Technische Universität München

sss & sssMOR

Analyse und Reduktion dynamischer Systeme sehr hoher Ordnung

A. Castagnotto, M. Cruz Varona, T. Emmert, S. Jaensch, M. Meindl, B. Lohmann, W. Polifke

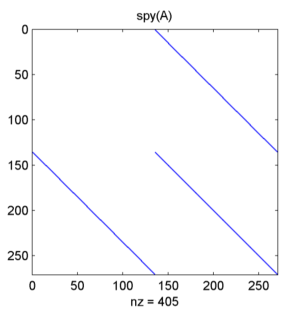
Kurzfassung

Die genaue Modellierung dynamischer Systeme resultiert oft in einer sehr hohen Anzahl an Zustandsvariablen. Die Systemmatrizen können dadurch zu groß werden, um noch Zustandsraum-modelle (**ss** Objekte) in MATLAB[1] definieren zu können. In diesem Beitrag werden Toolboxen vorgestellt, welche durch Einführung sog. **sparse state space Objekten** (**sss**) die Analyse hochdimensionaler Systeme ermöglichen. Mittels **Modellordnungsreduktion** (**sssMOR**) kann die wesentliche Dynamik in Modellen deutlich niedrigerer Ordnung abgebildet werden.

Dünnbesetztheit der Systemmatrizen ausnutzen

Lineare, zeitinvariante Systeme werden oft zum Zwecke des Reglerentwurfs durch Zustandsraummodelle repräsentiert. Wenn die Modellordnung N sehr hoch ist ($N \gg 10^3$), sind die Systemmatrizen i.A. dünnbesetzt (engl.: sparse), sodass die Anzahl der von Null verschiedenen Einträge wesentlich kleiner ist als N^2.

$$\left.\begin{aligned} \boxed{E}\,\dot{x} &= \boxed{A}\,x + \boxed{B}\,u \\ y &= \boxed{C}\,x + \boxed{D}\,u \end{aligned}\right\}\quad x \in \mathbb{R}^N$$



Besetzungsstruktur und Abbildung des ISS Modells[2]

Die **Control System Toolbox** in MATLAB kann leider diese Eigenschaft nicht ausnutzen und speichert alle Matrizen als "**full**" ab. Aus diesem Grund ist die Definition von Zustandsraummodellen mithilfe der Befehle

```
sys = ss(A,B,C,D)  oder  sys = dss(A,B,C,D,E)
```

auf einem Standardrechner nur bis zu einer Größenordnung von $\mathcal{O}\left(10^4\right)$ möglich. Tatsächlich erfordert die Definition einer vollen Einheitsmatrix der Dimension 10^5 80 GB, während die dünnbesetzte Matrix nur noch 2.4 MB benötigt!

Funktionalität

Mit **sss** kann die "sparsity" der Matrizen ausgenutzt werden, mit erheblichen Vorteilen in puncto Speicher- und Rechenaufwand. Dabei wird ein Zustandsraummodell einfach mit

```
sys = sss(A,B,C,D,E)
```

definiert. Weiterhin enthält die **sss** Toolbox viele Analysefunktionen, die aus der Control System Toolbox bekannt sind, und nutzt gezielt die Vorteile dünnbesetzter Matrizen aus.

Funktionen

Manipulation:

```
>> truncate(sys,p,m); connect(sys1,sys2);…
>> sys1-sys2; sys1*sys2; c2d(sysC,Ts);…
```

Analyse im Frequenzbereich:

```
>> freqresp(sys,w); bode(sys); sigma(sys);…
```

Analyse im Zeitbereich:

```
>> impulse(sys); step(sys); lsim(sys,u,Ts);…
```

Zusätzliche Eigenschaften:

```
>> sys.isDae; sys.isSym; sys.isSimo;…
>> norm(sys,2); norm(sys,inf); isstable(sys);…
>> eigs(sys); spy(sys); diag(sys);…
```

Kompatibilität

Alter Code:

```
>> sys = ss(A,B,C,D)
>> myCode(sys)
```

Neuer Code:

```
>> sys = sss(A,B,C,D)
>> myCode(sys)
```

Relevante Dynamik in Modellen niedriger Ordnung erfassen

Auch wenn man die sss Toolbox verwendet, sind Berechnungen mit hochdimensionalen Modellen sehr aufwändig. Deshalb werden reduzierte Modelle viel niedrigerer Ordnung $n \ll N$ gesucht, die das Übertragungsverhalten gut approximieren. Für lineare Systeme lässt sich die Reduktion als Petrov-Galerkin Projektion darstellen:

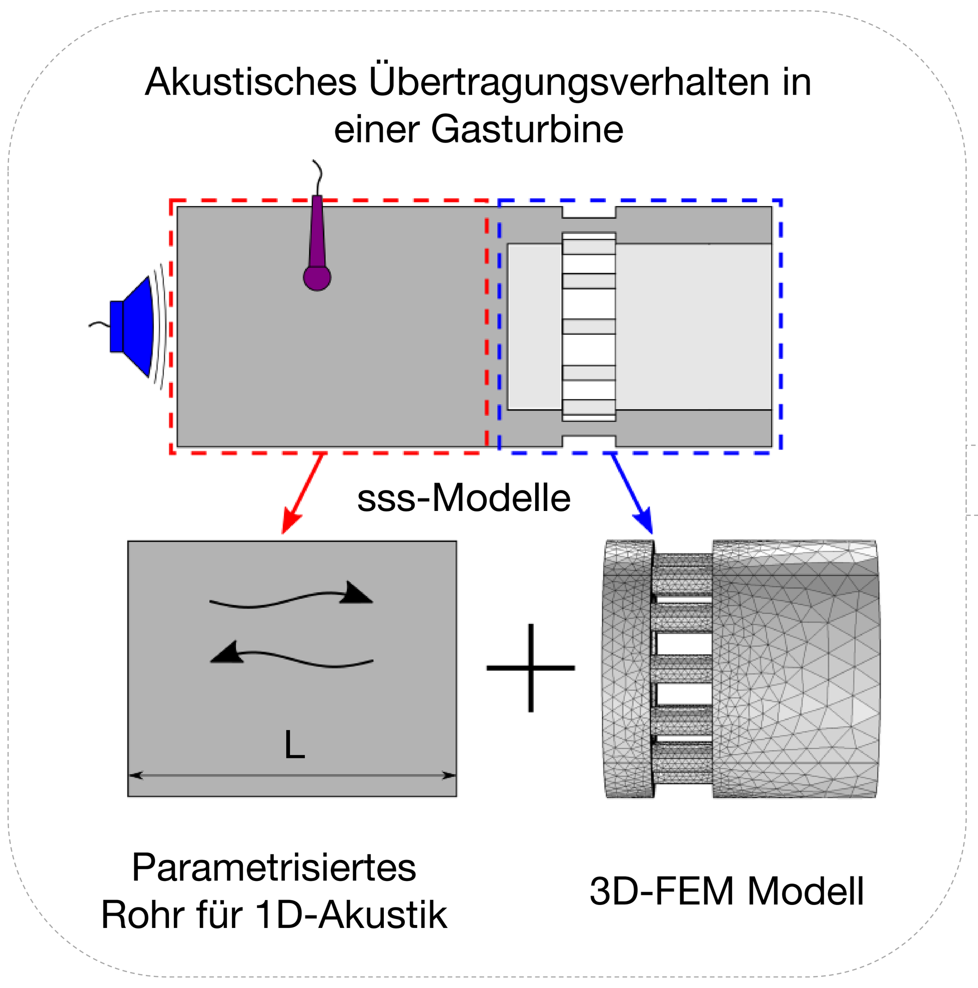
$$\left.\begin{aligned} \overbrace{W^\top E\, V}^{E_r}\,\dot{x}_r &= \overbrace{W^\top A\, V}^{A_r}\,x_r + \overbrace{W^\top B}^{B_r}\,u \\ y_r &= \underbrace{C\, V}_{C_r}\,x_r + \underbrace{D}_{D_r}\,u \end{aligned}\right\}\quad x_r \in \mathbb{R}^n$$

Die Projektionsmatrizen V, W können mit unterschiedlichen Verfahren berechnet werden, je nachdem welche Eigenschaften vom Originalmodell erhalten bleiben sollen. Klassische Methoden umfassen *modale Reduktion*, *Balanciertes Abschneiden und Krylov-Verfahren*, während *IRKA* und CUREd *SPARK* Beispiele für fortgeschrittene Algorithmen sind.

Funktionalität

Modellreduktion in **sssMOR** wird durchgeführt, indem der entsprechenden Funktion ein **sss** Objekt des Originalmodells übergeben wird, eventuell mit weiteren Parametern.

Funktionen	Beschreibung
`modalMor(sys,n)`	Modale Reduktion mit Erhaltung dominanter Eigenwerte
`tbr(sys,n)`	Balanciertes Abschneiden mit Erhaltung dominanter Hankel Singulärwerte
`rk(sys,s0)`	Krylow-Unterraum-Methoden mit Matching einiger Taylor Koeffizienten der Übertragungsfunktion
`irka(sys,s0)`	Iterativer Rationaler Krylow Algorithmus für $\mathcal{H}_2$-optimale Reduktion
`cirka(sys,s0)`	Confined IRKA Algorithmus für schnelle $\mathcal{H}_2$-optimale Reduktion
`spark(sys,s0)`	Stabilitätserhaltender, adaptiver rationaler Krylow Algorithmus
`porkV(…)`	$\mathcal{H}_2$-pseudo-optimaler rationaler Krylow Algorithmus
`cure(sys)`	Kumulative Reduktion mit adaptiver Wahl der reduzierten Ordnung



Anwendung von sss & sssMOR auf parametrische thermoakustische Netzwerkmodelle in taX

taX ist eine MATLAB Toolbox, mit der man thermoakustische Netzwerkmodelle erzeugen kann. Aufgrund der Dünnbesetztheit der Matrizen werden Zustandsraummodelle als sss Objekte definiert.

Aufbau des Gesamtsystems durch Verbinden der beiden Einzelsysteme:

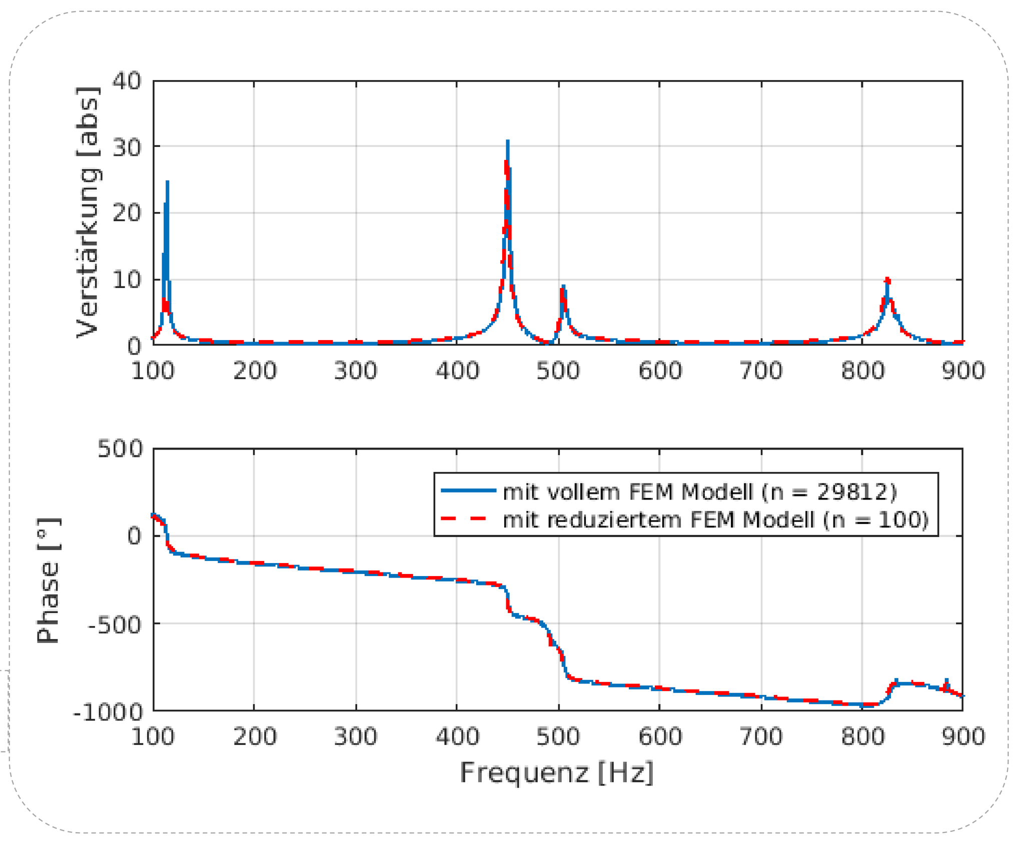
```
>> sysGasturbine = connect(sysRohr, sysFEM)
```

Reduktion des FEM-Modells mit dem `irka` Algorithmus aus **sssMOR**:

```
>> sysFEM_r = irka(sysFEM)
>> sysGasturbine_r = connect(sysRohr, sysFEM_r)
```

Plot des Übertragungsverhaltens mit der `bode` Funktion aus **sss**:

```
>> bode(sysGasturbine, sysGasturbine_r)
```



sss und **sssMOR** sind Open-Source Toolboxen vertrieben unter GPLv2 um den Austausch im Bereich hochdimensionaler Anwendungen sowie Modellreduktion zu fördern. Mehr Informationen unter www.rt.mw.tum.de/?sss oder www.rt.mw.tum.de/?sssMOR.

[1]MATLAB und die Control System Toolbox (Release 2015b) sind eingetragene Marken von The MathWorks, Inc., Natick, Massachusetts, United States.

[2]SLICOT Benchmark Modelle: http://slicot.org/20-site/126-benchmark-examples-for-model-reduction

Technische Universität München
Fakultät für Maschinenwesen

Lehrstuhl für Regelungstechnik
Prof. Dr.-Ing. habil. Boris Lohmann
www.rt.mw.tum.de
{a.castagnotto, maria.cruz, lohmann}@tum.de

Professur für Thermofluiddynamik
Prof. Wolfgang Polifke, Ph. D.
www.tfd.mw.tum.de
{emmert, jaensch, meindl, polifke}@tfd.mw.tum.de

Für die Förderung bedanken wir uns bei folgenden Institutionen:
Deutsche Forschungsgemeinschaft (DFG), LO408/19-1