandsgleichungen und Erhaltung der Port-Hamilton Struktur.

Nichtlineare Systeme:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), u(t), t), \\ y(t) &= \mathbf{g}(\mathbf{x}(t), u(t), t).\end{aligned}\tag{1.5}$$

Hier gilt es die Anzahl der Zustandsgleichungen zu reduzieren, und/oder deren Komplexität zu minimieren.

In dieser Vorlesung beschränken wir uns auf LZI-Systeme in expliziter Darstellung mit nur einem Ein- und Ausgang ($m=p=1$), die vorgestellten Methoden lassen sich aber auf MIMO verallgemeinern.

²Eine kurze Erklärung kann im Abschnitt 7.1 nachgelesen werden

1.3 Wieso Modellreduktion?

Grundsatz: Wenn die Systemordnung zu hoch ist!

Simulation: Bei zu hoher Systemordnung wird eine Simulation zeitaufwändig (z. B. mehrere Tage), oder sogar unmöglich (da begrenzter Arbeitsspeicher).

Regelung: Die Regelung von Strecken hoher Ordnung kann folgende Nachteile haben:

- Komplexität: Der Entwurf der Regelung wird umso komplizierter/undurchsichtiger je höher die Ordnung der Strecke ist (Ausnahme: Ausgangsrückführung).
- Rechenzeit: Die Berechnung der Stellgröße könnte länger dauern als es die Abtastzeit erlaubt; dies kann von verminderter Regelgüte bis hin zu Instabilitäten führen.
- Hardware: Je höher die Ordnung eines Reglers ist, desto leistungsstärkere Hardware muss verwendet werden.

Optimierung: Für jeden Optimierungsschritt und jedes Parameter-Update muss der zu optimierende Ausgang neu berechnet werden, d. h. das dynamische System neu simuliert werden. Bei komplexen Optimierungsproblemen sind üblicherweise mehrere tausend Iterationen bis zur Konvergenz nötig. Dies kann bei Modellen hoher Ordnung zu mehrtägigen Simulationen führen um eine optimale Lösung des Problems zu finden.

1.4 Woher kommen so große Modelle?

- Örtliche Diskretisierung partieller DGLs, z.B. mittels Finite Elemente/Volumen Methode. Beispiel: Die Wärmeleitungsgleichung für die Temperatur $T(x, t)$ entlang eines eindimensionalen Stabes lautet:

$$\frac{\partial T(x, t)}{\partial t} = \frac{\partial^2 T(x, t)}{\partial^2 x} + u(x, t). \quad (1.6)$$

Eine numerische Approximation erhält man durch eine örtliche Diskretisierung, d. h. durch Auswertung der PDGL (1.6) an bestimmten Punkten (x_1, x_2, \dots, x_n) . Führt man den Zustandsvektor $\mathbf{x} = [T(x_1, t), T(x_2, t), \dots, T(x_n, t)]^T$ ein, erhält man ein LZI-System der Art (1.2).

- Aus der Modellierung technischer Systeme, welche aus einer großen Vielzahl einzelner Komponenten bestehen. Als prominenteste Beispiele seien hier integrierte Schaltkreise mit hohem Integrationsgrad (s.a. *Very Large Scale Integration*) mit hunderttausenden bis Millionen von Transistoren.

1.5 Ziele der Modellreduktion

Gegeben:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A} \mathbf{x}(t) + \mathbf{b} u(t), \\ y(t) &= \mathbf{c} \mathbf{x}(t).\end{aligned}\tag{1.7}$$

mit $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{b}, \mathbf{c}^T \in \mathbb{R}^n$ und $n \approx 10^2 \dots 10^6$.

Gesucht:

$$\begin{aligned}\dot{\mathbf{x}}_r(t) &= \mathbf{A}_r \mathbf{x}_r(t) + \mathbf{b}_r u(t), \\ y_r(t) &= \mathbf{c}_r \mathbf{x}_r(t).\end{aligned}\tag{1.8}$$

der Ordnung $q \ll n$. Ein Verfahren der Modellordnungsreduktion liefert die reduzierten Matrizen \mathbf{A}_r , \mathbf{b}_r und \mathbf{c}_r unter Berücksichtigung der folgenden Ziele:

1. **Gute Approximation:** $y(t) - y_r(t)$ respektive $G(s) - G_r(s)$ soll klein sein, eventuell auch nur in einem bestimmten (interessierenden) Frequenz-/Zeitbereich.
2. **Erhaltung von Systemeigenschaften:** Grundlegende Eigenschaften des Originalmodells (z. B. Stabilität, Passivität, Port-Hamilton Struktur, ...) sollten im reduzierten System erhalten bleiben.
3. **Numerische Effizienz:** Die Reduktionsmethode sollte numerisch möglichst effizient (schnell) und stabil sein. Zu vermeiden sind: Matrizeninversion, Lösung von Ljapunow-/Sylvester-/Riccati-Gleichungen, Eigenwertprobleme, Singulärwertprobleme, ...

Da wir uns für die Approximation des Übertragungsverhaltens interessieren, werden wir im folgenden annehmen, dass die Anfangswerte $\mathbf{x}(t=0)$ und $\mathbf{x}_r(t=0)$ im Ursprung liegen.




Hinweis: Wenn das System (1.7) einen Durchgriff d besitzt, bleibt dieser durch die Reduktion unverändert, da die Dimensionen von d nur von der Anzahl der Ein- und Ausgänge abhängt, welche durch die Reduktion nicht beeinflusst wird. Es soll nur die Anzahl n der Zustände reduziert werden! (Es existieren jedoch Methoden mit $d_r \neq d$, um zusätzliche Freiheitsgrade bei der Reduktion nutzen zu können.)

1.6 MOR Software

In der Realität vorkommenden hochdimensionale Modelle sind i.d.R. zu komplex, um per Hand reduziert zu werden. Daher wird die Modellordnungsreduktion typischerweise an einem Rechner durchgeführt. Über die Zeit sind einige Softwarepakete in unterschiedlichen Sprachen (MATLAB, Python, C) entstanden. Eine Übersicht befindet sich unter https://morwiki.mpi-magdeburg.mpg.de/morwiki/index.php/Comparison_of_Software.

Die numerischen Untersuchungen im Rahmen von Vorlesung und Übung werden in MATLAB durchgeführt und orientieren sich an den MATLAB built-in Funktionen der Control System Toolbox, sowie an den Lehrstuhl eigenen `sss` und `sssMOR` Toolboxen, welche auf der Homepage des Lehrstuhls kostenfrei zum Download bereitstehen.

Im Skript werden Sie an einigen Stellen blaue Kästchen finden, in denen Funktionen aufgelistet sind, die im Zusammenhang mit den vorgestellten Inhalten stehen. Diese können verwendet werden, um die theoretische Diskussion aus Vorlesung und Skript mit numerischen Untersuchung in MATLAB anhand praxisnaher *Benchmark* Systemen zu ergänzen. Links zu verschiedenen Ansammlungen an Benchmark Systemen finden Sie unter <http://www.eu-mor.net/support/benchmark-collections/>.

	MATLAB Funktion(en):	<code>ss</code> , <code>dss</code>
	SSS Funktion(en):	<code>sss</code>
	SSSMOR Funktion(en):	<code>ssRed</code>

Kapitel 2

Mathematische Grundlagen

In diesem Kapitel werden einige notwendige mathematische Grundlagen wiederholt oder neu eingeführt.

2.1 Vektorraum

Definition 1. Ein **Vektorraum** \mathcal{V} ist eine Menge von Vektoren, in der die Addition der Vektoren und die Multiplikation eines Vektors mit einem Skalar aus dem Körper \mathbb{K} (für uns gilt: $\mathbb{K} = \mathbb{R}$ oder $\mathbb{K} = \mathbb{C}$) definiert sind und die Ergebnisse dieser Operationen wieder Elemente aus \mathcal{V} sind. Zudem gilt für alle $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{V}$ und $\alpha, \beta \in \mathbb{K}$:

$$\begin{aligned} \text{Kommutativgesetz:} & \quad \mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x} \\ \text{Assoziativgesetz:} & \quad (\mathbf{x} + \mathbf{y}) + \mathbf{z} = \mathbf{x} + (\mathbf{y} + \mathbf{z}) \\ \text{Existenz des Nullelements:} & \quad \exists \mathbf{0} \in \mathcal{V}: \quad \mathbf{x} + \mathbf{0} = \mathbf{x} \\ \text{Existenz des Inversen:} & \quad \exists (-\mathbf{x}) \in \mathcal{V}: \quad \mathbf{x} + (-\mathbf{x}) = \mathbf{0} \\ \text{Distributivgesetz 1:} & \quad (\alpha + \beta)\mathbf{x} = \alpha\mathbf{x} + \beta\mathbf{x} \\ \text{Distributivgesetz 2:} & \quad \alpha(\mathbf{x} + \mathbf{y}) = \alpha\mathbf{x} + \alpha\mathbf{y} \\ & \quad (\alpha\beta)\mathbf{x} = \alpha(\beta\mathbf{x}) \\ \text{Existenz des Einselements:} & \quad \exists 1 \in \mathbb{K}: \quad 1\mathbf{x} = \mathbf{x} \end{aligned}$$

Beispiel: Vektoren in \mathbb{R}^n .

Definition 2. Eine Teilmenge \mathcal{W} eines Vektorraums \mathcal{V} , d. h. $\mathcal{W} \subset \mathcal{V}$, heißt **Untervektorraum** (kurz Unterraum), wenn die Elemente von \mathcal{W} mit den gleichen Regeln des Vektorraums \mathcal{V} (Addition, Skalarmultiplikation) wieder einen Vektorraum bilden, d. h.

$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{W} \text{ und } \alpha \in \mathbb{K}: \quad \mathbf{x} + \mathbf{y} \in \mathcal{W} \text{ und } \alpha\mathbf{x} \in \mathcal{W}. \quad (2.1)$$

Beispiel: Ebene im \mathbb{R}^3 , z. B. die x_1 - x_2 -Ebene:

$$\left\{ \mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x} = \alpha \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \alpha, \beta \in \mathbb{R} \right\} \subset \mathbb{R}^3 \quad (2.2)$$

2.2 Lineare Abhängigkeit und Rang

Definition 3. Die Vektoren $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \in \mathcal{V}$ heißen **linear abhängig**, wenn es Zahlen $c_1, c_2, \dots, c_n \in \mathbb{R}$ gibt, die nicht alle gleich null sind und für die gilt:

$$c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_n \mathbf{v}_n = \mathbf{0}. \quad (2.3)$$

Anderfalls heißen sie **linear unabhängig**.

Lineare Abhängigkeit bedeutet also, dass ein Vektor \mathbf{v}_i mit Hilfe der Anderen durch eine Linearkombination dargestellt werden kann. Die Anzahl an linear unabhängigen Spaltenvektoren einer Matrix entspricht dem Rang.

Definition 4. Der Rang einer Matrix \mathbf{V} ist definiert als die Anzahl an linear unabhängigen Spaltenvektoren von \mathbf{V} .

Daraus folgt, dass eine Matrix $\mathbf{V} \in \mathbb{R}^{m,n}$ genau dann den vollen Rang $\min(m, n)$ besitzt, wenn alle Spaltenvektoren linear unabhängig sind. Man findet in der Literatur zusätzlich die Bezeichnungen "Zeilen-" und "Spaltenrang". Da beide Größen identisch sind, können diese drei Begriffe als Synonyme verwendet werden.

Beispiel: Gegeben sei folgende Matrix:

$$\mathbf{V} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \\ 1 & 4 & 5 \\ 2 & 5 & 7 \\ 3 & 6 & 9 \end{bmatrix} \quad (2.4)$$

Die Spalten von \mathbf{V} sind linear abhängig, da \mathbf{v}_3 die Summe der beiden anderen Spalten ist:

$$1\mathbf{v}_1 + 1\mathbf{v}_2 - 1\mathbf{v}_3 = \mathbf{0} \quad \Rightarrow \quad \mathbf{v}_3 = \mathbf{v}_1 + \mathbf{v}_2. \quad (2.5)$$

Hingegen sind bspw. \mathbf{v}_1 und \mathbf{v}_2 linear unabhängig, daher gilt $\text{Rang}(\mathbf{V}) = 2$.

Beispiel: Sei $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \in \mathbb{R}^n$ und $\mathbf{V} := [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n]$, dann gilt:

$$\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \text{ linear unabhängig} \Leftrightarrow \text{Rang}(\mathbf{V}) = n \Leftrightarrow \det(\mathbf{V}) \neq 0. \quad (2.6)$$

Hinweis: Während die Äquivalenz zwischen linearer Unabhängigkeit und Spaltenrang für beliebige Matrizen gilt ist die Bedingung mit der Determinante nur für quadratische Matrizen sinnvoll.

 MATLAB Funktion(en): rank, det

2.3 Basis

Definition 5. Eine Basis \mathcal{B} eines Vektorraums \mathcal{V} ist eine Teilmenge $\mathcal{B} \subset \mathcal{V}$ mit den beiden Eigenschaften:

1. Jedes Element von \mathcal{V} lässt sich als Linearkombination von Vektoren aus \mathcal{B} darstellen.
2. Die Basisvektoren sind linear unabhängig.

Beispiel: Die Spalten aller 3×3 Matrizen mit Rang 3 bilden Basen für den \mathbb{R}^3 .

Definition 6. Die Dimension eines Vektorraums \mathcal{V} ist definiert als die Anzahl der Basisvektoren in \mathcal{B} und wird mit $\dim(\mathcal{V})$ dargestellt.

Definition 7. Eine Basis $\mathcal{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_n\}$ heißt orthogonal, wenn alle Vektoren paarweise orthogonal zueinander sind, d. h.

$$\mathbf{b}_i^T \mathbf{b}_j = 0, \quad \text{für } i \neq j. \quad (2.7)$$

Sind die Basisvektoren zusätzlich normiert, d. h. $|\mathbf{b}_i| = 1$, so spricht man von einer orthonormalen Basis.

Beispiel: Kanonische Basis für \mathbb{R}^3 (orthonormale Basis):

$$\mathbf{b}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{b}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{b}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.8)$$

Matrizen, deren Spaltenvektoren eine orthonormale Basis bilden sind von besonderer Bedeutung und werden im folgenden eingeführt.

Definition 8. Eine quadratische reelle Matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ heißt orthogonal, wenn ihre Spaltenvektoren zueinander orthonormal sind (also nicht nur orthogonal). In diesem Fall ist ihre Transponierte gleich der Inversen:

$$\mathbf{A}^T \mathbf{A} = \mathbf{A} \mathbf{A}^T = \mathbf{I}_n \quad \Leftrightarrow \quad \mathbf{A}^T = \mathbf{A}^{-1} \quad (2.9)$$

Die Entsprechung bei komplexen Matrizen $\mathbf{A} \in \mathbb{C}^{n \times n}$ heißt unitäre Matrix.

Laut Definition muss eine orthogonale Matrix quadratisch sein. Bei der Modellordnungsreduktion spielen allerdings rechteckige Matrizen $\mathbf{V} \in \mathbb{R}^{n \times q}$ mit $n > q$ eine große Rolle, deren Spaltenvektoren eine orthonormale Basis für einen q -dimensionalen Unterraum des \mathbb{R}^n bilden. Da es für solche Matrizen in der mathematischen Literatur keine eindeutige Bezeichnung gibt, werden wir uns im folgenden einen sog. „abuse of notation“¹ erlauben und gelegentlich auch rechteckige Matrizen $\mathbf{V} \in \mathbb{R}^{n \times q}$ mit dem Attribut *orthogonal* versehen, wenn diese die Eigenschaft erfüllen:

$$\mathbf{V}^T \mathbf{V} = \mathbf{I}_q. \quad (2.10)$$

2.4 Bild-und Nullraum

Definition 9. Der *Bildraum* (engl.: *range, image*) einer Matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ ist die Menge aller Vektoren $\mathbf{y} \in \mathbb{R}^m$, welche sich als Produkt $\mathbf{y} = \mathbf{A}\mathbf{x}$ für beliebige $\mathbf{x} \in \mathbb{R}^n$ darstellen lassen.

Das Produkt $\mathbf{A}\mathbf{x}$ lässt sich darstellen als gewichtete Summe der Spalten \mathbf{A}_i in der Form

$$\mathbf{y} = \mathbf{A}\mathbf{x} = \mathbf{A}_1 x_1 + \mathbf{A}_2 x_2 + \cdots + \mathbf{A}_n x_n,$$

wodurch man erkennt, dass der Bildraum der Matrix genau der Vektorraum ist, welcher von dessen Spalten aufgespannt wird.

Definition 10. Der *Nullraum* oder *Kern* (engl.: *nullspace, kernel*) einer Matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ ist die Menge aller Vektoren $\mathbf{x} \in \mathbb{R}^n$, für die gilt: $\mathbf{A}\mathbf{x} = \mathbf{0}$.

Beispiel:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 \\ 3 & 3 & 6 \\ 0 & 2 & 2 \\ 1 & 1 & 2 \end{bmatrix} \quad (2.11)$$

Diese Matrix hat $\text{Rang}(\mathbf{A}) = 2$, d. h. der *Bildraum* hat Dimension 2:

$$\text{Bild}(\mathbf{A}) = \text{span} \left(\left(\left(\begin{bmatrix} 1 \\ 3 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 3 \\ 2 \\ 1 \end{bmatrix} \right) \right) \right) \quad (2.12)$$

und der *Nullraum* hat Dimension 1:

$$\text{Kern}(\mathbf{A}) = \left\{ \mathbf{x} \in \mathbb{R}^3 : \mathbf{x} = \alpha \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}, \alpha \in \mathbb{R} \right\}. \quad (2.13)$$

¹“In mathematics, abuse of notation occurs when an author uses a mathematical notation in a way that is not formally correct but that seems likely to simplify the exposition or suggest the correct intuition (while being unlikely to introduce errors or cause confusion)”. (Quelle: Wikipedia)

Eine Basis für den Bild-/Nullraum kann z. B. durch eine Singulärwertzerlegung berechnet werden.

 MATLAB Funktion(en): `null`

2.5 Singulärwertzerlegung







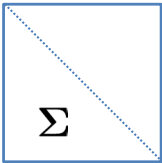





Die *Singulärwertzerlegung* (engl.: Singular Value Decomposition (SVD)) ist verwandt mit der Eigenwertzerlegung (engl.: Eigenvalue Decomposition (EVD)), aber auch für rechteckige Matrizen $\mathbf{A} \in \mathbb{R}^{m \times n}$ definiert:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \quad (2.14)$$

wobei die Matrizen $\mathbf{U} \in \mathbb{R}^{m \times m}$ und $\mathbf{V} \in \mathbb{R}^{n \times n}$ orthogonal sind ($\mathbf{U}^T\mathbf{U} = \mathbf{I}_m$ und $\mathbf{V}^T\mathbf{V} = \mathbf{I}_n$). Die Matrix $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$ hat nur Einträge auf der Diagonalen und ist abhängig von den Dimensionen von \mathbf{A} :

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_{\min(m,n)} \end{bmatrix}. \quad (2.15)$$

Die Einträge $\sigma_1 \geq \sigma_2 \dots \geq \sigma_{\min(m,n)}$ sind nichtnegativ (d. h. positiv oder Null), der Größe nach geordnet und heißen Singulärwerte der Matrix \mathbf{A} . Die folgende Abbildung verdeutlicht die Zerlegung für verschiedene Dimensionen:

	\mathbf{A} [$m \times n$]	=	\mathbf{U} [$m \times m$]	$\mathbf{\Sigma}$ [$m \times n$]	\mathbf{V}^T [$n \times n$]
$m > n :$		=			
$m = n :$		=			
$m < n :$		=			



MATLAB Funktion(en): `svd`, `svds`

2.5.1 Eigenschaften der SVD

Sei $\text{Rang}(A) = p \leq \min(m, n)$, und schreibe $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_m]$, $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_n]$, dann gilt:

1. $\sigma_i = 0$ für $i = p + 1, \dots, \min(m, n)$.
2. Die Singulärwerte σ_i sind die Wurzeln der ersten $\min(m, n)$ Eigenwerte von $\mathbf{A}\mathbf{A}^T$ bzw. $\mathbf{A}^T\mathbf{A}$:

$$\sigma_i = \sqrt{\lambda_i(\mathbf{A}^T\mathbf{A})} = \sqrt{\lambda_i(\mathbf{A}\mathbf{A}^T)}. \quad (2.16)$$

Deshalb sind alle Singulärwerte reell und nichtnegativ: $\sigma_i \geq 0$.

3. Die Eigenvektoren von $\mathbf{A}\mathbf{A}^T$ (bzw. $\mathbf{A}^T\mathbf{A}$) sind die Singulärvektoren \mathbf{u}_i (bzw. \mathbf{v}_i).
4. Die Vektoren $\mathbf{v}_{p+1}, \dots, \mathbf{v}_n$ bilden eine Basis für den Nullraum von \mathbf{A} .
5. Die Vektoren $\mathbf{u}_1, \dots, \mathbf{u}_p$ bilden eine Basis für den Bildraum von \mathbf{A} .
6. Aus der Definition der SVD folgt, dass eine Matrix als dyadische Summe dargestellt werden kann:

$$\mathbf{A} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \dots + \sigma_p \mathbf{u}_p \mathbf{v}_p^T. \quad (2.17)$$

Spezialfall: Für positiv (semi-)definite Matrizen (symmetrische Matrizen mit positiven (bzw. nicht negativen) Eigenwerten) gilt: SVD = EVD

$$\mathbf{A} = \mathbf{A}^T \succeq 0 \quad \Rightarrow \quad \mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T,$$

da die Eigenvektoren eine orthogonale Basis bilden und alle Eigenwerte nicht-negativ sind.

2.5.2 Matrix-Approximation

Die SVD kann u. a. dazu verwendet werden Matrizen zu approximieren. Dabei ist eine Matrix \mathbf{X} vom Rang $k \leq p$ gesucht, so dass folgende Differenz minimiert wird:

$$\min_{\text{Rang}(\mathbf{X})=k} \|\mathbf{A} - \mathbf{X}\|_2 \quad (2.18)$$

Die 2-Norm einer Matrix ist folgendermaßen als induzierte Norm definiert:

$$\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2 = \sigma_{\max}(\mathbf{A}) = \sqrt{\lambda_{\max}(\mathbf{A}^T\mathbf{A})} \quad (2.19)$$

Man kann zeigen, dass die Lösung des Minimierungsproblems genau die Summe der ersten k dyadischen Produkte der SVD ist,

$$\mathbf{X} = \sum_{i=1}^k \sigma_i \mathbf{u}_i \mathbf{v}_i^T = \mathbf{U} \widehat{\Sigma} \mathbf{V}^T, \quad (2.20)$$

wobei $\widehat{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0)$. Außerdem gilt für das Minimum:

$$\min_{\text{Rang}(X)=k} \|\mathbf{A} - \mathbf{X}\|_2 = \sigma_{k+1}(\mathbf{A}). \quad (2.21)$$

2.6 Normen

Für gewisse Analysezwecke ist es hilfreich, die „Intensität“ eines Zeitsignals oder eines dynamischen Systems in Form einer Norm zu beurteilen. D. h. einem Signal im Zeitbereich oder einer Funktion im Frequenzbereich wird durch die Norm ein skalarer nichtnegativer Wert zugeordnet. Eine weit verbreitete Beurteilung von Zeitsignalen sind die \mathcal{L}_p -Normen, wobei p in der Regel für 1, 2 oder ∞ steht. Entsprechendes gilt für die \mathcal{H}_p -Normen zur Beurteilung dynamischer Systeme. Wir werden folgende Normen verwenden:

Definition 11. Die \mathcal{L}_2 -Norm für Zeitsignale ist definiert als:

$$\|\mathbf{x}(t)\|_{\mathcal{L}_2} = \sqrt{\int_0^\infty |\mathbf{x}(t)|^2 dt}. \quad (2.22)$$

Definition 12. Die \mathcal{H}_2 -Norm für (SISO) Systeme ist definiert als:

$$\|G(s)\|_{\mathcal{H}_2} = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} |G(j\omega)|^2 d\omega} = \sqrt{\int_0^\infty |g(t)|^2 dt}. \quad (2.23)$$


Definition 13. Die \mathcal{H}_∞ -Norm für (SISO) Systeme ist definiert als:


$$\|G(s)\|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathbb{R}} |G(j\omega)|. \quad (2.24)$$

Für SISO-System ist die \mathcal{H}_2 -Norm somit gleichbedeutend mit der \mathcal{L}_2 -Norm der Impulsantwort $g(t)$, während die \mathcal{H}_∞ -Norm den höchsten Wert im Amplitudengang des Systems beschreibt. Dies ist gleichbedeutend mit der maximal möglichen stationären Verstärkung des Systems bei harmonischer Anregung.

Für die Bestimmung der Approximationsgüte bei der Reduktion von LZI-Modellen wird häufig der Approximationsfehler im Frequenzbereich $G(s) - G_r(s)$ anhand der \mathcal{H}_2 - oder \mathcal{H}_∞ -Normen gemessen. $G_r(s)$ stellt dabei die Übertragungsfunktion des reduzierten Modells dar. Es lässt sich zeigen, dass die Fehlermaße im Frequenzbereich auch eine obere Schranke für den Approximationsfehler im Zeitbereich darstellen, gemäß der Beziehungen

$$\begin{aligned} \|y(t) - y_r(t)\|_{\mathcal{L}_\infty} &\leq \|G(s) - G_r(s)\|_{\mathcal{H}_2} \|u(t)\|_{\mathcal{L}_2}, \\ \|y(t) - y_r(t)\|_{\mathcal{L}_2} &\leq \|G(s) - G_r(s)\|_{\mathcal{H}_\infty} \|u(t)\|_{\mathcal{L}_2}. \end{aligned} \quad (2.25)$$

 MATLAB Funktion(en): `norm(sys)`, `norm(sys, inf)`

 SSS Funktion(en): `norm(sys)`, `norm(sys, inf)`

Kapitel 3

Projektive Modellreduktion

Die Methoden der MOR, die wir hier betrachten wollen, basieren auf Projektionen. Ein anschauliches Beispiel einer Projektion ist der Schattenwurf: Der Schatten ist ein zweidimensionales Abbild eines dreidimensionalen Gegenstandes. Dieses Kapitel führt Projektionen allgemein ein, unabhängig von der gewählten Methode zur MOR.

3.1 Der Projektor

Mathematisch gesehen ist eine Projektion eine lineare Abbildung eines Vektorraums \mathcal{V} in sich selbst, und wird repräsentiert durch Multiplikation mit dem Projektor $\mathbf{P} \in \mathbb{R}^{n \times n}$: $\mathbf{x}_{\text{projiziert}} = \mathbf{P}\mathbf{x}$. Das Bild einer Projektion ist entweder ein Unterraum von \mathcal{V} oder \mathcal{V} selbst.

Definition 14. *Ein Projektor ist definiert durch:*

$$\mathbf{P} = \mathbf{P}^2. \quad (3.1)$$

Diese einfache Definition ist tatsächlich ausreichend und man bezeichnet solche Matrizen auch als *idempotent*. Eine anschauliche Erklärung ist, dass eine wiederholte Projektion nichts mehr ändert, d.h. wenn man einen projizierten Vektor $\mathbf{x}_{\text{projiziert}} = \mathbf{P}\mathbf{x}$ als neuen Ausgangsvektor nimmt und diesen erneut projiziert, muss sich der gleiche Vektor ergeben: $\mathbf{P}\mathbf{x}_{\text{projiziert}} = \mathbf{x}_{\text{projiziert}}$.

Satz 1. *Die Eigenwerte eines Projektors \mathbf{P} sind entweder 0 oder 1.*

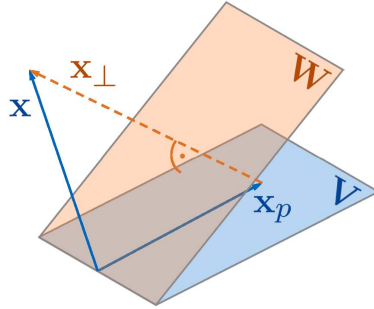
Beweis. Sei $\mathbf{P} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$ die Eigenzerlegung des Projektors. Aus der Definition folgt

$$\begin{aligned} \mathbf{P}^2 &= \mathbf{V}\mathbf{D}\mathbf{V}^{-1}\mathbf{V}\mathbf{D}\mathbf{V}^{-1} = \mathbf{V}\mathbf{D}^2\mathbf{V}^{-1} \stackrel{\mathbf{P}=\mathbf{P}^2}{\implies} \mathbf{D} = \mathbf{D}^2 \quad (\text{Diagonalmatrix}) \\ &\implies \lambda_i(\mathbf{P}) = \lambda_i(\mathbf{D}) \in \{0, 1\}. \quad \square \end{aligned}$$

Bemerkung: Umgekehrt gilt der Satz i.A. nicht, wie die Beispielmatrix $\begin{bmatrix} 1 & 3 \\ 0 & 1 \end{bmatrix}$ zeigt.

3.2 Erzeugung von Projektoren

Wie können gezielt Projektoren \mathbf{P} erzeugt werden, um eine gewünschte Projektion zu erreichen? Das soll beispielhaft an einer Projektion im \mathbb{R}^3 hergeleitet werden:



Der Vektor $\mathbf{x} \in \mathbb{R}^3$ soll auf die Ebene projiziert werden, welche durch die Spalten der Matrix $\mathbf{V} \in \mathbb{R}^{3 \times 2}$ aufgespannt wird. Die Projektion soll entlang der Richtung \mathbf{x}_\perp erfolgen. Hierzu wird die Matrix $\mathbf{W} \in \mathbb{R}^{3 \times 2}$ definiert, deren Spalten orthogonal zu \mathbf{x}_\perp sind. Damit der projizierte Vektor \mathbf{x}_p in \mathbf{V} liegt, muss sich \mathbf{x}_p als eine Linearkombination der Spalten von \mathbf{V} darstellen lassen, also:

$$\mathbf{x}_p = \mathbf{V}\mathbf{r}, \quad (3.2)$$

wobei $\mathbf{r} \in \mathbb{R}^2$ noch unbekannt ist. Wir wissen außerdem, dass die Vektorkette geschlossen sein muss:

$$\mathbf{x}_\perp = \mathbf{x} - \mathbf{x}_p. \quad (3.3)$$

Die Spalten von \mathbf{W} sind orthogonal zu \mathbf{x}_\perp , also gilt:

$$\mathbf{W}^T \mathbf{x}_\perp \stackrel{!}{=} \mathbf{0}. \quad (3.4)$$

Einsetzen der Gleichungen (3.2) und (3.3) in (3.4) liefert:

$$\mathbf{W}^T \mathbf{x}_\perp = \mathbf{W}^T \mathbf{x} - \mathbf{W}^T \mathbf{x}_p = \mathbf{W}^T \mathbf{x} - \mathbf{W}^T \mathbf{V}\mathbf{r} = \mathbf{0} \quad (3.5)$$

$$\Leftrightarrow \mathbf{W}^T \mathbf{V}\mathbf{r} = \mathbf{W}^T \mathbf{x} \quad (3.6)$$

$$\Leftrightarrow \mathbf{r} = (\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T \mathbf{x} \quad (3.7)$$

Mit Gleichung (3.2) kann damit der Projektor angegeben werden:

$$\mathbf{x}_p = \mathbf{V}\mathbf{r} = \underbrace{\mathbf{V} (\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T}_{\mathbf{P}} \mathbf{x} \quad (3.8)$$

Diese Herleitung gilt auch für beliebige höhere Dimensionen, weshalb beliebige Projektionsmatrizen $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n \times q}$ zu einem Projektor $\mathbf{P} \in \mathbb{R}^{n \times n}$ führen:

$$\mathbf{P} = \mathbf{V} (\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T. \quad (3.9)$$

$$\square = \square (\square)^{-1} \square$$

Dabei möchten wir ohne Einschränkung der Allgemeinheit annehmen, dass Projektionsmatrizen \mathbf{V} und \mathbf{W} den vollen Spaltenrang besitzen, d.h. Basismatrizen zu den jeweiligen Unterräumen $\text{Bild}(\mathbf{V})$ und $\text{Bild}(\mathbf{W})$ sind. In ähnlicher Weise ist die Annahme der Invertierbarkeit von $\mathbf{W}^T\mathbf{V}$, welche in Gleichung (3.9) erforderlich ist, sinnvoll: Ist die Matrix $\mathbf{W}^T\mathbf{V}$ singulär, so ist die damit verbundene Projektion nicht sinnvoll, denn Projektionsrichtung und Projektionsebene verlaufen teilweise parallel zueinander, wie folgender Satz zeigt.

Satz 2. Die Beziehung $\det(\mathbf{W}^T\mathbf{V}) = 0$ gilt genau dann wenn ein Vektor $\tilde{\mathbf{v}} \in \text{Bild}(\mathbf{V})$ existiert, sodass $\mathbf{W}^T\tilde{\mathbf{v}} = 0$ gilt.

Beweis.

$$\begin{aligned} \det(\mathbf{W}^T\mathbf{V}) = 0 &\iff \exists \mathbf{x} \in \mathbb{R}^q, \mathbf{x} \neq 0 \mid (\mathbf{W}^T\mathbf{V})\mathbf{x} = 0 \\ &\iff \exists \mathbf{x} \in \mathbb{R}^q, \mathbf{x} \neq 0 \mid \mathbf{W}^T \begin{bmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_q \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_q \end{bmatrix} \\ &\iff \exists \tilde{\mathbf{v}} \in \text{Bild}(\mathbf{V}), \tilde{\mathbf{v}} \neq 0 \mid \mathbf{W}^T\tilde{\mathbf{v}} = 0 \end{aligned}$$

□

Satz 3. $\mathbf{P} = \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T$ ist ein Projektor.

Beweis. Der Beweis erfolgt durch prüfen der Definition:

$$\mathbf{P}^2 = \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1} \underbrace{\mathbf{W}^T\mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}}_{=\mathbf{I}} \mathbf{W}^T = \mathbf{P}. \quad \square$$

Für unsere Zwecke ist es sinnvoll Projektoren in der Formulierung (3.9) zu betrachten, da $\text{Bild}(\mathbf{V})$ der Unterraum ist auf den projiziert wird, und durch $\text{Bild}(\mathbf{W})$ die Richtung der Projektion festgelegt wird. Hierzu benötigen wir folgende Definition.

Definition 15. Sei \mathcal{W} Unterraum von \mathbb{R}^n . Dann ist das orthogonale Komplement \mathcal{W}^\perp von \mathcal{W} definiert als:

$$\mathcal{W}^\perp := \{ \mathbf{x} \in \mathbb{R}^n : \mathbf{x}^T\mathbf{y} = 0 \text{ für alle } \mathbf{y} \in \mathcal{W} \}$$

Die Richtungen einer Projektion sind das orthogonale Komplement von $\text{Bild}(\mathbf{W})$, also $[\text{Bild}(\mathbf{W})]^\perp$, denn beim Produkt $\mathbf{P}\mathbf{x}$ wird rechts $\mathbf{W}^T\mathbf{x}$ gebildet. Durch das Produkt $\mathbf{W}^T\mathbf{x}$ werden alle Richtungen die orthogonal zu $\text{Bild}(\mathbf{W})$ sind bei der Projektion „gelöscht“. In unserem Beispiel oben ist das genau \mathbf{x}_\perp , denn $\mathbf{W}^T\mathbf{x}_\perp = \mathbf{0}$. Aus ähnlichen Überlegungen und aus Gleichung (3.2) folgt außerdem:

$$\text{Bild}(\mathbf{P}) = \text{Bild}(\mathbf{V}), \quad \text{Kern}(\mathbf{P}) = [\text{Bild}(\mathbf{W})]^\perp. \quad (3.10)$$

Definition 16. Eine Projektion heißt orthogonal, wenn zusätzlich zu $\mathbf{P} = \mathbf{P}^2$ noch gilt: $\mathbf{P} = \mathbf{P}^T$. Andernfalls spricht man von einer schiefen Projektion.

Beweis. Die Eigenvektoren einer symmetrischen Matrix $\mathbf{P} = \mathbf{P}^T$ bilden eine orthogonale Basis. Daraus folgt, dass $\text{Bild}(\mathbf{P}) \perp \text{Kern}(\mathbf{P})$. Da aber $\text{Kern}(\mathbf{P})$ die Richtung der Projektion definiert, wird somit orthogonal auf $\text{Bild}(\mathbf{P})$ projiziert. \square

Für einen Projektor (3.9) führt die Wahl $\mathbf{W} := \mathbf{V}$ zu einer orthogonalen Projektion: $\mathbf{P} = \mathbf{V}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T = \mathbf{P}^T$. Das lässt sich auch mit dem Bild am Anfang des Kapitels verifizieren. Das bisher gesagte zu Projektionen wird in folgender Tabelle zusammen gefasst:

	Allgemeine Projektion	Erzeugte Projektion	Anschauliches Beispiel
Projektion	Abbildung mit: \mathbf{P}	Abbildung mit: $\mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T$	Schattenwurf
Projektion auf	$\text{Bild}(\mathbf{P})$	$\text{Bild}(\mathbf{V})$	Boden
Richtung der Projektion	$\text{Kern}(\mathbf{P})$	orthogonal zu \mathbf{W} bzw. $[\text{Bild}(\mathbf{W})]^\perp$	Sonnenachse
Orthogonale Projektion	$\mathbf{P} = \mathbf{P}^T$	$\mathbf{V}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T$	Sonne im Zenit
Schiefe Projektion	$\mathbf{P} \neq \mathbf{P}^T$	$\mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T$, mit: $\text{Bild}(\mathbf{V}) \neq \text{Bild}(\mathbf{W})$	z. B. Abendsonne

Wie in der Übung 2.2 gezeigt wird sind Projektoren *unabhängig* von der Wahl der Basismatrizen \mathbf{V}, \mathbf{W} für die Unterräume \mathcal{V}, \mathcal{W} . Dadurch gilt für beliebige reguläre Matrizen $\mathbf{T}_v, \mathbf{T}_w \in \mathbb{R}^{q \times q}$ mit $\tilde{\mathbf{V}} = \mathbf{V}\mathbf{T}_v$ und $\tilde{\mathbf{W}} = \mathbf{W}\mathbf{T}_w$

$$\mathbf{P} = \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T = \tilde{\mathbf{V}}\left(\tilde{\mathbf{W}}^T\tilde{\mathbf{V}}\right)^{-1}\tilde{\mathbf{W}}^T. \quad (3.11)$$

Daraus folgt, dass bei einer orthogonalen Projektion ohne Einschränkung der Allgemeinheit $\mathbf{W} = \mathbf{V}$ angenommen werden darf, denn die Wahl jeder anderen Basis zum selben Projektor führt.

Es sind noch zwei wichtige Sonderfälle zu erwähnen:

1. Sind die zwei Projektionsmatrizen biorthogonal, d. h. $\mathbf{W}^T\mathbf{V} = \mathbf{I}_q$, folgt der (i. Allg. schiefe) Projektor:

$$\mathbf{P} = \mathbf{V}\mathbf{W}^T. \quad (3.12)$$

Achtung: „Orthogonale Projektion“ und „biorthogonale Basen“ bezeichnen zwei unterschiedliche Dinge!

2. Ist eine „Abschneide-Operation“ gewünscht (d. h. beibehalten nur der ersten q Koordinatenrichtungen eines Vektors) werden die Projektionsmatrizen

$$\mathbf{V} = \mathbf{W} = \begin{bmatrix} \mathbf{I}_q \\ \mathbf{0} \end{bmatrix} \text{ verwendet.} \Rightarrow \mathbf{P} = \begin{bmatrix} \mathbf{I}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

3.3 Projektive MOR

Bei der Modellreduktion soll der Zustandsvektor $\mathbf{x}(t) \in \mathbb{R}^n$ durch einen Vektor $\mathbf{x}_r(t) \in \mathbb{R}^q$ mit niedrigerer Dimension $q \ll n$ approximiert werden. Man wählt hierzu eine Projektionsmatrix $\mathbf{V} \in \mathbb{R}^{n \times q}$, die einen q -dimensionalen Unterraum aufspannt und deren Spalten die Basisvektoren für den reduzierten Zustand $\mathbf{x}_r(t)$ darstellen. Beinhaltet der Unterraum die Hauptdynamik des originalen Systems, ist die folgende Approximation zulässig:

$$\mathbf{x}(t) \approx \mathbf{V}\mathbf{x}_r(t). \quad (3.13)$$

Für ein System in minimaler Realisierung kann diese Gleichung nie exakt erfüllt sein, es liegt ein Fehler $\mathbf{e}(t)$ vor:

$$\mathbf{x}(t) = \mathbf{V}\mathbf{x}_r(t) + \mathbf{e}(t). \quad (3.14)$$

Setzt man dies in die Zustandsgleichung des Originalsystems ein, ergibt sich

$$\mathbf{V}\dot{\mathbf{x}}_r(t) = \mathbf{A}\mathbf{V}\mathbf{x}_r(t) + \mathbf{b}u(t) + \boldsymbol{\epsilon}(t), \quad (3.15)$$

wobei das Residuum $\boldsymbol{\epsilon}(t) = \mathbf{A}\mathbf{e}(t) - \dot{\mathbf{e}}(t)$ die Fehlerterme durch die Approximation zusammenfasst.

Die Zustandsdifferentialgleichung ist überbestimmt: q Variablen in \mathbf{x}_r und n Gleichungen. Wie aus der linearen Algebra bekannt besitzt ein überbestimmtes Gleichungssystem der allgemeinen Form $\mathbf{M}\boldsymbol{\xi} = \boldsymbol{\gamma}$ genau dann eine eindeutige Lösung, wenn die rechte Seite $\boldsymbol{\gamma}$ im Vektorraum $\text{Bild}(\mathbf{M})$ ist, d.h. wenn $\boldsymbol{\gamma}$ sich vollständig aus einer Linearkombination der Spalten von \mathbf{M} darstellen lässt. Für Gleichung (3.15) ist diese Bedingung i.A. nicht erfüllt. Um dennoch eine eindeutige Lösung für $\mathbf{x}_r(t)$ zu erhalten, wird diese Bedingung *erzwingen* durch Projektion der Zustandsdifferentialgleichung auf den q -dimensionalen Unterraum $\text{Bild}(\mathbf{V})$ ¹:

$$\mathbf{P}\mathbf{V}\dot{\mathbf{x}}_r(t) = \mathbf{P}\mathbf{A}\mathbf{V}\mathbf{x}_r(t) + \mathbf{P}\mathbf{b}u(t) + \mathbf{P}\boldsymbol{\epsilon}(t), \quad (3.16)$$

$$\begin{aligned} \Rightarrow \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T\mathbf{V}\dot{\mathbf{x}}_r(t) &= \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T\mathbf{A}\mathbf{V}\mathbf{x}_r(t) + \\ &+ \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T\mathbf{b}u(t) + \\ &+ \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T\boldsymbol{\epsilon}(t) \end{aligned} \quad (3.17)$$

¹Man beachte an dieser Stelle die Ähnlichkeit zum „Least-Squares“ Ansatz zum Lösen überbestimmter Gleichungssysteme. Da es i.A. keine eindeutige Lösung gibt, wird diejenige Lösung berechnet, welche die kleinste Euklidische Norm des Fehlers verursacht. Dies wird durch eine orthogonale Projektion der Gleichung auf $\text{Bild}(\mathbf{V})$ erzielt.

Das projizierte Gleichungssystem (3.17) kann somit eindeutig in reduzierten Koordinaten gelöst werden für jedes beliebige Residuum $\epsilon(t)$, welcher jedoch im Allgemeinen nicht bekannt ist. Um das Problem zu beheben wird die Lösung der projizierten Differentialgleichung so gewählt, dass das resultierende $\epsilon(t)$ bei der Projektion verschwindet: $\mathbf{V} (\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T \epsilon(t) = \mathbf{0}$. Eine Interpretation ist, dass das Residuum $\epsilon(t)$ dadurch im Kern(\mathbf{P}) liegt, d. h. $\mathbf{W}^T \epsilon(t) = \mathbf{0}$, oder $\epsilon(t) \perp \mathbf{W}$.

Definition 17. Die Bedingung $\epsilon(t) \perp \mathbf{W}$ wird als Petrov-Galerkin-Bedingung bezeichnet. Liegt eine orthogonale Projektion vor – d. h. $\mathbf{W} = \mathbf{V}$ und damit $\epsilon(t) \perp \mathbf{V}$ – nennt man dies Galerkin-Bedingung.

Bei jedem Summanden in (3.17) steht links die Matrix \mathbf{V} und rechts davon ein q -dimensionaler Vektor. Da angenommen wird, dass die Spalten von \mathbf{V} linear unabhängig sind, ist die Gleichung (3.17) genau dann erfüllt, wenn sie von den q -dimensionalen Vektoren rechts von \mathbf{V} erfüllt wird; das bedeutet: man kann in diesem Fall die Matrix \mathbf{V} einfach weglassen. Übrig bleibt das reduzierte Modell, d. h. q Gleichungen mit q Unbekannten, die für $\mathbf{x}_r(t)$ eindeutig gelöst werden können:

$$\begin{aligned} \dot{\mathbf{x}}_r(t) &= \overbrace{(\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T \mathbf{A} \mathbf{V}}^{\mathbf{A}_r} \mathbf{x}_r(t) + \overbrace{(\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T \mathbf{b}}^{\mathbf{b}_r} u(t), \\ y_r &= \underbrace{\mathbf{c} \mathbf{V}}_{\mathbf{c}_r} \mathbf{x}_r(t). \end{aligned} \quad (3.18)$$

Sind die Basen \mathbf{V} und \mathbf{W} biorthogonal, also $\mathbf{W}^T \mathbf{V} = \mathbf{I}_q$, lautet das reduzierte System:


$$\begin{aligned} \dot{\mathbf{x}}_r(t) &= \mathbf{W}^T \mathbf{A} \mathbf{V} \mathbf{x}_r(t) + \mathbf{W}^T \mathbf{b} u(t), \\ y_r &= \mathbf{c} \mathbf{V} \mathbf{x}_r(t). \end{aligned} \quad (3.19)$$

Hinweis: Die Petrov-Galerkin Bedingung wird benötigt, um die (unbekannten) Fehlerterme, die zwangsläufig durch die Approximation des Zustandsvektors entstehen, aus den Gleichungen zu entfernen. Das resultierende reduzierte Modell würde nämlich genau die Form (3.18) annehmen, wenn man in der Herleitung nur den Ansatz $\mathbf{x}(t) \approx \mathbf{V} \mathbf{x}_r(t)$ einsetzen würde, also ohne Berücksichtigung der Fehlerterme. Allerdings spielt der Fehler eine wichtige Rolle bei der Interpretation von (3.18): Durch Lösen der projizierten Differentialgleichung in reduzierten Koordinaten ergibt sich ein Fehler in der Zustandsdifferentialgleichung (das Residuum $\epsilon(t)$), welcher durch die Projektion verschwindet. Durch Lösung des reduzierten Modells (3.18) wird also die Petrov-Galerkin Bedingung immer erfüllt!

Beweis:

$$\begin{aligned} \dot{\mathbf{x}}(t) - \mathbf{A} \mathbf{x}(t) - \mathbf{b} u(t) &= 0, \\ \Rightarrow \mathbf{P} (\dot{\mathbf{x}}(t) - \mathbf{A} \mathbf{x}(t) - \mathbf{b} u(t)) &= 0, \\ \Rightarrow \mathbf{P} (\mathbf{V} \dot{\mathbf{x}}_r(t) - \mathbf{A} \mathbf{V} \mathbf{x}_r(t) - \mathbf{b} u(t)) - \mathbf{P} \epsilon(t) &= 0, \\ \Rightarrow \mathbf{P} \epsilon(t) &= 0, \end{aligned} \quad (3.20)$$

wobei der letzte Schritt aus der Lösung von (3.18) folgt.

 sssMOR Funktion(en): projectiveMOR

3.4 Zusammenfassung

In diesem Kapitel haben wir das prinzipielle Vorgehen bei der Modellreduktion von linearen Systemen kennengelernt, bei dem das reduzierte Modell durch Projektion aus dem originalen System hervorgeht. Ziel einer Methode zur Modellreduktion ist somit die Bestimmung von zweckmäßigen Basen \mathbf{V} und \mathbf{W} . Im Folgenden werden entsprechend gängige Verfahren zur Berechnung geeigneter Projektionsmatrizen vorgestellt.

Kapitel 4

Modale Reduktion

Engl.: Modal Reduction oder Modal Truncation

Einer der ersten Ansätze zur Ordnungsreduktion eines LZI Systems in Zustandsraumdarstellung war die *modale Reduktion*, die ihren Ursprung in den 1960ern Jahren hat. Die Grundidee basiert auf der Betrachtung des Übertragungsverhaltens nach Zustandstransformation auf *modale Koordinaten*. Damit lässt sich das System durch n *entkoppelte Pfade* darstellen, die jeweils zu einem Eigenwert und einer modalen Koordinate gehören. Dabei ist es naheliegend, Pfade (entsprechend Modalkoordinaten und Eigenwerte) zu vernachlässigen, die keine signifikante Auswirkung auf das Übertragungsverhalten aufweisen.

In Folgenden wird das Verfahren nach Litz vorgestellt, so wie es im Buch (Föllinger, 2013, Kap. 8.6) zu finden ist. Dieses setzt voraus, dass alle Eigenwerte von \mathbf{A} einfach sind.

4.1 Modaltransformation

Durch Eigenzerlegung der Systemmatrix $\mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \mathbf{\Lambda}$ und Basiswechsel $\mathbf{x} = \mathbf{T}\mathbf{z}$ (*Modaltransformation*) ergibt sich die Zustandsraumdarstellung in Modalkoordinaten

$$\begin{aligned}\dot{\mathbf{z}}(t) &= \mathbf{\Lambda}\mathbf{z}(t) + \hat{\mathbf{b}}u(t), \\ y(t) &= \hat{\mathbf{c}}\mathbf{z}(t),\end{aligned}\tag{4.1}$$

mit $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_n)$, $\hat{\mathbf{b}} = \mathbf{T}^{-1}\mathbf{b}$, $\hat{\mathbf{c}} = \mathbf{c}\mathbf{T}$. Dabei ist \mathbf{T} die Eigenvektormatrix von \mathbf{A} , sodass für jede Spalte \mathbf{t}_k gilt: $\mathbf{A}\mathbf{t}_k = \lambda_k\mathbf{t}_k$. Die obere Gleichung in (4.1) stellt somit ein System an n entkoppelten Differentialgleichungen in den modalen Koordinaten dar:

$$\dot{z}_k(t) = \lambda_k z_k(t) + \hat{b}_k u(t), \quad k = 1, \dots, n.\tag{4.2}$$

Durch Laplace-Transformation von (4.1) folgt

$$\begin{aligned} z_1(s) &= \frac{1}{s - \lambda_1} \hat{b}_1 u(s), \\ &\vdots \\ z_n(s) &= \frac{1}{s - \lambda_n} \hat{b}_n u(s), \\ y(s) &= \sum_{k=1}^n \hat{c}_k z_k(s), \end{aligned} \tag{4.3}$$

woraus sich das Strukturbild 4.1 mit n entkoppelten Pfaden leicht ablesen lässt.

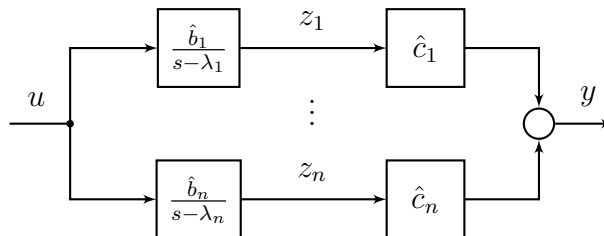


Abbildung 4.1: Strukturbild des Systems in Modalkoordinaten.

Außerdem nimmt die Übertragungsfunktion $G(s)$ die einfache Form

$$G(s) = \hat{\mathbf{c}} (s\mathbf{I} - \mathbf{\Lambda})^{-1} \hat{\mathbf{b}} = \sum_{k=1}^n \frac{\hat{c}_k \hat{b}_k}{s - \lambda_k}. \tag{4.4}$$

Zu jedem der n Eigenwerte λ_k und jeder Modalkoordinate z_k gehört also genau ein Pfad mit dem Anteil $\frac{\hat{c}_k \hat{b}_k}{s - \lambda_k}$ am Übertragungsverhalten.

 MATLAB Funktion(en): eig

4.2 Litzches Dominanzmaß

Um zu entscheiden, welche Modalkoordinaten bei einer Reduktion vernachlässigt werden dürfen, wird jedem Eigenwert λ_k eine positive reelle Zahl D_k zugewiesen, die ein Maß für dessen Bedeutung im Übertragungsverhalten (4.4) darstellt.

Wie man an den Summanden von (4.4) erkennt, wird eine alleinige Betrachtung des Eigenwertes λ_k oder seines Betrags zur Beurteilung seiner Dominanz nicht ausreichen. Vielmehr

müssen die Koeffizienten \hat{c}_k und \hat{b}_k (die gemäß Gilbert-Kriterium auch Aussagen zur Beobachtbarkeit und Steuerbarkeit von λ_k erlauben) einbezogen werden.

Aus diesem Grund hat Litz in den 1970er Jahren folgendes Dominanzmaß vorgeschlagen

$$D_k = \left| \frac{\hat{c}_k \hat{b}_k}{\lambda_k} \right|. \quad (4.5)$$

Dieses kann, gemäß Endwertsatz der Laplace-Transformation, als Beitrag des k -ten Pfades zum stationären Endwert von $y(t)$ interpretiert werden.

Es sei an dieser Stelle angemerkt, dass das Dominanzmaß nach Litz eine direkte Erweiterung für Mehrgrößensystemen ($m, p > 1$) besitzt, die auf den Einfluss eines Eigenwertes auf einzelne Übertragungspfade basiert. Eine ausführlichere Erklärung sowie weitere Dominanzmaße sind im Buch (Föllinger, 2013) zu finden.

4.3 Modales Abschneiden

Die Zustandsraumdarstellung in Modalkoordinaten (4.1) kann mithilfe des Dominanzmaßes neu angeordnet werden, sodass die Eigenwerte ihrer Dominanz nach, von hoch bis niedrig, auftreten:

$$\frac{d}{dt} \begin{bmatrix} z_1 \\ \vdots \\ z_q \\ z_{q+1} \\ \vdots \\ z_n \end{bmatrix} = \left[\begin{array}{ccc|ccc} \lambda_1 & & & & & \\ & \ddots & & & & \\ & & \lambda_q & & & \\ \hline & & & \lambda_{q+1} & & \\ & & & & \ddots & \\ \mathbf{0} & & & & & \lambda_n \end{array} \right] \begin{bmatrix} z_1 \\ \vdots \\ z_q \\ z_{q+1} \\ \vdots \\ z_n \end{bmatrix} + \begin{bmatrix} \hat{b}_1 \\ \vdots \\ \hat{b}_q \\ \hat{b}_{q+1} \\ \vdots \\ \hat{b}_n \end{bmatrix} u \quad (4.6)$$

$$y = \left[\hat{c}_1 \quad \dots \quad \hat{c}_q \mid \hat{c}_{q+1} \quad \dots \quad \hat{c}_n \right] \begin{bmatrix} z_1 \\ \vdots \\ z_q \\ z_{q+1} \\ \vdots \\ z_n \end{bmatrix}.$$

Ausgehend von der sortierten Darstellung (4.6) ergibt sich das reduzierte Modell durch Abschneiden des nichtdominanten Teilsystems zu

$$\begin{aligned} \dot{\mathbf{z}}_1(t) &= \mathbf{\Lambda}_1 \mathbf{z}_1(t) + \hat{\mathbf{b}}_1 u(t), \\ y(t) &= \hat{\mathbf{c}}_1 \mathbf{z}_1(t), \end{aligned} \quad (4.7)$$

entsprechend der Partitionierung $\mathbf{\Lambda}_1 = \text{diag}(\lambda_1, \dots, \lambda_q)$ usw.

Dabei ist die Wahl einer geeigneten reduzierten Ordnung q von Fall zu Fall unterschiedlich und orientiert sich i.d.R. entweder am relativen Abfall des Dominanzmaßes oder am absoluten Wert von D_k , welcher eine gewisse Toleranz unterschreiten sollte.

Modales Abschneiden als Projektion

Man beachte, dass die modale Reduktion auch ein projektives Modellreduktionsverfahren ist, denn das reduzierte Modell (4.7) kann auch direkt aus der ursprünglichen Zustandsraumdarstellung durch Projektion (3.18) gewonnen werden.

Dabei werden die Eigenvektormatrix \mathbf{T} und deren Inverse \mathbf{T}^{-1} spalten- bzw. zeilenweise betrachtet

$$\mathbf{T} = [\mathbf{t}_1 \quad \dots \quad \mathbf{t}_n] \quad \text{und} \quad \mathbf{T}^{-1} = \begin{bmatrix} \mathbf{l}_1^T \\ \vdots \\ \mathbf{l}_n^T \end{bmatrix}. \quad (4.8)$$

Die Projektionsmatrizen \mathbf{V} und \mathbf{W} ergeben sich dann aus der Selektion der Spalten bzw. Zeilen, welche zu den q dominantesten Eigenwerten gehören. Gilt bspw. $D_3 > D_1 > D_2 \gg D_k$ für alle weiteren Indizes $k=4, \dots, n$, so gilt für die Projektionsmatrizen

$$\mathbf{V} = [\mathbf{t}_3 \quad \mathbf{t}_1 \quad \mathbf{t}_2] \quad \text{und} \quad \mathbf{W}^T = \begin{bmatrix} \mathbf{l}_3^T \\ \mathbf{l}_1^T \\ \mathbf{l}_2^T \end{bmatrix}. \quad (4.9)$$

Entsprechend der Definition der Vektoren \mathbf{l}_i gilt

$$\mathbf{l}_i^T \mathbf{t}_j = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}, \quad (4.10)$$

insbesondere also $\mathbf{W}^T \mathbf{V} = \mathbf{I}_q$, sodass \mathbf{V} und \mathbf{W} biorthogonal sind.

Hinweis: Die Vektoren \mathbf{l}_k , $k = 1, \dots, n$, sind *Linkseigenvektoren* der Matrix \mathbf{A} , denn aus $\mathbf{A}\mathbf{T} = \mathbf{T}\mathbf{\Lambda}$ folgt

$$\mathbf{T}^{-1} \mathbf{A} = \mathbf{\Lambda} \mathbf{T}^{-1} \quad \Rightarrow \quad \mathbf{l}_k^T \mathbf{A} = \lambda_k \mathbf{l}_k^T. \quad (4.11)$$

Somit kann die modale Reduktion als Projektion auf den Unterraum der dominantesten (Rechts)Eigenvektoren, senkrecht auf den Unterraum der Linkseigenvektoren, gedeutet werden.

 sssMOR Funktion(en): modalMor

4.4 Reduktion instabiler Systeme

Die Dominanzanalyse aus Abschnitt 4.2 ist nur für asymptotisch stabile Eigenwerte sinnvoll, denn Eigenwerte rechts oder auf der imaginären Achse sind stets als dominant anzusehen und entsprechend in das reduzierte Modell zu übernehmen.

In der Tat gilt grundsätzlich für alle Modellreduktionsverfahren: Liegt ein nicht asymptotisch stabiles Modell vor, so sollte zunächst eine Aufspaltung durchgeführt werden, ähnlich einer modalen Reduktion, bei der das *asymptotisch stabile* Teilsystem vom *instabilen* Teilsystem getrennt wird

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} \mathbf{z}_s \\ \mathbf{z}_a \end{bmatrix} &= \begin{bmatrix} \mathbf{\Lambda}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_a \end{bmatrix} \begin{bmatrix} \mathbf{z}_s \\ \mathbf{z}_a \end{bmatrix} + \begin{bmatrix} \hat{\mathbf{b}}_s \\ \hat{\mathbf{b}}_a \end{bmatrix} u \\ y &= \begin{bmatrix} \hat{\mathbf{c}}_s & \hat{\mathbf{c}}_a \end{bmatrix} \begin{bmatrix} \mathbf{z}_s \\ \mathbf{z}_a \end{bmatrix}. \end{aligned} \quad (4.12)$$

Anschließend kann eine Reduktion des stabilen Teilsystems durchgeführt werden, beispielsweise mit modaler Reduktion, balanciertem Abschneiden oder Krylow-Unterraummethoden. Im Anschluss sollte das instabile Teilsystem im reduzierten Modell wieder integriert werden

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} \mathbf{z}_{s,r} \\ \mathbf{z}_a \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_{s,r} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_a \end{bmatrix} \begin{bmatrix} \mathbf{z}_{s,r} \\ \mathbf{z}_a \end{bmatrix} + \begin{bmatrix} \mathbf{b}_{s,r} \\ \mathbf{b}_a \end{bmatrix} u \\ y &= \begin{bmatrix} \mathbf{c}_{s,r} & \hat{\mathbf{c}}_a \end{bmatrix} \begin{bmatrix} \mathbf{z}_{s,r} \\ \mathbf{z}_a \end{bmatrix}. \end{aligned} \quad (4.13)$$



MATLAB Funktion(en): `stabsep`

4.5 Residualisierung

Ein Nachteil der modalen Reduktion besteht darin, dass i.A. keine stationäre Genauigkeit garantiert werden kann. Das heißt, der Amplitudengang bei der Frequenz $s = 0$ und somit der stationäre Endwert der Sprungantwort $y(t \rightarrow \infty)$ von original und reduziertem Modell können abweichen.

Da die stationäre Genauigkeit jedoch in vielen Anwendungen eine bedeutende Rolle spielt, kann diese durch das Verfahren der *Residualisierung* garantiert werden. Diese Strategie wird im Folgenden zusammengefasst und kann in (Antoulas, 2005, S.285) ausführlicher nachgelesen werden. Da das Problem der stationären Genauigkeit grundsätzlich bei Approximationen vorkommen kann, bei denen ein Teil der Zustandsvariablen abgeschnitten wird, wird es hier in allgemeiner Form vorgestellt.

Ausgangspunkt ist eine beliebige, partitionierte Zustandsraumdarstellung der Form

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} u \\ y &= \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix}, \end{aligned} \quad (4.14)$$

in der die relevante Dynamik durch die Zustandsrichtungen \mathbf{z}_1 dargestellt wird und die restliche Dynamik vernachlässigt werden soll. Der übliche Ansatz an dieser Stelle wäre der des *Abschneidens* (engl: *truncation*), das zum reduzierten Modell

$$\begin{aligned}\dot{\mathbf{z}}_1 &= \mathbf{A}_1 \mathbf{z}_1 + \mathbf{b}_1 u, \\ y_r &= \mathbf{c}_1 \mathbf{z}_1,\end{aligned}\tag{4.15}$$

führt, welches i.A. keine stationäre Genauigkeit aufweist.

Durch Vernachlässigung der Dynamik der Zustandsrichtungen in \mathbf{z}_2 , d.h. durch die Annahme $\dot{\mathbf{z}}_2 = 0$, kann der Teilzustandsvektor \mathbf{z}_2 als Funktion von \mathbf{z}_1 dargestellt werden

$$\mathbf{z}_2 \approx -\mathbf{A}_{22}^{-1} (\mathbf{A}_{21} \mathbf{z}_1 + \mathbf{b}_2 u),\tag{4.16}$$

wobei \mathbf{A}_{22} invertierbar sein muss. Einsetzen von \mathbf{z}_2 in (4.14) liefert letztendlich das reduzierte Modell durch *Residualisierung* (engl: *residualization*)

$$\begin{aligned}\dot{\mathbf{z}}_1 &= \overbrace{(\mathbf{A}_{11} - \mathbf{A}_{12} \mathbf{A}_{22}^{-1} \mathbf{A}_{21})}^{\mathbf{A}_r} \mathbf{z}_1 + \overbrace{(\mathbf{b}_1 - \mathbf{A}_{12} \mathbf{A}_{22}^{-1} \mathbf{b}_2)}^{\mathbf{b}_r} u, \\ y_r &= \underbrace{(\mathbf{c}_1 - \mathbf{c}_2 \mathbf{A}_{22}^{-1} \mathbf{A}_{21})}_{\mathbf{c}_r} \mathbf{z}_1 + \underbrace{(-\mathbf{c}_2 \mathbf{A}_{22}^{-1} \mathbf{b}_2)}_{d_r} u,\end{aligned}\tag{4.17}$$

und es gilt

$$G(0) = \mathbf{c} (-\mathbf{A})^{-1} \mathbf{b} = \mathbf{c}_r (-\mathbf{A}_r)^{-1} \mathbf{b}_r + d_r = G_r(0)\tag{4.18}$$

Der Beweis folgt direkt aus der Inversion von Block-Matrizen und kann in (Fernando und Nicholson, 1982) nachgelesen werden. Man beachte, dass durch die Residualisierung die stationäre Genauigkeit erreicht wird, allerdings i.A. auf Kosten der Genauigkeit bei hohen Frequenzen:

$$G(s \rightarrow \infty) = \lim_{s \rightarrow \infty} \mathbf{c} (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{b} = 0 \neq d_r = G_r(s \rightarrow \infty)\tag{4.19}$$

4.6 Eigenschaften der modalen Reduktion

- + Das reduzierte System liegt in Modalkoordinaten vor.
- + Aus der Stabilität des Originalsystems folgt auch Stabilität des reduzierten Systems, da die Eigenwerte des reduzierten Systems eine Teilmenge der Eigenwerte des Originalsystems sind.
- + Eigenwerte und Modalkoordinaten bleiben exakt erhalten. Dies ist v.a. von Vorteil, wenn bestimmte Eigenwerte und deren Eigenbewegungen von besonderer Bedeutung sind und unbedingt erhalten bleiben sollten (bspw. instabile Eigenwerte).
- + Der reduzierte Zustandsvektor besitzt eine physikalische Bedeutung, er beschreibt nämlich die Richtung entlang bestimmter Modalkoordinaten.

- Die Wahl der reduzierten Koordinaten ist beschränkt auf die Menge der Eigenvektoren des Originalmodells. Dies kann teilweise eine Einschränkung darstellen, insb. wenn andere Zustandsrichtungen eine größere Rolle im Übertragungsverhalten spielen (s. Übung).
- Das Litzsche Dominanzmaß betrachtet nur den Betrag des Beitrags einzelner Eigenwerte zum Übertragungsverhalten. Es gibt aber Systeme, bei denen Kompensationseffekte auftreten, die von diesem Maß nicht erfasst werden können. In diesen Fällen kann das Reduktionsergebnis unzufriedenstellend werden.
- Der Hauptnachteil der Methode ist vor allem der hohe Speicherbedarf $\mathcal{O}(n^2)$. Selbst wenn die Systemmatrix \mathbf{A} dünn besetzt ist, ist die Matrix der Eigenvektoren \mathbf{T} i.A. vollbesetzt. Dazu kommt noch der hohe Rechenaufwand $\mathcal{O}(n^3)$ für die Lösung des Eigenwertproblems (Schur Zerlegung, vgl. (Golub und Van Loan, 1996)), weshalb das Verfahren für sehr hochdimensionale Modelle nicht ohne Weiteres angewandt werden kann.

Modale Reduktion wie in diesem Kapitel vorgestellt eignet sich nur für mittelgroße Modelle. Es existieren allerdings Erweiterungen auf hochdimensionale, dünnbesetzte Modelle. Dabei wird keine volle Eigenzerlegung $\mathbf{T}^{-1}\mathbf{A}\mathbf{T} = \mathbf{\Lambda}$ berechnet, denn die Matrix $\mathbf{T} \in \mathbb{C}^{n \times n}$ ist i.A. vollbesetzt! Stattdessen können mittels *Powermethoden* oder *Krylow-Verfahren* iterativ Eigenräume zu bestimmten Eigenwerten berechnet werden, beispielsweise solche mit größtem oder geringsten Betrag. Man vergleiche hierzu auch die MATLAB Funktionen `eig`, `eigs` sowie die Bücher (Demmel, 1997) und (Saad, 2003).



MATLAB Funktion(en): `eigs`



SSS Funktion(en): `eigs`



SSSMOR Funktion(en): `modalMor`

Kapitel 5

Balanciertes Abschneiden

Engl.: Balanced Truncation (BT) oder Truncated Balanced Realization (TBR)

In diesem Kapitel lernen wir eine weit verbreitete Methode zur Modellreduktion kennen, welche ihren Ursprung in den 1980ern hat. Es handelt sich um das sogenannte *Balancierte Abschneiden*: das System wird zunächst durch eine Zustandstransformation auf balancierte Darstellung gebracht, in der die „unwichtigen Zustände“ leicht identifiziert werden können; durch anschließendes Abschneiden erhält man das reduzierte System.

Die Grundidee des TBR lässt sich so formulieren: Finde eine Zustandsdarstellung, in der jeder einzelne Zustand x_i genauso stark bzw. schwach steuerbar ist, wie er beobachtbar ist (= Balancieren). Vernachlässige anschließend die Zustände, die den geringsten Beitrag zum Übertragungsverhalten leisten (= Abschneiden).

Die Begriffe Steuerbarkeit und Beobachtbarkeit werden in der folgenden Gegenüberstellung wiederholt.

Steuerbarkeit

- Zusammenhang zwischen Eingang $u(t)$ und Zustand $\mathbf{x}(t)$
- Matrizen \mathbf{A} und \mathbf{b}
- Vollständige Steuerbarkeit: das System kann durch geeignetes $u(t)$ in endlicher Zeit aus dem Anfangszustand $\mathbf{x}(t_0) = \mathbf{0}$ in einen beliebigen Endzustand $\mathbf{x}(t) = \mathbf{x}_e$ überführt werden.

Beobachtbarkeit

- Zusammenhang zwischen Zustand $\mathbf{x}(t)$ und Ausgang $y(t)$
- Matrizen \mathbf{A} und \mathbf{c}
- Vollständige Beobachtbarkeit: es sei $u(t)$ bekannt; dann kann aus der alleinigen Messung von $y(t)$ über eine endliche Zeitspanne der Anfangszustand $\mathbf{x}(t_0) = \mathbf{x}_0$ eindeutig ermittelt werden.

5.1 Steuerbarkeit

Welche Zustände sind gut steuerbar? Um Steuerbarkeit beurteilen zu können, benötigen wir ein geeignetes Maß; das soll im Folgenden die Energie¹ sein: die Zustände $\mathbf{x}(t_e) = \mathbf{x}_e$, die mit wenig Energie aus dem Ursprung $\mathbf{x}(t=0) = \mathbf{0}$ erreicht werden können, sind gut/leicht steuerbar. Um diese Zustände identifizieren zu können, nehmen wir ein asymptotisch stabiles System an, welches sich in der Ruhelage $\mathbf{x} = \mathbf{0}$ befindet. Um das System in den Endzustand \mathbf{x}_e zu bewegen, wird Stellenergie benötigt. Das Ziel ist es deshalb, die minimale Energie zu bestimmen, die notwendig ist um das System in den Endzustand \mathbf{x}_e zu überführen.

Diese Aufgabenstellung lässt sich als Optimierungsproblem (wie aus Moderne Methoden 2 bekannt) formulieren:

$$\begin{aligned} \text{Gütemaß:} \quad & J = \int_0^{t_e} u^2(t) \, dt \\ \text{mit der Nebenbedingung:} \quad & \dot{\mathbf{x}}(t) = \mathbf{A} \mathbf{x}(t) + \mathbf{b} u(t) \\ \text{und den Randbedingungen:} \quad & \mathbf{x}(t=0) = \mathbf{0} \quad \text{und} \quad \mathbf{x}(t=t_e) = \mathbf{x}_e, \end{aligned}$$

wobei t_e frei und somit Teil der Optimierung ist. Um dies zu umgehen, könnten wir $t_e \rightarrow \infty$ wählen; dann würde aber das Integral nicht mehr existieren, da ständig Stellenergie benötigt wird um das System in \mathbf{x}_e zu halten. Stattdessen müssen wir einen Trick anwenden, wir betrachten das Problem in negativer Zeitrichtung: $\tau = t_e - t$; $\frac{d\tau}{dt} = -1 \Leftrightarrow dt = -d\tau$. Dadurch ändert sich das Vorzeichen auf der rechten Seite der DGL und das Gütemaß lautet:

$$J = \int_{t=0}^{t=t_e} u^2(t) \, dt = \int_{\tau=t_e}^{\tau=0} -u^2(\tau) \, d\tau = \int_0^{t_e} u^2(\tau) \, d\tau. \quad (5.1)$$

Für die Zeiten $\tau > t_e$ bleibt das System ohne Stellenergie in der Ruhelage, weshalb wir nun genauso gut $t_e \rightarrow \infty$ setzen können um die Optimierungsvariable t_e zu eliminieren. Damit ergibt sich das Optimierungsproblem in bekannter Form:

$$\begin{aligned} \text{Gütemaß:} \quad & J = \int_0^{\infty} u^2(\tau) \, d\tau \\ \text{mit der Nebenbedingung:} \quad & \dot{\mathbf{x}}(\tau) = -\mathbf{A} \mathbf{x}(\tau) - \mathbf{b} u(\tau) \\ \text{und den Randbedingungen:} \quad & \mathbf{x}(\tau=0) = \mathbf{x}_e \quad \text{und} \quad \mathbf{x}(\tau \rightarrow \infty) = \mathbf{0}. \end{aligned}$$

Die Lösung liefert das optimale Regelgesetz (siehe Moderne Methoden 2):

$$u_{\text{opt}}(\tau) = -\frac{1}{2} (-\mathbf{b}^T) \mathbf{P} \mathbf{x}(\tau) = \frac{1}{2} \mathbf{b}^T \mathbf{P} \mathbf{x}(\tau), \quad (5.2)$$

wobei \mathbf{P} die Lösung einer Riccati-Gleichung ist. Allerdings findet im Gütemaß keine Gewichtung $\mathbf{x}^T \mathbf{Q} \mathbf{x}$ der Zustände statt. Der entsprechende Term entfällt somit, und die Riccati-

¹Es sei angemerkt, dass die hier als Energie bezeichnete Größe eigentlich als *generalisierte Energie* aufzufassen ist und bspw. nicht die Einheit Joule besitzen muss. Sie soll ein Maß für die benötigte Stellenergie sein, vgl. z.B. Gleichung (5.1).

Gleichung wird zur Ljapunow-Gleichung:

$$(-\mathbf{A}^T)\mathbf{P} + \mathbf{P}(-\mathbf{A}) - \frac{1}{2}\mathbf{P}(-\mathbf{b})(-\mathbf{b}^T)\mathbf{P} = \mathbf{0}, \quad (5.3)$$

$$\Leftrightarrow -\mathbf{A}^T\mathbf{P} - \mathbf{P}\mathbf{A} - \frac{1}{2}\mathbf{P}\mathbf{b}\mathbf{b}^T\mathbf{P} = \mathbf{0}, \quad (5.4)$$

$$\Leftrightarrow 2\mathbf{P}^{-1}\mathbf{A}^T + 2\mathbf{A}\mathbf{P}^{-1} + \mathbf{b}\mathbf{b}^T = \mathbf{0}. \quad (5.5)$$

5.1.1 Gram'sche Steuerbarkeitsmatrix

Definition 18. Wir definieren $\mathbf{W}_c := 2\mathbf{P}^{-1}$ als die Gram'sche Steuerbarkeitsmatrix (engl: Controllability Gramian). Sie ist Lösung der Ljapunow-Gleichung:

$$\mathbf{A}\mathbf{W}_c + \mathbf{W}_c\mathbf{A}^T + \mathbf{b}\mathbf{b}^T = \mathbf{0}. \quad (5.6)$$

Weiterhin kann man zeigen (siehe Übung), dass gilt:

$$\mathbf{W}_c = \int_0^\infty e^{\mathbf{A}t}\mathbf{b}\mathbf{b}^T e^{\mathbf{A}^T t} dt. \quad (5.7)$$



MATLAB Funktion(en): `gram(sys, 'c')`, `lyap`, `lyapchol`

5.1.2 Energiebetrachtung

Satz 4. Die Gram'sche Steuerbarkeitsmatrix gibt den minimalen Wert des Gütemaßes $J^*(\mathbf{x}_e)$ in Abhängigkeit des Endzustandes \mathbf{x}_e an:

$$J^*(\mathbf{x}_e) = \mathbf{x}_e^T \mathbf{W}_c^{-1} \mathbf{x}_e. \quad (5.8)$$

Beweis. Wird das optimale Regelgesetz $\mathbf{u}(\tau) = \frac{1}{2}\mathbf{b}^T\mathbf{P}\mathbf{x}(\tau) = \mathbf{b}^T\mathbf{W}_c^{-1}\mathbf{x}(\tau)$ in die Zustandsgleichung eingesetzt, ergibt sich die Dynamik:

$$\dot{\mathbf{x}}(\tau) = (-\mathbf{A} - \mathbf{b}\mathbf{b}^T\mathbf{W}_c^{-1})\mathbf{x}(\tau). \quad (5.9)$$

Weiterhin wird noch folgender Term benötigt:

$$\frac{d}{dt} (\mathbf{x}^T(\tau)\mathbf{W}_c^{-1}\mathbf{x}(\tau)) = \dot{\mathbf{x}}^T(\tau)\mathbf{W}_c^{-1}\mathbf{x}(\tau) + \mathbf{x}^T(\tau)\mathbf{W}_c^{-1}\dot{\mathbf{x}}(\tau) \quad (5.10)$$

$$\stackrel{(5.9)}{=} \mathbf{x}^T(\tau) (-\mathbf{A}^T\mathbf{W}_c^{-1} - \mathbf{W}_c^{-1}\mathbf{A} - 2\mathbf{W}_c^{-1}\mathbf{b}\mathbf{b}^T\mathbf{W}_c^{-1})\mathbf{x}(\tau) \quad (5.11)$$

$$= -\mathbf{x}^T(\tau)\mathbf{W}_c^{-1} (\mathbf{W}_c\mathbf{A}^T + \mathbf{A}\mathbf{W}_c + 2\mathbf{b}\mathbf{b}^T) \mathbf{W}_c^{-1}\mathbf{x}(\tau) \quad (5.12)$$

$$\stackrel{(5.6)}{=} -\mathbf{x}^T(\tau)\mathbf{W}_c^{-1}\mathbf{b}\mathbf{b}^T\mathbf{W}_c^{-1}\mathbf{x}(\tau) \quad (5.13)$$

Der Term $\mathbf{x}^T(\tau)\mathbf{W}_c^{-1}\mathbf{x}(\tau)$ ist somit Stammfunktion von (5.13), und es folgt:

$$J^*(\mathbf{x}_e) = \int_0^\infty u_{\text{opt}}^2(\tau) \, d\tau = \int_0^\infty u_{\text{opt}}^T(\tau)u_{\text{opt}}(\tau) \, d\tau \quad (5.14)$$

$$\stackrel{(5.2)}{=} \int_0^\infty \mathbf{x}^T(\tau)\mathbf{W}_c^{-1}\mathbf{b}\mathbf{b}^T\mathbf{W}_c^{-1}\mathbf{x}(\tau)d\tau \stackrel{(5.13)}{=} [-\mathbf{x}^T(\tau)\mathbf{W}_c^{-1}\mathbf{x}(\tau)]_0^\infty \quad (5.15)$$

$$= -\mathbf{x}^T(\tau \rightarrow \infty)\mathbf{W}_c^{-1}\mathbf{x}(\tau \rightarrow \infty) + \mathbf{x}^T(0)\mathbf{W}_c^{-1}\mathbf{x}(0) = 0 + \mathbf{x}_e^T\mathbf{W}_c^{-1}\mathbf{x}_e \quad (5.16)$$

□

Das Produkt $\mathbf{x}_e^T\mathbf{W}_c^{-1}\mathbf{x}_e$ beschreibt also die minimale Energie, die notwendig ist um den Endzustand \mathbf{x}_e zu erreichen. Dieses Ergebnis erlaubt die folgende Interpretation: Nach Singulärwertzerlegung von $\mathbf{W}_c = \mathbf{U}\Sigma\mathbf{U}^T$ mit $\sigma_1 \geq \sigma_2 \dots \geq \sigma_n$ (bzw. Eigenzerlegung, da $\mathbf{W}_c = \mathbf{W}_c^T > \mathbf{0}$) gilt:

- Die ersten Singulärvektoren \mathbf{u}_i zeigen in die Richtungen, die „einfach“ zu steuern sind: σ_1 ist maximal, und mit $\mathbf{W}_c^{-1} = \mathbf{U}\Sigma^{-1}\mathbf{U}^T$ folgt, dass $\mathbf{u}_1^T\mathbf{W}_c^{-1}\mathbf{u}_1 = \frac{1}{\sigma_1}$ minimal ist.
- Die entsprechenden Singulärwerte σ_i beschreiben, mit welchem Energieaufwand man diese Richtungen im Zustandsraum erreichen kann.

Man vergleiche hierzu auch die Ausführung in (Lunze, 2010, Kap.3.1.2).

5.2 Beobachtbarkeit

Welche Zustände sind gut beobachtbar? Auch die Beobachtbarkeit soll anhand der Energie beurteilt werden: die Zustände $\mathbf{x}(t=0) = \mathbf{x}_0$, die für $u(t) \equiv 0$ im Ausschwingvorgang die meiste Energie am Ausgang liefern, sind gut beobachtbar. Um diese Zustände identifizieren zu können, definieren wir die Gram'sche Beobachtbarkeitsmatrix.

5.2.1 Gram'sche Beobachtbarkeitsmatrix

Definition 19. Wir definieren die Lösung $\mathbf{W}_o = \mathbf{W}_o^T > \mathbf{0}$ der Ljapunow-Gleichung:

$$\mathbf{A}^T\mathbf{W}_o + \mathbf{W}_o\mathbf{A} + \mathbf{c}^T\mathbf{c} = \mathbf{0}, \quad (5.17)$$

als Gram'sche Beobachtbarkeitsmatrix (engl: Observability Gramian). Analog gilt:

$$\mathbf{W}_o = \int_0^\infty e^{\mathbf{A}^T t}\mathbf{c}^T\mathbf{c}e^{\mathbf{A}t} dt. \quad (5.18)$$



MATLAB Funktion(en): `gram(sys, 'o')`, `lyap`, `lyapchol`

5.2.2 Energiebetrachtung

Die Gram'sche Beobachtbarkeitsmatrix beschreibt, wie gut oder schlecht die einzelnen Zustände beobachtbar sind. Dieser Zusammenhang ergibt sich aus den folgenden Betrachtungen. Das unaktuierte System lautet:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A} \mathbf{x}(t) \\ y(t) &= \mathbf{c} \mathbf{x}(t).\end{aligned}\tag{5.19}$$

Die Lösung $y(t) = \mathbf{c} \mathbf{x}(t)$ mit dem Anfangswert $\mathbf{x}(t=0) = \mathbf{x}_0$ ist gegeben durch:

$$y(t) = \mathbf{c} e^{\mathbf{A}t} \mathbf{x}_0.\tag{5.20}$$

Damit liefert der Anfangszustand \mathbf{x}_0 die folgende Energie am Ausgang (gemessen in der \mathcal{L}_2 -Norm):

$$\|y(t)\|_{\mathcal{L}_2}^2 = \int_0^\infty y^T(t) y(t) dt\tag{5.21}$$

$$\stackrel{(5.20)}{=} \mathbf{x}_0^T \int_0^\infty \left(e^{\mathbf{A}^T t} \mathbf{c}^T \mathbf{c} e^{\mathbf{A}t} \right) dt \mathbf{x}_0\tag{5.22}$$

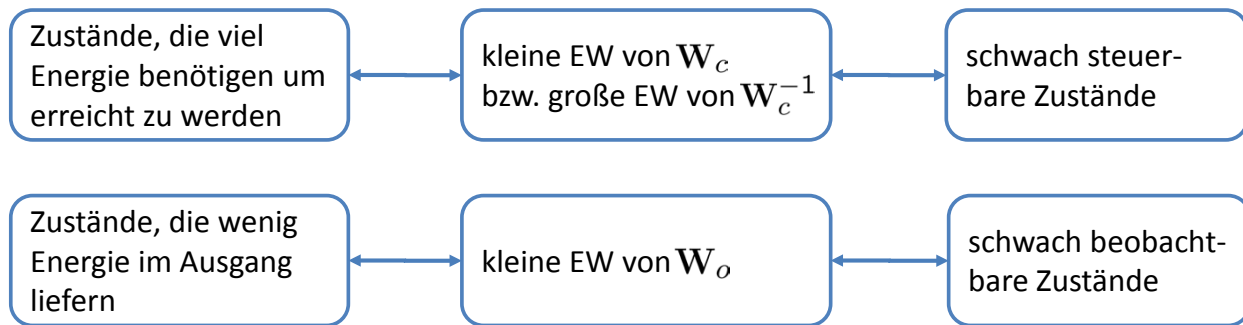
$$\stackrel{(5.18)}{=} \mathbf{x}_0^T \mathbf{W}_o \mathbf{x}_0.\tag{5.23}$$

Das Produkt $\mathbf{x}_0^T \mathbf{W}_o \mathbf{x}_0$ beschreibt also die Energie, welche der Zustand \mathbf{x}_0 durch Beobachtung am Ausgang liefert. Nach der Singulärwertzerlegung von \mathbf{W}_o erlaubt dies die analoge Interpretation:

- Die ersten Singulärvektoren zeigen in die Richtungen, die die meiste Energie im Ausgang erzeugen, also „stark“ beobachtbar sind.
- Die entsprechenden Eigenwerte beschreiben, wie groß diese Energie ist.

5.3 Balancierung

Die vorherigen Betrachtungen zum Energiefluss im System lassen sich in folgender Grafik zusammenfassen:



Wie aus den Überlegungen aus Übung 1 folgt, dürfen nur solche Zustände vernachlässigt werden, die sowohl schlecht steuerbar als auch schlecht beobachtbar sind. Es müssen also die Informationen von \mathbf{W}_c und \mathbf{W}_o gleichzeitig berücksichtigt werden! Eine Richtung, die z. B. gut steuerbar ist, muss aber nicht zwangsweise mit einer gut beobachtbaren Richtung zusammenfallen. Um aber Richtungen zu finden, die genauso gut steuer- wie beobachtbar sind, müssen wir eine Zustandstransformation der Art $\mathbf{z} = \mathbf{T}\mathbf{x}$ einführen, da hierdurch Linearkombinationen der Zustände \mathbf{x} als neue Zustände \mathbf{z} eingeführt werden. Das führt zunächst auf die Frage: Wie ändern sich die Gram'schen Matrizen bei einer solchen Zustandstransformation?

Satz 5. Seien $\mathbf{W}_c, \mathbf{W}_o$ die Gram'schen Systemmatrizen in den Koordinaten \mathbf{x} . Die transformierten Gram'schen $\widehat{\mathbf{W}}_c, \widehat{\mathbf{W}}_o$ in den Koordinaten $\mathbf{z} = \mathbf{T}\mathbf{x}$ lauten:

$$\widehat{\mathbf{W}}_c = \mathbf{T}\mathbf{W}_c\mathbf{T}^T, \quad (5.24)$$

$$\widehat{\mathbf{W}}_o = \mathbf{T}^{-T}\mathbf{W}_o\mathbf{T}^{-1}. \quad (5.25)$$

(Beweis siehe Übung.)

Das bedeutet, dass sich die Eigenwerte von \mathbf{W}_c und \mathbf{W}_o mit der Darstellung des Systems ändern! Allerdings erkennt man, dass die Eigenwerte des Produktes $\mathbf{W}_c\mathbf{W}_o$ invariant gegenüber Zustandstransformationen sind,

$$\widehat{\mathbf{W}}_c\widehat{\mathbf{W}}_o = \mathbf{T}(\mathbf{W}_c\mathbf{W}_o)\mathbf{T}^{-1}. \quad (5.26)$$

Die Eigenwerte von $\mathbf{W}_c\mathbf{W}_o$ sind somit unabhängig von der Darstellung des Systems. Sie sind unveränderliche Systemgrößen zur Beschreibung des Übertragungsverhaltens.

Definition 20. Die Hankel-Singulärwerte σ_i (Engl.: Hankel Singular Values, HSV) eines Systems sind definiert als

$$\sigma_i = \sqrt{\lambda_i(\mathbf{W}_c\mathbf{W}_o)}, \quad (5.27)$$

und stellen ein Maß für den Energietransfer von den Eingängen zu den Ausgängen dar. Sie sind unabhängig von der Zustandsdarstellung.

Zur Erinnerung: Wir suchen Zustandsrichtungen, welche sowohl schlecht steuerbar als auch schlecht beobachtbar sind! Die Lösung liefert die *balancierte Darstellung*, welche mit Hilfe der HSVs definiert werden kann.

Definition 21. Ein System heißt **balanciert**, wenn gilt: $\mathbf{W}_c = \mathbf{W}_o = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$ mit $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$. Das bedeutet: Die beiden Gram'schen Systemmatrizen sind gleich, liegen in Diagonalform vor und haben die HSVs als Einträge.

In einem balanciertem System sind somit alle Zustände genauso gut steuerbar wie beobachtbar. Eine mögliche Berechnung der Zustandstransformation \mathbf{T} , welche das System auf balancierte Darstellung transformiert, wird in folgendem Satz angegeben.

Satz 6. Die Zustandstransformation \mathbf{T} auf balancierte Darstellung kann folgendermaßen berechnet werden:

1. Cholesky-Zerlegung² von $\mathbf{W}_c = \mathbf{S}\mathbf{S}^T$ und $\mathbf{W}_o = \mathbf{R}\mathbf{R}^T$
2. SVD von $\mathbf{R}^T\mathbf{S} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$
3. $\mathbf{T} := \mathbf{\Sigma}^{-1/2}\mathbf{U}^T\mathbf{R}^T$ und $\mathbf{T}^{-1} := \mathbf{S}\mathbf{V}\mathbf{\Sigma}^{-1/2}$

Hinweis: Durch diese Transformation verlieren die Zustände ihre physikalische Bedeutung!

Beweis.

$$\begin{aligned} \widehat{\mathbf{W}}_c &= \mathbf{T}\mathbf{W}_c\mathbf{T}^T \quad \text{mit } \mathbf{W}_c = \mathbf{S}\mathbf{S}^T \\ &= \mathbf{\Sigma}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{R}^T(\mathbf{S}\mathbf{S}^T)\mathbf{R}\mathbf{U}\mathbf{\Sigma}^{-\frac{1}{2}} \\ &= \mathbf{\Sigma}^{-\frac{1}{2}}\underbrace{\mathbf{U}^T\mathbf{U}}_{=\mathbf{I}}\mathbf{\Sigma}\underbrace{\mathbf{V}^T\mathbf{V}}_{=\mathbf{I}}\mathbf{\Sigma}\underbrace{\mathbf{U}^T\mathbf{U}}_{=\mathbf{I}}\mathbf{\Sigma}^{-\frac{1}{2}} \\ &= \mathbf{\Sigma}^{-\frac{1}{2}}\mathbf{\Sigma}^2\mathbf{\Sigma}^{-\frac{1}{2}} = \mathbf{\Sigma}. \end{aligned}$$

Der Beweis für $\widehat{\mathbf{W}}_o = \mathbf{\Sigma}$ erfolgt analog. □

In der balancierten Darstellung sind die Zustände ihrer Wichtigkeit zum Übertragungsverhalten nach geordnet. Damit haben wir ein Maß gefunden, welche Zustände wir im reduzierten System behalten wollen, und welche wir vernachlässigen können.



MATLAB Funktion(en): `hsvd`, `hsvplot`, `balreal`

5.4 Balanciertes Abschneiden

Ist die balancierte Darstellung des Systems gefunden (d. h. nach Zustandstransformation mit \mathbf{T}), kann schlussendlich die Reduktion vorgenommen werden. Das balancierte System wird

²Die *Cholesky*-Zerlegung ist nur für symmetrische positiv definite Matrizen definiert. Die Faktoren \mathbf{S} , \mathbf{R} sind dabei untere Dreiecksmatrizen, vgl. (Golub und Van Loan, 1996).

Algorithmus 1 Ordnungsreduktion durch balanciertes Abschneiden

1. Lösen der Ljapunow-Gleichungen:

$$\begin{aligned}\mathbf{A}\mathbf{W}_c + \mathbf{W}_c\mathbf{A}^T + \mathbf{b}\mathbf{b}^T &= \mathbf{0}, \\ \mathbf{A}^T\mathbf{W}_o + \mathbf{W}_o\mathbf{A} + \mathbf{c}^T\mathbf{c} &= \mathbf{0}.\end{aligned}$$

2. Cholesky-Zerlegungen: $\mathbf{W}_c = \mathbf{S}\mathbf{S}^T$, $\mathbf{W}_o = \mathbf{R}\mathbf{R}^T$

3. SVD: $\mathbf{R}^T\mathbf{S} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_{i=1}^n \sigma_i \mathbf{u}_i \mathbf{v}_i^T$

4. Die ersten q Spalten von \mathbf{U} und \mathbf{V} wählen, $\mathbf{U}_q = [\mathbf{u}_1, \dots, \mathbf{u}_q]$, $\mathbf{V}_q = [\mathbf{v}_1, \dots, \mathbf{v}_q]$, und die ersten Hankel-Singulärwerte wählen: $\mathbf{\Sigma}_q = \text{diag}(\sigma_1, \dots, \sigma_q)$. Projektionsmatrizen \mathbf{V}_B und \mathbf{W}_B berechnen:

$$\begin{aligned}\mathbf{V}_{BA} &:= \mathbf{S}\mathbf{V}_q\mathbf{\Sigma}_q^{-1/2} \in \mathbb{R}^{n \times q}, \\ \mathbf{W}_{BA}^T &:= \mathbf{\Sigma}_q^{-1/2}\mathbf{U}_q^T\mathbf{R}^T \in \mathbb{R}^{q \times n}.\end{aligned}$$

5. Da \mathbf{V}_{BA} und \mathbf{W}_{BA} biorthogonal sind, lautet das reduzierte System der Ordnung $q \ll n$:

$$\mathbf{A}_r = \mathbf{W}_{BA}^T \mathbf{A} \mathbf{V}_{BA}, \quad \mathbf{b}_r = \mathbf{W}_{BA}^T \mathbf{b}, \quad \mathbf{c}_r = \mathbf{c} \mathbf{V}_{BA}.$$

5.5 Eigenschaften des balancierten Abschneidens

- + Das reduzierte System ist ebenfalls balanciert und eine minimale Realisierung.
- + Aus der Stabilität des Originalsystems folgt auch Stabilität des reduzierten Systems: Die reduzierten Gram'schen sind diagonal und haben die größten HSV's als Einträge \Rightarrow positiv definite Ljapunow-Lösungen \Rightarrow Stabilitätsbeweis nach Ljapunow.
- + Man kann zeigen, dass gilt:

$$\|G(s) - G_r(s)\|_{\mathcal{H}_\infty} \leq 2 \sum_{i=q+1}^n \sigma_i. \quad (5.31)$$

Der Fehler bezüglich der \mathcal{H}_∞ -Norm ist also beschränkt durch die doppelte Summe der abgeschnittenen Hankel-Singulärwerte. Das bedeutet, dass a priori – also vor Berechnung des reduzierten Systems – durch die Wahl der reduzierten Ordnung q der Approximationsfehler beschränkt werden kann. Dies ist ein erheblicher Vorteil des Verfahrens, denn die zuverlässige Schätzung des Approximationsfehlers ist eine der schwierigsten Fragestellungen bei der Modellreduktion.

- Der Hauptnachteil der TBR-Methode ist vor allem der hohe Speicherbedarf $\mathcal{O}(n^2)$. Selbst wenn die Systemmatrix \mathbf{A} dünn besetzt ist sind die Gram'sche Matrizen und deren Cholesky Faktoren i.A. vollbesetzt. Dazu kommt noch der hohe Rechenaufwand $\mathcal{O}(n^3)$ für die Lösung der Ljapunow-Gleichungen und der SVD, weshalb das Verfahren für sehr hochdimensionale Modelle nicht ohne Weiteres angewandt werden kann.

Beispiel: Bei Erhöhung der Ordnung des Originalsystems von $n = 500$ auf $n = 1000$ (Faktor 2) benötigt das balancierte Abschneiden ungefähr $4\times$ so viel Speicher und $8\times$ so viel Zeit. Auf einem heutigem Standardrechner ist TBR bis maximal zur Ordnung $n \approx 3000$ möglich, denn die Gram'schen Systemmatrizen benötigen dann bereits über 7 GB an Speicher!

Balanciertes Abschneiden aus Algorithmus 1 eignet sich nur für mittelgroße Modelle. Es existieren allerdings Erweiterungen auf hochdimensionale, dünnbesetzte Modelle, wie z.B. die *Low Rank Square Root BT* Methode. Der zentrale Unterschied besteht in der approximativen Lösung $\mathbf{W}_c \approx \mathbf{Z}_c \mathbf{Z}_c^T$ und $\mathbf{W}_o \approx \mathbf{Z}_o \mathbf{Z}_o^T$ der beiden Ljapunow-Gleichungen mittels Niedrigrang-Faktoren der Dimension $\mathbf{Z}_c \in \mathbb{R}^{n \times q_c}$, $\mathbf{Z}_o \in \mathbb{R}^{n \times q_o}$ und $q_c, q_o \ll n$. Die wichtigsten Verfahren in dieser Hinsicht sind das *Low Rank Alternating Direction Implicit* (LR-ADI) (s. Penzl, 2000) oder das *Rational Krylov Subspace Method* (RKSM) (s. Simoncini, 2007). Beide Typen von Verfahren stehen in sehr enger Beziehung zu den Krylow-Unterraummethoden, die im nächsten Abschnitt vorgestellt werden. Es sei der Vollständigkeit halber angemerkt, dass durch diese Approximation die oben beschriebenen Vorteile des Balancierten Abschneidens i.A. verloren gehen.

Kapitel 6

Krylow-Unterraummethoden

In diesem Kapitel lernen wir eine weitere wichtige Methode zur Modellreduktion kennen: die *Krylow-Unterraummethoden*. Die grundlegende Idee ist die *lokale* Approximation des Übertragungsverhaltens. So wie eine beliebige Funktion $f(x)$ durch eine Taylor-Reihe um einen Punkt x_0 angenähert werden kann, so wollen wir auch die Übertragungsfunktion $G(s)$ approximieren. Dadurch erhalten wir zwar ein mächtiges Werkzeug, welches aber nicht einfach zu bedienen ist! Professor Athanasios C. Antoulas, eine Koryphäe auf dem Gebiet, drückt dies ungefähr so aus:

Der Vorteil von Krylow-Unterraummethoden sind die vielen Freiheiten.

Der Nachteil von Krylow-Unterraummethoden sind die vielen Freiheiten.

Das Problem ist, dass nicht immer genau klar ist, wie die Freiheiten der Methode sinnvoll zu wählen sind. Deshalb sind die Krylow-Unterraummethoden noch immer Gegenstand intensiver Forschung. Im Englischen wird die Methode auch *Moment Matching* genannt.

6.1 Moment Matching

Die Übertragungsfunktion $G(s)$ eines Systems ist eine rationale Funktion des Laplace-Operators s und kann in eine unendliche Taylorreihe um eine (komplexe) Frequenz s_0 mit den (noch zu bestimmenden) Koeffizienten m_i entwickelt werden:

$$G(s) = -m_0 - m_1(s - s_0) - m_2(s - s_0)^2 - \dots = - \sum_{i=0}^{\infty} m_i (s - s_0)^i \quad (6.1)$$

Das Minus vor den m_i ist eine Vorzeichenkonvention. Das Ziel des Moment Matching ist ein reduziertes System zu finden, so dass die ersten q Koeffizienten m_i , $i = 0, 1, \dots, (q - 1)$, der Taylorreihe von originalem und reduziertem System übereinstimmen. Es handelt sich also um eine lokale Approximation um die Frequenz s_0 .

6.1.1 Momente

Definition 22. Die Momente m_i einer Übertragungsfunktion (bzw. eines dynamischen Systems) um einen Entwicklungspunkt s_0 sind definiert als die negativen Koeffizienten der Taylorreihendarstellung um s_0 :

$$G(s) = - \sum_{i=0}^{\infty} m_i (s - s_0)^i, \quad (6.2)$$

oder anders ausgedrückt:

$$m_i = - \frac{1}{i!} \left. \frac{d^i G(s)}{ds^i} \right|_{s=s_0} \quad i = 0, 1, \dots \quad (6.3)$$

Für ein dynamisches System in Zustandsraumdarstellung gilt:

$$m_i = \mathbf{c} (\mathbf{A} - s_0 \mathbf{I})^{-(i+1)} \mathbf{b}. \quad (6.4)$$

Beweis. Die Übertragungsfunktion lautet

$$\begin{aligned} G(s) &= \mathbf{c} (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{b} = \mathbf{c} (s\mathbf{I} - \mathbf{A} + s_0\mathbf{I} - s_0\mathbf{I})^{-1} \mathbf{b} = \mathbf{c} \left[\underbrace{(s - s_0)\mathbf{I}}_{\hat{\mathbf{s}}} - \underbrace{(\mathbf{A} - s_0\mathbf{I})}_{\hat{\mathbf{A}}} \right]^{-1} \mathbf{b} \\ &= \mathbf{c} \left[\hat{\mathbf{A}} \left(\hat{\mathbf{s}}\hat{\mathbf{A}}^{-1} - \mathbf{I} \right) \right]^{-1} \mathbf{b} = -\mathbf{c} \left(\mathbf{I} - \hat{\mathbf{s}}\hat{\mathbf{A}}^{-1} \right)^{-1} \hat{\mathbf{A}}^{-1} \mathbf{b}. \end{aligned}$$

Wir verwenden die sogenannte Neumann-Reihe: $(\mathbf{I} - \mathbf{T})^{-1} = \sum_{i=0}^{\infty} \mathbf{T}^i$. Mit $\mathbf{T} := \hat{\mathbf{s}}\hat{\mathbf{A}}^{-1}$ folgt:

$$G(s) = -\mathbf{c} \sum_{i=0}^{\infty} \left(\hat{\mathbf{s}}\hat{\mathbf{A}}^{-1} \right)^i \hat{\mathbf{A}}^{-1} \mathbf{b} = - \sum_{i=0}^{\infty} \mathbf{c} (\mathbf{A} - s_0\mathbf{I})^{-(i+1)} \mathbf{b} (s - s_0)^i.$$

Der Vergleich mit (6.2) liefert das Ergebnis: $m_i = \mathbf{c} (\mathbf{A} - s_0\mathbf{I})^{-(i+1)} \mathbf{b}$. □

Das Ziel des *Moment Matching* ist die Übereinstimmung der ersten q Momente von originalem und reduziertem System:

$$\begin{aligned} G(s) &= -m_0 - m_1(s - s_0) - \dots - m_{q-1}(s - s_0)^{q-1} - m_q(s - s_0)^q - \dots \\ G_r(s) &= -m_{r,0} - m_{r,1}(s - s_0) - \dots - m_{r,q-1}(s - s_0)^{q-1} - m_{r,q}(s - s_0)^q - \dots \\ \text{mit} \quad m_i &= m_{r,i}, \quad i = 0, 1, \dots, q-1 \\ m_j &\neq m_{r,j}, \quad j = q, q+1, \dots \end{aligned}$$

Da wir Systeme in Zustandsraumdarstellung betrachten, suchen wir also ein reduziertes System, so dass gilt:

$$\mathbf{c} (\mathbf{A} - s_0\mathbf{I})^{-(i+1)} \mathbf{b} \stackrel{!}{=} \mathbf{c}_r (\mathbf{A}_r - s_0\mathbf{I})^{-(i+1)} \mathbf{b}_r, \quad i = 0, 1, \dots, (q-1) \quad (6.5)$$

Es wird sich zeigen, dass wir dies erreichen, wenn wir als Projektionsmatrizen \mathbf{V} und \mathbf{W} sogenannte Krylow-Unterräume verwenden. sssMORMoments

6.1.2 Krylow-Unterräume

Definition 23. Der **Krylow-Unterraum** \mathcal{K}_q einer Matrix $\mathbf{M} \in \mathbb{R}^{n \times n}$ und eines Vektors $\mathbf{v} \in \mathbb{R}^n$ ist allgemein definiert als:

$$\mathcal{K}_q(\mathbf{M}, \mathbf{v}) = \text{span} \{ \mathbf{v}, \mathbf{M}\mathbf{v}, \mathbf{M}^2\mathbf{v}, \dots, \mathbf{M}^{q-1}\mathbf{v} \}. \quad (6.6)$$

Ein Krylow-Raum bezeichnet also einen q -dimensionalen Unterraum $\mathcal{K}_q \subseteq \mathbb{R}^n$, zu dem (unendlich) viele Basen $\mathbf{V} \in \mathbb{R}^{n \times q}$ existieren. Krylow-Unterräume wurden eingeführt um Eigenwertprobleme und lineare Gleichungssysteme zu lösen. Für die Modellreduktion benötigen wir zwei bestimmte Krylow-Unterräume, welche im Folgenden definiert werden.

Definition 24. $\mathcal{K}_q((\mathbf{A} - s_0\mathbf{I})^{-1}, (\mathbf{A} - s_0\mathbf{I})^{-1}\mathbf{b})$ heißt **Eingangs-Krylow-Unterraum** und $\mathcal{K}_q((\mathbf{A} - s_0\mathbf{I})^{-T}, (\mathbf{A} - s_0\mathbf{I})^{-T}\mathbf{c}^T)$ heißt **Ausgangs-Krylow-Unterraum** um den Entwicklungspunkt s_0 .

6.1.3 Moment Matching

Mit Hilfe der bisherigen Definitionen können wir bereits die (theoretische) Lösung zum Moment-Matching-Problem formulieren. Die praktikable Umsetzung folgt in Kapitel 6.2.

Satz 7 (Moment Matching 1). *Bilden die Spalten von \mathbf{V} eine q -dimensionale Basis des Eingangs-Krylow-Unterraums um s_0 und ist $\mathbf{W} \in \mathbb{R}^{n \times q}$ eine beliebige Matrix, aber so dass $\det(\mathbf{A}_r - s_0\mathbf{I}) \neq 0$, dann sind die ersten q Momente von originalem und reduziertem System um s_0 identisch.*

Beweis. Das erste Moment des reduzierten Systems lautet:

$$m_{r,0} = \mathbf{c}_r (\mathbf{A}_r - s_0\mathbf{I})^{-1} \mathbf{b}_r \quad (6.7)$$

$$= \mathbf{c}\mathbf{V} \left((\mathbf{W}^T\mathbf{V})^{-1} \mathbf{W}^T\mathbf{A}\mathbf{V} - s_0\mathbf{I} \right)^{-1} (\mathbf{W}^T\mathbf{V})^{-1} \mathbf{W}^T\mathbf{b} \quad (6.8)$$

$$= \mathbf{c}\mathbf{V} \left((\mathbf{W}^T\mathbf{V})^{-1} (\mathbf{W}^T\mathbf{A}\mathbf{V} - s_0\mathbf{W}^T\mathbf{V}) \right)^{-1} (\mathbf{W}^T\mathbf{V})^{-1} \mathbf{W}^T\mathbf{b} \quad (6.9)$$

$$= \mathbf{c}\mathbf{V} (\mathbf{W}^T\mathbf{A}\mathbf{V} - s_0\mathbf{W}^T\mathbf{V})^{-1} \mathbf{W}^T\mathbf{b} \quad (6.10)$$

$$= \mathbf{c}\mathbf{V} (\mathbf{W}^T\mathbf{A}\mathbf{V} - s_0\mathbf{W}^T\mathbf{V})^{-1} \mathbf{W}^T \underbrace{(\mathbf{A} - s_0\mathbf{I})(\mathbf{A} - s_0\mathbf{I})^{-1}}_{=\mathbf{I}} \mathbf{b} \quad (6.11)$$

Da $(\mathbf{A} - s_0\mathbf{I})^{-1} \mathbf{b}$ die erste Richtung des Krylow-Unterraumes $\mathcal{K}_q((\mathbf{A} - s_0\mathbf{I})^{-1}, (\mathbf{A} - s_0\mathbf{I})^{-1}\mathbf{b})$ ist und \mathbf{V} eine Basis dieses Krylow-Unterraums darstellt, gilt:

$$\exists \mathbf{r}_0 \in \mathbb{R}^q : (\mathbf{A} - s_0\mathbf{I})^{-1} \mathbf{b} = \mathbf{V}\mathbf{r}_0. \quad (6.12)$$

Daraus folgt:

$$m_{r,0} = \mathbf{cV} (\mathbf{W}^T \mathbf{A} \mathbf{V} - s_0 \mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T (\mathbf{A} - s_0 \mathbf{I}) \mathbf{V} \mathbf{r}_0 \quad (6.13)$$

$$= \mathbf{cV} (\mathbf{W}^T \mathbf{A} \mathbf{V} - s_0 \mathbf{W}^T \mathbf{V})^{-1} (\mathbf{W}^T \mathbf{A} \mathbf{V} - s_0 \mathbf{W}^T \mathbf{V}) \mathbf{r}_0 \quad (6.14)$$

$$= \mathbf{cV} \mathbf{r}_0 \stackrel{(6.12)}{=} \mathbf{c} (\mathbf{A} - s_0 \mathbf{I})^{-1} \mathbf{b} = m_0. \quad (6.15)$$

Der Beweis für die restlichen $q - 1$ Momente kann analog und mithilfe der vollständigen Induktion durchgeführt werden. \square

Satz 8 (Moment Matching 2). *Bilden die Spalten von \mathbf{W} eine q -dimensionale Basis des Ausgangs-Krylow-Unterraums um s_0 und ist \mathbf{V} derart, dass $\det(\mathbf{A}_r - s_0 \mathbf{I}) \neq 0$, dann sind die ersten q Momente von originalem und reduziertem System um s_0 identisch.*

Beweis. Der Beweis folgt aus der Dualität, d. h. es muss bei obigem Beweis \mathbf{A} mit \mathbf{A}^T , \mathbf{b} mit \mathbf{c}^T und \mathbf{V} mit \mathbf{W} substituiert werden und die Bedingung

$$\exists \mathbf{l}_0 \in \mathbb{R}^q : (\mathbf{A} - s_0 \mathbf{I})^{-T} \mathbf{c}^T = \mathbf{W} \mathbf{l}_0. \quad (6.16)$$

verwendet werden. \square

Folgerung 1 (Moment Matching 3). *Bilden die Spalten von \mathbf{V} eine Basis des Eingangs-Krylow-Unterraums **und** die Spalten von \mathbf{W} eine Basis des Ausgangs-Krylow-Unterraums, dann sind die ersten $2q$ Momente von originalem und reduziertem System um s_0 identisch.*

6.1.4 Markov Parameter¹

Möchte man das transiente Verhalten approximieren, also für $t \rightarrow 0$ und $s \rightarrow \infty$, dann müssen die sogenannten Markov-Parameter zur Übereinstimmung gebracht werden. Sie sind das Analogon zu den Momenten für $s_0 \rightarrow \infty$.

Definition 25. *Die Markov-Parameter M_i eines Systems sind definiert als:*

$$M_i = - \left. \frac{1}{i!} \frac{d^i g(t)}{dt^i} \right|_{t=0} = \mathbf{c} \mathbf{A}^i \mathbf{b} \quad i = 0, 1, \dots \quad (6.17)$$

und man kann zeigen, dass gilt:

$$G(s) = \sum_{i=0}^{\infty} M_i \frac{1}{s^{i+1}} = M_0 \frac{1}{s} + M_1 \frac{1}{s^2} + M_2 \frac{1}{s^3} + \dots \quad (6.18)$$

Um Markov-Parameter im reduzierten System zur Übereinstimmung zu bringen, benötigt man das folgende Theorem.

Satz 9 (Moment Matching 4). *Bilden die Spalten von \mathbf{V} eine Basis des Krylow-Unterraums $\mathcal{K}_q(\mathbf{A}, \mathbf{b})$ und die Spalten von \mathbf{W} eine Basis des Krylow-Unterraums $\mathcal{K}_q(\mathbf{A}^T, \mathbf{c}^T)$, dann stimmen die ersten $2q$ Markov-Parameter von originalem und reduziertem System überein.*

¹Das Kapitel über Markov Parameter wird nur der Vollständigkeit halber und ohne Beweise angegeben.

6.1.5 Mehrere Entwicklungspunkte

Es ist möglich Moment Matching um mehr als einen Entwicklungspunkt zu garantieren (inkl. $s_0 \rightarrow \infty$). Hierzu bildet man die Projektionsmatrizen \mathbf{V} und \mathbf{W} derart, dass sie die Vereinigung geeigneter Krylow-Unterräume aufspannen.

Beispiel: $\mathbf{V} = [\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3]$, wobei:

$$\begin{aligned} \text{Bild}(\mathbf{V}_1) &= \mathcal{K}_{q_1} \left((\mathbf{A} - s_1 \mathbf{I})^{-1}, (\mathbf{A} - s_1 \mathbf{I})^{-1} \mathbf{b} \right), \\ \text{Bild}(\mathbf{V}_2) &= \mathcal{K}_{q_2} \left((\mathbf{A} - s_2 \mathbf{I})^{-1}, (\mathbf{A} - s_2 \mathbf{I})^{-1} \mathbf{b} \right), \\ \text{Bild}(\mathbf{V}_3) &= \mathcal{K}_{q_3} (\mathbf{A}, \mathbf{b}). \end{aligned}$$

Dann werden q_1 Momente um s_1 zur Übereinstimmung gebracht, q_2 Momente um s_2 und zusätzlich q_3 Markov-Parameter.

6.1.6 Zusammenfassung

Im Allgemeinen ist $2q$ die maximal mögliche Zahl zu matchender Momente, da eine Übertragungsfunktion der Ordnung q nur $2q$ Freiheitsgrade besitzt (in Partialbruchzerlegung: q Residuen plus q Pole im Nenner). Nach der Anzahl der Momente die übereinstimmen, unterscheidet man zwischen *einseitiger* und *zweiseitiger* Methode:

Einseitige Reduktion		Zweiseitige Reduktion
\mathbf{V} Eingangs-Krylow-Raum und $\mathbf{W} := \mathbf{V}$	\mathbf{W} Ausgangs-Krylow-Raum und $\mathbf{V} := \mathbf{W}$	\mathbf{V} Eingangs-Krylow-Raum \mathbf{W} Ausgangs-Krylow-Raum
q Momente matchen		$2q$ Momente matchen
Zustandstransformation im originalen System ändert die reduzierte Dynamik!		$2q$ Freiheitsgrade fest \Rightarrow reduzierte Dynamik ist eindeutig!

Hinweis: Es existieren aber auch andere Methoden, z. B. dass \mathbf{V} einen Eingangs-Krylowraum aufspannt und $\mathbf{W} := \mathbf{W}_o \mathbf{V}$ gewählt wird. Diese Wahl erhält die Stabilität im reduzierten System, geht aber über den Rahmen der Vorlesung hinaus.

 sssMOR Funktion(en): rk

6.2 Berechnung der Matrizen \mathbf{V} und \mathbf{W}

Um Moment Matching zu garantieren, muss eine Matrix \mathbf{V} oder \mathbf{W} berechnet werden, deren Spalten den gewünschten Krylow-Raum aufspannen. Auf die explizite Berechnung der

Krylow-Richtungen $(\mathbf{A} - s_0\mathbf{I})^{-i} \mathbf{b}$ muss dabei verzichtet werden, da dies zu numerischen Problemen führen würde. Betrachten wir dies am Beispiel $s_0 = 0$, d. h. $\mathbf{V} = [\mathbf{A}^{-1}\mathbf{b}, \dots, \mathbf{A}^{-q}\mathbf{b}]$. Die einzelnen Krylow-Richtungen könnten theoretisch folgendermaßen bestimmt werden:

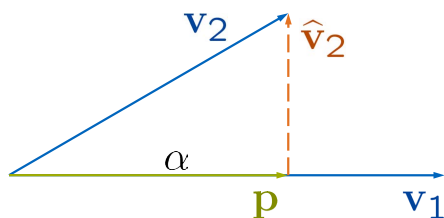
$$\mathbf{v}_1 = \mathbf{A}^{-1}\mathbf{b} \quad (6.19)$$

$$\mathbf{v}_i = \mathbf{A}^{-1}\mathbf{v}_{i-1}, \quad \text{für } i = 2, \dots, q \quad (6.20)$$

Das Problem liegt in der sukzessiven Multiplikation mit ein und derselben Matrix. Durch eine Multiplikation werden die Anteile eines Eigenvektors genau mit seinem Eigenwert gestreckt. Das bedeutet, dass Anteile an Eigenvektoren zu kleinen Eigenwerten immer kürzer werden im Vergleich zu Anteilen an Eigenvektoren zu großen Eigenwerten. Je stärker sich die Eigenwerte von \mathbf{A} betragsmäßig unterscheiden, desto schneller wird dieser Unterschied hochpotenziert. Die Richtungen \mathbf{v}_i werden dadurch parallelisiert, so dass bereits nach wenigen Iterationen numerisch keine q unabhängigen Spalten mehr in \mathbf{V} vorliegen! Numerisch am besten konditioniert sind orthogonale Matrizen, d. h. $\mathbf{V}^T\mathbf{V} = \mathbf{I}$. Das Ziel dieses Kapitels ist deshalb eine orthonormale Basis \mathbf{V} eines gewünschten Krylow-Raumes zu finden. Hierzu benötigen wir zunächst das sogenannte Gram-Schmidt Verfahren.

6.2.1 Gram-Schmidt Verfahren

Beispielhaft seien zwei Vektoren \mathbf{v}_1 und \mathbf{v}_2 gegeben, welche einen zweidimensionalen Unterraum aufspannen. Es ist nun eine orthogonale Basis $\{\mathbf{v}_1, \hat{\mathbf{v}}_2\}$ des selben Unterraumes gesucht. Das folgende Bild verdeutlicht den Zusammenhang,



wobei $\alpha \in \mathbb{R}$ eine geeignete Skalierung von \mathbf{v}_1 darstellt:

$$\mathbf{p} = \alpha\mathbf{v}_1 \quad (6.21)$$

$$\hat{\mathbf{v}}_2 = \mathbf{v}_2 - \mathbf{p} \stackrel{(6.21)}{=} \mathbf{v}_2 - \alpha\mathbf{v}_1 \quad (6.22)$$

Das Ziel ist $\hat{\mathbf{v}}_2 \perp \mathbf{v}_1$, d. h. $\hat{\mathbf{v}}_2^T\mathbf{v}_1 = 0$:

$$(\mathbf{v}_2 - \alpha\mathbf{v}_1)^T\mathbf{v}_1 = 0 \quad (6.23)$$

$$\Leftrightarrow \alpha = \frac{\mathbf{v}_1^T\mathbf{v}_2}{\mathbf{v}_1^T\mathbf{v}_1} \quad (6.24)$$

$$\stackrel{(6.22)}{\implies} \hat{\mathbf{v}}_2 = \mathbf{v}_2 - \frac{\mathbf{v}_1^T\mathbf{v}_2}{\mathbf{v}_1^T\mathbf{v}_1}\mathbf{v}_1 \quad (6.25)$$

Die Vektoren $\{\mathbf{v}_1, \widehat{\mathbf{v}}_2\}$ spannen den gleichen Raum auf wie $\{\mathbf{v}_1, \mathbf{v}_2\}$, sind aber orthogonal zueinander. Um nun eine *orthonormale* Basis zu erhalten wählen wir:

$$\mathbf{v}_1^* := \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}, \quad \mathbf{v}_2^* := \frac{\widehat{\mathbf{v}}_2}{\|\widehat{\mathbf{v}}_2\|}. \quad (6.26)$$

Die resultierende Projektionmatrix $\mathbf{V} = [\mathbf{v}_1^*, \mathbf{v}_2^*]$ ist dann orthogonal, d. h. $\mathbf{V}^T \mathbf{V} = \mathbf{I}$. Das ganze Vorgehen wird im folgenden Algorithmus für beliebige Unterräume zusammengefasst.

Algorithmus 2 Gram-Schmidt Verfahren

Input: $\mathbf{v}_1, \dots, \mathbf{v}_q$

Output: orthonormale Basis $\mathbf{v}_1^*, \dots, \mathbf{v}_q^*$

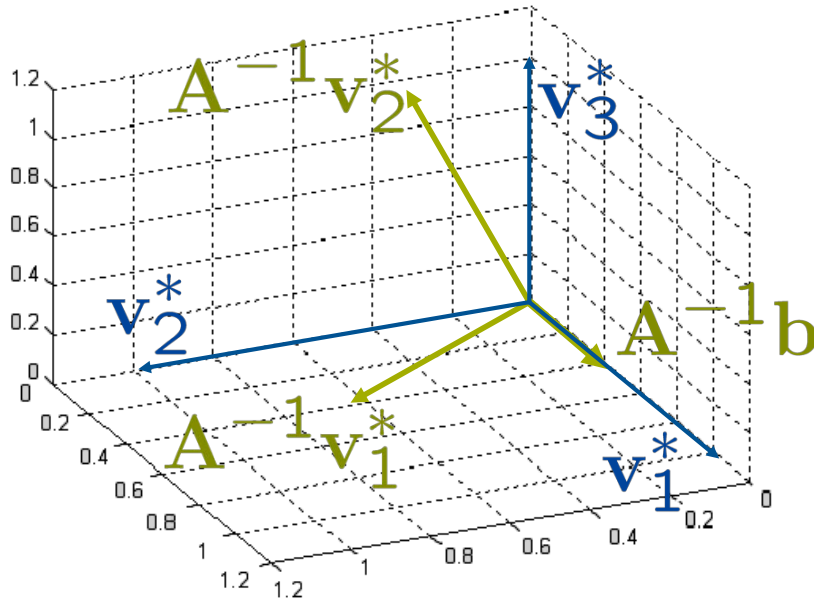
```

1:  $\mathbf{v}_1^* = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}$  // Normierung des ersten Vektors
2: for  $i = 2$  to  $q$  do
3:   for  $j = 1$  to  $(i - 1)$  do
4:      $\mathbf{v}_i \leftarrow \mathbf{v}_i - \frac{\mathbf{v}_j^T \mathbf{v}_i}{\mathbf{v}_j^T \mathbf{v}_j} \mathbf{v}_j$  // Orthogonalisierung zu allen vorherigen Vektoren
5:   end for
6:    $\mathbf{v}_i^* = \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|}$  // Normierung des resultierenden Vektors
7: end for

```

6.2.2 Arnoldi Algorithmus

Das Gram-Schmidt Verfahren berechnet eine orthonormale Basis für einen gegebenen Satz von Vektoren. Um die numerische Konditionierung der Iteration (6.19), (6.20) zu verbessern, muss das Gram-Schmidt Verfahren in jeder Iteration angewendet werden. Im Beispiel (6.19), (6.20) bedeutet das, dass statt $\mathbf{v}_3 := \mathbf{A}^{-1} \mathbf{v}_2$ die orthogonale Richtung \mathbf{v}_2^* verwendet werden muss: $\mathbf{v}_3 := \mathbf{A}^{-1} \mathbf{v}_2^*$. Anschließend muss \mathbf{v}_3 mit Hilfe von Gram-Schmidt zu \mathbf{v}_1^* und \mathbf{v}_2^* orthonormiert werden, wodurch man \mathbf{v}_3^* erhält. Damit muss dann die nächste Richtung bestimmt werden, $\mathbf{v}_4 = \mathbf{A}^{-1} \mathbf{v}_3^*$; man startet wieder das Gram-Schmidt Verfahren; und so weiter... Das folgende Bild zeigt eine graphische Illustration, wie die Richtungen aussehen könnten.



Die resultierende orthonormale Basis $\mathbf{V} := [\mathbf{v}_1^*, \mathbf{v}_2^*, \dots]$ spannt den gewünschten Krylow-Raum auf (Beweis: s. Übung). Die Kombination des Gram-Schmidt Verfahrens mit der iterativen Berechnung der nächsten Krylow-Richtungen wird im Arnoldi Algorithmus zusammengefasst. Bevor der Algorithmus angegeben wird, müssen wir aber noch ein weiteres Problem aus numerischer Sicht behandeln: zur Berechnung einer Krylow-Richtung $(\mathbf{A} - s_0\mathbf{I})^{-1}\mathbf{v}_i$ wird (theoretisch) die Inverse der Matrix $\mathbf{A} - s_0\mathbf{I}$ benötigt. Diese ist aber numerisch teuer ($\mathcal{O}(n^3)$), i.A. vollbesetzt und entsprechend speicherintensiv ($\mathcal{O}(n^2)$), schlecht konditioniert und sollte aus all diesen Gründen unbedingt vermieden werden! Um die explizite Inversion zu vermeiden kann das Problem äquivalent in ein lineares Gleichungssystem umformuliert werden:

$$\mathbf{v}_{i+1} = (\mathbf{A} - s_0\mathbf{I})^{-1}\mathbf{v}_i \iff (\mathbf{A} - s_0\mathbf{I})\mathbf{v}_{i+1} = \mathbf{v}_i \quad (6.27)$$

Dieses Gleichungssystem kann z.B. durch das Gauß-Jordan Verfahren effizienter gelöst werden („\“-Operator in Matlab). Da ein solches Gleichungssystem aber mehrfach für eine Krylow-Basis (nur mit unterschiedlichen rechten Seiten \mathbf{v}_i) gelöst werden muss, ist es zusätzlich sinnvoll eine sogenannte LU -Zerlegung durchzuführen. Dabei wird eine beliebige Matrix in das Produkt einer oberen und unteren Dreiecksmatrix zerlegt (Befehl `lu` in Matlab):

$$\mathbf{A} = \mathbf{LU} = \underbrace{\begin{pmatrix} \diagdown \\ \diagdown \\ \diagdown \end{pmatrix}}_{\mathbf{L}} \underbrace{\begin{pmatrix} \diagdown \\ \diagdown \\ \diagdown \end{pmatrix}}_{\mathbf{U}}$$

Gleichungssysteme mit Dreiecksmatrizen können einfach durch iteratives Einsetzen gelöst werden. Durch die Dreiecksform von \mathbf{L} und \mathbf{U} wird das Gleichungssystem (6.27) effektiv in zwei Schritten gelöst, $(\mathbf{A} - s_0\mathbf{I})^{-1} = (\mathbf{LU})^{-1} = \mathbf{U}^{-1}\mathbf{L}^{-1}$:

1. Löse $\mathbf{L}\mathbf{y} = \mathbf{v}_i$ beginnend mit dem ersten Eintrag von \mathbf{y} .
2. Löse $\mathbf{U}\mathbf{v}_{i+1} = \mathbf{y}$ beginnend mit dem letzten Eintrag von \mathbf{v}_{i+1} .

Diese zwei Schritte lassen sich in MATLAB Notation mithilfe des „\“-Operators besonders kurz fassen:

$$\mathbf{L}\mathbf{U}\mathbf{v}_{i+1} = \mathbf{v}_i \iff \mathbf{v}_{i+1} = \mathbf{U} \setminus (\mathbf{L} \setminus \mathbf{v}_i)$$

Die LU-Zerlegung ist deshalb sinnvoll, weil sie nur einmal am Anfang ausgeführt werden muss. Mit den gespeicherten Matrizen \mathbf{L} und \mathbf{U} können die q Gleichungssysteme für q Krylow-Richtungen sofort gelöst werden! Der numerisch aufwändigste Teil bei der Reduktion mit Krylow-Unterräumen ist deshalb die LU-Zerlegung.

Algorithmus 3 zeigt den *Arnoldi*-Algorithmus zur effizienten Berechnung einer orthogonalen Basis eines Krylow-Unterraums. Er fasst die LU-Zerlegung, das Gram-Schmidt Verfahren und die iterative Berechnung der Krylow-Richtungen zusammen.

Algorithmus 3 Arnoldi Algorithmus


Input: Matrizen \mathbf{A} , \mathbf{b} , Entwicklungspunkt s_0 , Ordnung q

Output: orthonormale Basis des Krylow-Unterraumes $\mathbf{v}_1^*, \dots, \mathbf{v}_q^*$

```

1:  $\mathbf{L}, \mathbf{U} \leftarrow \text{lu}(\mathbf{A} - s_0\mathbf{I})$  // LU-Zerlegung
2:  $\mathbf{v}_1 = \mathbf{U} \setminus (\mathbf{L} \setminus \mathbf{b})$  // erste Krylow-Richtung
3:  $\mathbf{v}_1^* = \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}$  // Normierung des ersten Vektors
4: for  $i = 2$  to  $q$  do
5:    $\mathbf{v}_i = \mathbf{U} \setminus (\mathbf{L} \setminus \mathbf{v}_{i-1}^*)$  // Berechnung der nächsten Krylow-Richtung
6:    $\mathbf{v}_i^* = \text{gram-schmidt}(\mathbf{v}_1^*, \dots, \mathbf{v}_{i-1}^*, \mathbf{v}_i)$  // Orthonormierung zu vorherigen Richtungen
7: end for

```

 sssMOR Funktion(en): `arnoldi`, `rk`

6.3 \mathcal{H}_2 -optimale Reduktion

Wir sind nun in der Lage ein dynamisches System lokal numerisch effizient zu approximieren. Wenn man aber im gesamten Frequenzbereich eine möglichst gute Approximation wünscht, stellt sich die Frage, wie die Entwicklungspunkte zu wählen sind. Eine Möglichkeit ist der im Jahr 2006 erstmals veröffentlichte *Iterative Rational Krylow Algorithm* (IRKA)². Hintergrund von IRKA ist, dass das reduzierte Modell mit minimalem Fehler in der \mathcal{H}_2 -Norm folgendes erfüllen muss: die ersten beiden Momente um die an der Imaginärachse gespiegelten,

²S. Gugercin, A. C. Antoulas and Christopher A. Beattie: *A rational Krylow Iteration for Optimal \mathcal{H}_2 Model Reduction*. Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems, pp. 1665-1667, July 2006.

reduzierten Eigenwerte müssen übereinstimmen. Diesen Zusammenhang haben Meier und Luenberger bereits 1965 erkannt. Wüsste man die reduzierten Eigenwerte im Voraus, könnte man die Entwicklungspunkte entsprechend wählen. Da dies aber nicht der Fall ist, wird mit IRKA ein iterativer Algorithmus vorgeschlagen, der gegen ein solches, lokales \mathcal{H}_2 -Optimum konvergiert. Ohne die Hintergründe genauer zu beleuchten, soll hier nur der Algorithmus vorgestellt werden:

Algorithmus 4 IRKA

Input: System \mathbf{A} , \mathbf{b} , \mathbf{c} , Startwerte $s_{0,i}$ (z. B. $s_{0,i} = 0$, $i = 1, \dots, q$), Toleranz ε

Output: Reduziertes System, lokal \mathcal{H}_2 -optimal

1: **wiederhole:**

2: $\mathbf{V} \leftarrow$ Eingangskrylowraum um die Entwicklungspunkte $s_{0,i}$

3: $\mathbf{W} \leftarrow$ Ausgangskrylowraum um die Entwicklungspunkte $s_{0,i}$

4: $\mathbf{A}_r = (\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T \mathbf{A} \mathbf{V}$, $\mathbf{b}_r = (\mathbf{W}^T \mathbf{V})^{-1} \mathbf{W}^T \mathbf{b}$, $\mathbf{c}_r = \mathbf{c} \mathbf{V}$ // System projizieren


5: $s_{0,i} \leftarrow -\bar{\lambda}_i(\mathbf{A}_r)$ // Neue Entwicklungspunkte aus Spiegelung

6: **solange bis:** $|s_{0,i} + \bar{\lambda}_i(\mathbf{A}_r)| < \varepsilon$, $\forall i = 1, \dots, q$

Eine Stärke von IRKA ist die einfache Implementierung: Man muss nur die reduzierten Eigenwerte berechnen; die negativen Eigenwerte werden dann in der nächsten Iteration als Entwicklungspunkte verwendet! Der Nachteil dabei ist, dass in jeder Iteration ein neuer Krylowraum berechnet werden muss, was zu einem erhöhten numerischen Aufwand führt.

Bis heute existiert allerdings noch kein allgemeingültiger Beweis, dass IRKA konvergiert. In der Praxis zeigt sich aber, dass IRKA in den meisten Fällen tatsächlich konvergiert. In einer jüngst erschienen Publikation wird ohne weitere Erklärung behauptet, dass IRKA nur in lokale Minima konvergieren kann, da lokale Maxima abstoßend sind. All diese Eigenschaften machen – trotz der fehlenden Beweise – IRKA heute zu einer der wichtigsten Methoden der Modellreduktion!

Allerdings muss selbst nach Konvergenz von IRKA sichergestellt werden, dass das reduzierte Modell stabil ist. Denn wie bei den meisten Krylow Verfahren ist auch hier die Stabilitätserhaltung nicht garantiert! Insbesondere sei angemerkt, dass für instabile Systeme die \mathcal{H}_2 -Norm nicht definiert ist und somit bei instabilen reduzierten Modellen die vorangegangenen Überlegungen ihre Gültigkeit verlieren.

 sssMOR Funktion(en): `irka`, `cirka`

6.4 Eigenschaften der Krylow-Unterraummethode

- + Numerisch effizient, da keine „teuren“ Berechnungen notwendig: abgesehen von den LU-Zerlegungen sind nur Matrix-Vektor-Multiplikationen involviert.

- + Die Momente selbst werden nie explizit berechnet; man spricht von implizitem Moment Matching.
- Die Stabilität des reduzierten Systems ist nicht garantiert, d. h. selbst wenn das originale System stabil ist, muss das reduzierte System nicht zwangsweise stabil sein.
- Es existieren keine allgemeingültigen, effizient berechenbaren Fehlerschranken der Approximationsgüte; nur für Spezialfälle können diese bisher angegeben werden.
- Das Ergebnis einer einseitigen Reduktion ($\mathbf{W} = \mathbf{V}$) ist nicht eindeutig und abhängig von der originalen Zustandsraumdarstellung, d. h. eine Zustandstransformation $\mathbf{z} = \mathbf{T} \mathbf{x}$ im Originalsystem ändert das reduzierte System!
- Das Problem der Wahl und Anzahl der Entwicklungspunkte, um eine gute Approximation zu erreichen, ist noch nicht vollständig gelöst. Wie gesagt:

*Der Vorteil von Krylow-Unterraummethoden sind die vielen Freiheiten.
Der Nachteil von Krylow-Unterraummethoden sind die vielen Freiheiten.*

6.5 Vergleich Krylow und TBR

Die Eigenschaften der Krylow-Unterraum Methoden und des Balancierten Abschneidens werden in der folgenden Tabelle nochmals gegenüber gestellt:

	TBR	Krylow
Vorteile	+ Automatisierbar + Stabilität bleibt erhalten + a priori berechenbare Fehlerschranke	+ Numerisch effiziente Berechnung + für sehr große Systeme geeignet, z. B. Ordnung 10^6
Nachteile	- Rechenintensiv - Hoher Speicherbedarf - nur für kleine bis mittelgroße Systeme geeignet, ca. bis Ordnung 5000	- Stabilität bleibt i.A. nicht erhalten - Keine Fehlerschranke - Viele Einstellparameter: z. B. Ordnung, Entwicklungspunkt(e), ...

Kapitel 7

Weiterführende Themen

Folgende Themen dienen als Ergänzung und Ausblick zu den bislang beschriebenen Inhalten und werden je nach Zeit und Interesse auch im Rahmen der Vorlesung behandelt. Für die Klausur sind daher nur Inhalte aus diesem Abschnitt relevant, welche auch in Vorlesung und/oder Übung behandelt wurden.

7.1 Systeme in impliziter Zustandsraumdarstellung

Die hier vorgestellte Theorie kann auch auf implizite Zustandsraumdarstellungen der Form:

$$\begin{aligned}\mathbf{E}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t), \\ y(t) &= \mathbf{c}\mathbf{x}(t),\end{aligned}\tag{7.1}$$

mit $\det(\mathbf{E}) \neq 0$ erweitert werden. Die direkte Variante ist, die Zustandsgleichung mit \mathbf{E}^{-1} durchzumultiplizieren und dann wie bisher vorzugehen. Dies ist aufgrund der Inversen aber meist nicht sinnvoll! Zum einen ist die Berechnung der Inverse numerisch schlecht konditioniert, zum anderen geht i.A. die Dünnbesetztheit der Matrizen durch die Inversion und anschließende Multiplikation verloren. Dies ist in Abbildung 7.1 für das Benchmark Modell `heat-cont` dargestellt.

Tatsächlich können Krylow-Räume und Ljapunow-Gleichungen auch auf den Fall mit \mathbf{E} -Matrix verallgemeinert werden. Der Eingangs-Krylow-Raum z. B. lautet dann

$$\mathcal{K}_q \left((\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E}, (\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{b} \right),\tag{7.2}$$

und die verallgemeinerte Ljapunow-Gleichung zur Berechnung der Steuerbarkeits-Gram'sche:

$$\mathbf{A}\mathbf{W}_c\mathbf{E}^T + \mathbf{E}\mathbf{W}_c\mathbf{A}^T + \mathbf{b}\mathbf{b}^T = \mathbf{0}.\tag{7.3}$$

Dadurch kann die Inverse \mathbf{E}^{-1} vermieden werden und das reduzierte System in impliziter Form lautet:

$$\begin{aligned}\mathbf{W}^T\mathbf{E}\mathbf{V}\dot{\mathbf{x}}_r(t) &= \mathbf{W}^T\mathbf{A}\mathbf{V}\mathbf{x}_r(t) + \mathbf{W}^T\mathbf{b}u(t), \\ y_r(t) &= \mathbf{c}\mathbf{V}\mathbf{x}_r(t) + du(t).\end{aligned}\tag{7.4}$$

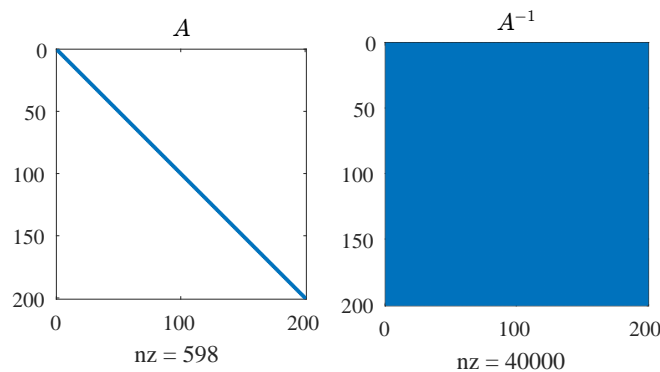


Abbildung 7.1: Sparsity Pattern für die Matrix A und deren Inverse.

Für eine effiziente Implementierung der Verfahren sollte dieser Umgang mit der Matrix \mathbf{E} unbedingt beachtet werden! Für Krylow-Unterraumverfahren ist die Erweiterung für den Fall $\mathbf{E} \neq \mathbf{I}$ ausgiebig behandelt, z.B. in (Grimme, 1997; Beattie und Gugercin, 2014). Beim Balancierten Abschneiden ist die Behandlung der Erweiterung in unterschiedlichen Quellen nicht einheitlich und teilweise fehlerhaft, weswegen sie an dieser Stelle mit unserer Nomenklatur kurz zusammengefasst wird.

7.1.1 Balanciertes Abschneiden für $\mathbf{E} \neq \mathbf{I}$

Gegeben sei ein (i.A. MIMO) Zustandsraummodell in impliziter Form

$$\begin{aligned} \mathbf{E}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t), \end{aligned} \quad (7.5)$$

welcher durch die Bedingung $\det \mathbf{E} \neq 0$ äquivalent ist zur expliziten Zustandsraumdarstellung

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \overbrace{\mathbf{E}^{-1}\mathbf{A}}^{\mathbf{A}_e} \mathbf{x}(t) + \overbrace{\mathbf{E}^{-1}\mathbf{B}}^{\mathbf{B}_e} \mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t). \end{aligned} \quad (7.6)$$

Gram'sche Systemmatrizen Wie bereits bekannt können mithilfe von (7.6) die Gram'sche Matrizen für Steuerbarkeit (\mathbf{W}_c) und Beobachtbarkeit (\mathbf{W}_o) als Lösungen folgender Ljapunow-Gleichungen angegeben werden

$$\mathbf{A}_e \mathbf{W}_c + \mathbf{W}_c \mathbf{A}_e^T + \mathbf{B}_e \mathbf{B}_e^T = 0, \quad (7.7a)$$

$$\mathbf{A}_e^T \mathbf{W}_o + \mathbf{W}_o \mathbf{A}_e + \mathbf{C}^T \mathbf{C} = 0. \quad (7.7b)$$

Einsetzen der Beziehungen aus (7.6) liefert für die Gram'sche Steuerbarkeitsmatrix folgende Gleichung

$$\begin{aligned} & \mathbf{A}_e \mathbf{W}_c + \mathbf{W}_c \mathbf{A}_e^T + \mathbf{B}_e \mathbf{B}_e^T = 0 \\ \Rightarrow & \mathbf{E}^{-1} \mathbf{A} \mathbf{W}_c + \mathbf{W}_c \mathbf{A}^T \mathbf{E}^{-T} + \mathbf{E}^{-1} \mathbf{B} \mathbf{B}^T \mathbf{E}^{-T} = 0 \\ \Rightarrow & \mathbf{A} \mathbf{W}_c \mathbf{E}^T + \mathbf{E} \mathbf{W}_c \mathbf{A}^T + \mathbf{B} \mathbf{B}^T = 0. \end{aligned} \quad (7.8)$$

Gleichungen der Form (7.8) werden auch *generalisierte* Ljapunow-Gleichungen genannt, für die es auch gesonderte Lösungsverfahren gibt. Die Berechnung der Gram'schen Beobachtbarkeitsmatrix mit der impliziten Zustandsraumdarstellung (7.5) erfordert einen zusätzlichen Zwischenschritt:

$$\begin{aligned} & \mathbf{A}_e^T \mathbf{W}_o + \mathbf{W}_o \mathbf{A}_e + \mathbf{C}^T \mathbf{C} = 0 \\ \Rightarrow & \mathbf{A}^T \mathbf{E}^{-T} \mathbf{W}_o + \mathbf{W}_o \mathbf{E}^{-1} \mathbf{A} + \mathbf{C}^T \mathbf{C} = 0 \\ \Rightarrow & \mathbf{A}^T \widetilde{\mathbf{W}}_o \mathbf{E} + \mathbf{E}^T \widetilde{\mathbf{W}}_o \mathbf{A} + \mathbf{C}^T \mathbf{C} = 0. \end{aligned} \quad (7.9)$$

wobei die Hilfsvariable $\widetilde{\mathbf{W}}_o$ als Lösung von (7.9) eingeführt wurde gemäß der Beziehung

$$\mathbf{W}_o = \mathbf{E}^T \widetilde{\mathbf{W}}_o \mathbf{E}. \quad (7.10)$$

Wie bereits diskutiert basiert die "square root" Variante des Balancierten Abschneidens auf die Berechnung der *Cholesky* Faktoren der Gram'schen Systemmatrizen. Da Cholesky Faktoren für die Lösungen der generalisierten Ljapunow-Gleichungen (7.8) und (7.9) direkt berechnet werden können, ist es auch hier sinnvoll, die Cholesky Zerlegung für \mathbf{W}_c und $\widetilde{\mathbf{W}}_o$ vorzugeben:

$$\mathbf{W}_c = \mathbf{S} \mathbf{S}^T, \quad (7.11a)$$

$$\widetilde{\mathbf{W}}_o = \widetilde{\mathbf{R}} \widetilde{\mathbf{R}}^T. \quad (7.11b)$$

Balancierte Darstellung Durch geeignete Zustandstransformation

$$\mathbf{x}_b = \mathbf{T}_b \mathbf{x} \quad (7.12)$$

kann (7.5) in die balancierte Darstellung

$$\begin{aligned} \underbrace{\mathbf{T}_b \mathbf{E} \mathbf{T}_b^{-1}}_{\mathbf{E}_b} \dot{\mathbf{x}}_b(t) &= \underbrace{\mathbf{T}_b \mathbf{A} \mathbf{T}_b^{-1}}_{\mathbf{A}_b} \mathbf{x}_b(t) + \underbrace{\mathbf{T}_b \mathbf{B}}_{\mathbf{B}_b} \mathbf{u}(t), \\ \mathbf{y}(t) &= \underbrace{\mathbf{C} \mathbf{T}_b^{-1}}_{\mathbf{C}_b} \mathbf{x}_b(t), \end{aligned} \quad (7.13)$$

gebracht werden. Analog zum expliziten Fall ergibt sich für die Gram'sche Matrizen in balancierter Darstellung die Transformationsvorschrift

$$\mathbf{W}_{c,b} = \mathbf{T}_b \mathbf{W}_c \mathbf{T}_b^T, \quad (7.14a)$$

$$\mathbf{W}_{o,b} = \mathbf{T}_b^{-T} \mathbf{W}_o \mathbf{T}_b^{-1} = \mathbf{T}_b^{-T} \mathbf{E}^T \widetilde{\mathbf{W}}_o \mathbf{E} \mathbf{T}_b^{-1}. \quad (7.14b)$$

Durch die Forderung $\mathbf{W}_{c,b} = \mathbf{W}_{o,b} = \mathbf{\Sigma}$ mit $\mathbf{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_n)$ und $\sigma_i = \sqrt{\lambda_i(\mathbf{W}_c \mathbf{W}_o)}$ kann der Ausdruck für \mathbf{T}_b hergeleitet werden.

Satz 10. Die Zustandstransformation \mathbf{T}_b auf balancierte Darstellung kann folgendermaßen berechnet werden:

1. Lösung der generalisierten Ljapunow-Gleichungen (7.8) und (7.9) nach den Cholesky-Faktoren $\mathbf{W}_c = \mathbf{S}\mathbf{S}^T$ und $\tilde{\mathbf{W}}_o = \tilde{\mathbf{R}}\tilde{\mathbf{R}}^T$.
2. SVD von $\tilde{\mathbf{R}}^T \mathbf{E} \mathbf{S} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$.

Dann ergeben sich \mathbf{T}_b und \mathbf{T}_b^{-1} zu

$$\mathbf{T}_b := \mathbf{\Sigma}^{-1/2} \mathbf{U}^T \tilde{\mathbf{R}}^T \mathbf{E} \quad (7.15a)$$

$$\mathbf{T}_b^{-1} := \mathbf{S} \mathbf{V} \mathbf{\Sigma}^{-1/2} \quad (7.15b)$$

Beweis.

$$\begin{aligned} \mathbf{W}_{c,b} &= \mathbf{T}_b \mathbf{W}_c \mathbf{T}_b^T \quad \text{mit} \quad \mathbf{W}_c = \mathbf{S}\mathbf{S}^T \\ &= \mathbf{\Sigma}^{-1/2} \mathbf{U}^T \tilde{\mathbf{R}}^T \mathbf{E} (\mathbf{S}\mathbf{S}^T) \mathbf{E}^T \tilde{\mathbf{R}} \mathbf{U} \mathbf{\Sigma}^{-1/2} \\ &= \mathbf{\Sigma}^{-1/2} \underbrace{\mathbf{U}^T \mathbf{U}}_{=\mathbf{I}} \mathbf{\Sigma} \underbrace{\mathbf{V}^T \mathbf{V}}_{=\mathbf{I}} \mathbf{\Sigma} \underbrace{\mathbf{U}^T \mathbf{U}}_{=\mathbf{I}} \mathbf{\Sigma}^{-1/2} \\ &= \mathbf{\Sigma}^{-1/2} \mathbf{\Sigma}^2 \mathbf{\Sigma}^{-1/2} = \mathbf{\Sigma}. \end{aligned}$$

Der Beweis für $\mathbf{W}_{o,b} = \mathbf{\Sigma}$ erfolgt analog mithilfe von (7.14b) und (7.11b). \square

Hinweis 1. Man erkennt, dass Gleichungen (7.15) für den Fall $\mathbf{E} = \mathbf{I}$ mit denen übereinstimmen, die in Abschnitt 5.3 hergeleitet wurden.

Hinweis 2. Es sei angemerkt, dass man in der Literatur manchmal die Vorschrift

$$\mathbf{T}_b := \mathbf{\Sigma}^{-1/2} \mathbf{U}^T \mathbf{R} \quad (7.16)$$

findet. Dabei muss beachtet werden, dass \mathbf{R} aus der Cholesky Zerlegung $\mathbf{W}_o = \mathbf{R}\mathbf{R}^T$ resultieren muss und laut (7.10) daher $\mathbf{R} = \mathbf{E}^T \tilde{\mathbf{R}}$ gilt. Sofern dieser Zusammenhang respektiert wird stimmen Gleichung (7.15a) und (7.16) überein.

Genau so können sich leichte Änderungen in den Vorschriften ergeben, wenn die Cholesky Faktoren als $\mathbf{W}_c = \mathbf{S}^T \mathbf{S}$ etc. definiert werden, oder die SVD von $\mathbf{S}^T \mathbf{E}^T \tilde{\mathbf{R}}$ durchgeführt wird.

Balanciertes Abschneiden Anders als im expliziten Fall $\mathbf{E} = \mathbf{I}$ ergeben sich die Projektionsmatrizen in diesem Fall *nicht* durch das Abschneiden in balancierter Darstellung. Um auf eine stabile, balancierte Darstellung für das reduzierte Modell zu gelangen müssen in diesem Fall die Projektionsmatrizen wie folgt definiert werden

$$\mathbf{V}_{BA} := \mathbf{S}\mathbf{V}_q\boldsymbol{\Sigma}_q^{-1/2}, \quad (7.17a)$$

$$\mathbf{W}_{BA}^T := \boldsymbol{\Sigma}_q^{-1/2}\mathbf{U}_q^T\tilde{\mathbf{R}}^T. \quad (7.17b)$$

Insbesondere enthält (7.17b), im Gegensatz zu (7.15b) keine Matrix \mathbf{E} . Für das reduzierte Modell gilt

$$\mathbf{E}_r = \mathbf{W}_{BA}^T \mathbf{E} \mathbf{V}_{BA} = \boldsymbol{\Sigma}_q^{-1/2} \mathbf{U}_q^T \left(\tilde{\mathbf{R}}^T \mathbf{E} \mathbf{S} \right) \mathbf{V}_q \boldsymbol{\Sigma}_q^{-1/2} = \mathbf{I}_q, \quad (7.18)$$

d.h. \mathbf{V}_{BA} und \mathbf{W}_{BA} sind bi-orthogonal bezüglich \mathbf{E} . Für die reduzierte Gram'sche Steuerbarkeitsmatrix folgt aus

$$\begin{aligned} & \mathbf{A}\mathbf{W}_c\mathbf{E}^T + \mathbf{E}\mathbf{W}_c\mathbf{A}^T + \mathbf{B}\mathbf{B}^T &= 0, \\ \implies & \mathbf{W}_{BA}^T (\mathbf{A}\mathbf{W}_c\mathbf{E}^T + \mathbf{E}\mathbf{W}_c\mathbf{A}^T + \mathbf{B}\mathbf{B}^T) \mathbf{W}_{BA} &= 0, \\ \iff & \mathbf{W}_{BA}^T \mathbf{A} (\mathbf{S}\mathbf{S}^T) \mathbf{E}^T \tilde{\mathbf{R}} \mathbf{U}_q \boldsymbol{\Sigma}_q^{-1/2} + \boldsymbol{\Sigma}_q^{-1/2} \mathbf{U}_q^T \tilde{\mathbf{R}}^T \mathbf{E} (\mathbf{S}\mathbf{S}^T) \mathbf{A}^T \mathbf{W}_{BA} + \mathbf{B}_r \mathbf{B}_r^T &= 0, \\ \iff & \mathbf{W}_{BA}^T \mathbf{A} (\mathbf{S}\mathbf{V}_q \boldsymbol{\Sigma}_q^{-1/2}) \boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_q (\boldsymbol{\Sigma}_q^{-1/2} \mathbf{V}_q^T \mathbf{S}^T) \mathbf{A}^T \mathbf{W}_{BA} + \mathbf{B}_r \mathbf{B}_r^T &= 0, \\ \iff & \mathbf{A}_r \boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_q \mathbf{A}_r^T + \mathbf{B}_r \mathbf{B}_r^T &= 0. \end{aligned} \quad (7.19)$$

Ähnlich gilt für die Gram'sche Beobachtbarkeitsmatrix

$$\begin{aligned} & \mathbf{A}^T \tilde{\mathbf{W}}_o \mathbf{E} + \mathbf{E}^T \tilde{\mathbf{W}}_o \mathbf{A} + \mathbf{C}^T \mathbf{C} &= 0, \\ \implies & \mathbf{V}_{BA}^T \left(\mathbf{A}^T \tilde{\mathbf{W}}_o \mathbf{E} + \mathbf{E}^T \tilde{\mathbf{W}}_o \mathbf{A} + \mathbf{C}^T \mathbf{C} \right) \mathbf{V}_{BA} &= 0, \\ \iff & \mathbf{V}_{BA}^T \mathbf{A}^T \left(\tilde{\mathbf{R}} \tilde{\mathbf{R}}^T \right) \mathbf{E} \mathbf{S} \mathbf{V}_q \boldsymbol{\Sigma}_q^{-1/2} + \boldsymbol{\Sigma}_q^{-1/2} \mathbf{V}_q^T \mathbf{S}^T \mathbf{E}^T \left(\tilde{\mathbf{R}} \tilde{\mathbf{R}}^T \right) \mathbf{A} \mathbf{V}_{BA} + \mathbf{C}_r^T \mathbf{C}_r &= 0, \\ \iff & \mathbf{V}_{BA}^T \mathbf{A}^T \left(\tilde{\mathbf{R}} \mathbf{U}_q \boldsymbol{\Sigma}_q^{-1/2} \right) \boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_q \left(\boldsymbol{\Sigma}_q^{-1/2} \mathbf{U}_q^T \tilde{\mathbf{R}}^T \right) \mathbf{A} \mathbf{V}_{BA} + \mathbf{C}_r^T \mathbf{C}_r &= 0, \\ \iff & \mathbf{A}_r^T \boldsymbol{\Sigma}_q + \boldsymbol{\Sigma}_q \mathbf{A}_r + \mathbf{C}_r^T \mathbf{C}_r &= 0. \end{aligned} \quad (7.20)$$

Somit ist bewiesen, dass durch (7.17) das reduzierte Modell balanciert wird und nur die Zustandsrichtungen enthält, die zu den größten q Hankel Singulärwerte gehören.

Anhang

Literatur

Mathematische Grundlagen

- Bartels, R.H. und G.W. Stewart (1972). “Solution of the matrix equation $AX+XB= C$ ”. In: *Comm. ACM* 15.9, S. 820–826.
- Demmel, J. W. (1997). *Applied numerical linear algebra*. SIAM.
- Golub, G. H. und C. F. Van Loan (1996). *Matrix Computations*. Johns Hopkins University Press, Baltimore.
- Liesen, J. und V. Mehrmann (2015). *Lineare algebra*. Springer.

Grundlagen Systemtheorie und Regelungstechnik

- Föllinger, O. (2013). *Regelungstechnik*. 11. Aufl. VDE Verlag.
- Lunze, J. (2010). *Regelungstechnik 2: Mehrgrößensysteme, Digitale Regelung*. 6. Aufl. Springer-Verlag.

Grundlagen MOR

- Antoulas, A. C. (2005). *Approximation of Large-Scale Dynamical Systems*. SIAM. DOI: 10.1137/1.9780898718713. URL: <http://epubs.siam.org/doi/abs/10.1137/1.9780898718713>.
- Beattie, C. und S. Gugercin (2014). “Model reduction by rational interpolation”. Accepted to appear in *Model Reduction and Approximation for Complex Systems*.
- Föllinger, O. (2013). *Regelungstechnik*. 11. Aufl. VDE Verlag.
- Grimme, E. J. (1997). “Krylov Projection Methods for Model Reduction”. Diss. Dep. of Electrical Eng., Uni. Illinois at Urbana Champaign.

Aktuelle Veröffentlichungen

- Castagnotto, A., C. Beattie und S. Gugercin (2017). “Interpolatory methods for \mathcal{H}_∞ model reduction of multi-input/multi-output systems”. In: *Model Reduction of Parametrized Systems III*. Bd. Modeling, Simulation & Applications (MS&A) Series. (Accepted) Available from <http://https://arxiv.org/pdf/1610.01039.pdf>. Springer.
- Castagnotto, A., M. Cruz Varona u. a. (2017). “sss & sssMOR: Analysis and Reduction of Large-Scale Dynamic Systems in MATLAB”. In: *at-Automatisierungstechnik* 65.2, S. 134–150. DOI: 10.1515/auto-2016-0137.
- Castagnotto, A., H. K. F. Panzer und B. Lohmann (2016). “Fast \mathcal{H}_2 -Optimal Model Order Reduction Exploiting the Local Nature of Krylov-Subspace Methods”. en. In: *European Control Conference 2016*. Aalborg, Denmark, S. 6.
- Gallivan, K. A., A. Vandendorpe und P. Van Dooren (2004). “Model reduction of MIMO systems via tangential interpolation”. In: *SIAM Journal on Matrix Analysis and Applications* 26.2, S. 328–349.
- Gugercin, S., A. C. Antoulas und C. Beattie (2008). “ \mathcal{H}_2 model reduction for large-scale linear dynamical systems”. In: *SIAM Journal on Matrix Analysis and Applications* 30.2, S. 609–638.
- Panzer, H. K. F., S. Jaensch u. a. (2013). “A greedy rational Krylov method for \mathcal{H}_2 -pseudooptimal model order reduction with preservation of stability”. In: *American Control Conference*, S. 5532–5537.
- Penzl, T. (2000). “A cyclic low-rank Smith method for large sparse Lyapunov equations”. In: *SIAM Journal on Scientific Computing* 21.4, S. 1401–1418.
- Simoncini, V. (2007). “A new iterative method for solving large-scale Lyapunov matrix equations”. In: *SIAM Journal on Scientific Computing* 29.3, S. 1268–1288.
- Wolf, T., H. K. F. Panzer und B. Lohmann (2013). “ \mathcal{H}_2 pseudo-optimality in model order reduction by Krylov subspace methods”. In: *European Control Conference*.

Deskriptor Systeme (DAEs)

- Benner, P. und T. Stykel (2015). *Model Order Reduction for Differential-Algebraic Equations: a Survey*. Preprint MPIMD/15-19. Available from <http://www.mpi-magdeburg.mpg.de/preprints/>. Max Planck Institute Magdeburg.
- Gugercin, S., T. Stykel und S. Wyatt (2013). “Model reduction of descriptor systems by interpolatory projection methods”. In: *SIAM J. Sci. Comput.* 35.5, B1010–B1033. DOI: 10.1137/130906635.
- Kunkel, P. und V. Mehrmann (2006). *Differential-algebraic equations*. EMS textbooks in mathematics. European Math. Soc., Zürich. DOI: 10.4171/017.

Sonstiges

- Fernando, K und H Nicholson (1982). “Singular perturbational model reduction in the frequency domain”. In: *IEEE Transactions on Automatic Control* 27.4, S. 969–970.

Panzer, H. K. F., J. Mohring u. a. (2010). “Parametric Model order reduction by matrix interpolation”. In: *at-Automatisierungstechnik* 58.8, S. 475–484. DOI: 10.1524/auto.2010.0863.