KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT DER TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Kard. Mercierlaan 94, 3001 Leuven (Heverlee)

MODEL REDUCTION FOR CONTROL DESIGN

Promotoren:
Prof. Dr. ir. J. Vandewalle
Prof. Dr. ir. B. De Moor

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

Geert SCHELFHOUT

Januari 1996

KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT DER TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Kard. Mercierlaan 94, 3001 Leuven (Heverlee)

MODEL REDUCTION FOR CONTROL DESIGN

Jury:
Prof. Dr. ir. W. Dutré, vice-decaan, voorzitter
Prof. Dr. ir. J. Vandewalle, promotor
Prof. Dr. ir. B. De Moor, promotor
Prof. Dr. ir. S. Van Huffel
Prof. Dr. ir. A. Bultheel
Prof. Dr. ir. O. Bosgra (T.U. Delft)
Prof. Dr. ir. P. Van Dooren (U.C. Louvain)

Proefschrift voorgedragen tot
het behalen van het doctoraat
in de toegepaste wetenschappen

door

Geert SCHELFHOUT

U.D.C. 681.5

Januari 1996

D/1995/7515/12

ISBN 90-5682-012-5

Voorwoord

Dit voorwoord biedt een uitstekende gelegenheid iedereen te bedanken, die bijgedragen heeft tot de wording van dit doctoraat.

Vooreerst gaat mijn dankbaarheid uit naar mijn promotoren, Prof. Vandewalle en Prof. De Moor, omdat zij in eerste plaats mij de mogelijkheid gegeven hebben in hun groep te doctoreren.
Het charisma van Bart De Moor bracht mij naar Leuven; zijn vele ideeën en zijn enthousiasme zijn belangrijke stimulansen geweest voor mijn onderzoek.

Ik wens Prof. Bultheel te danken voor de zorg waarmee hij, als lid van het leescomité, dit proefschrift heeft nagelezen.
Ook dank ik graag Prof. Dutré, Prof. Van Huffel, Prof. Van Dooren, en Prof. Bosgra, voor hun bereidwilligheid deel uit te maken van de examenjury.

De niet geringe bijdrage van Peter Van Overschee tot het onderzoek van de iteratieve algoritmes uit Hoofdstukken 5 en 7 wil ik niet vergeten. Met plezier denk ik ook terug aan de leerrijke elektronische conversaties met Pepijn Wortelboer, wiens doctoraatswerk een belangrijke inspiratiebron voor mij was. I wish to thank also Yi Cheng for his valuable help.

Voorts wil ik de leden van SISTA loven en prijzen voor de vlotte en positieve sfeer waarin ik mocht werken. Dat de groep nu te groot is om allen op te sommen is bijna een vaste formule geworden in onze thesisvoorwoorden; in het bijzonder gedenk ik het goede en aangename gezelschap van Jan Van Impe, Marc Moonen, Peter Vanoverschee, Lieven Vandenberghe, en Søren Holdt Jensen.

Ook bedank ik graag Ida Tassens, op wiens hulpvaardigheid ik verschillende keren beroep mocht doen gedurende de korte tijd dat zij bij SISTA is.

Dan is er natuurlijk nog een hele resem mensen, wiens bijdrage aan dit doctoraat moeilijker onder woorden te brengen is. Ik kan ze niet allemaal vermelden,

maar mijn ouders en schoonouders verdienen toch mijn speciale dank, en bovenal Christina, om haar niet aflatende liefdevolle zorg voor mijn lichamelijk en geestelijk welzijn.

Tenslotte wens ik ook mijn erkentelijkheid uit te spreken jegens het N.F.W.O., voor de adequate financiële en administratieve ondersteuning die ik genoten heb.

Geert Schelfhout
Leuven, 1995

Abstract

This dissertation gives a number of contributions to the field of controller reduction, by providing some new frequency weighted or time-domain weighted model reduction procedures, and showing equivalences of certain reduction algorithms.

1. It is proven that Enns's well-known reduction scheme Frequency Weighted Balanced Truncation (FWBT) with certain performance-oriented weights is equivalent to Closed Loop Balanced Truncation (CLBT), a recently proposed technique which is less costly to compute.
   Furthermore, it is shown that the computations for CLBT with observer-based controllers are structured, so that more savings are realized.
   The same approach is developed for FWBT with a stability-oriented weight.
   For the special case of H_2-optimal controllers, equivalence of CLBT and BCRAM, an older reduction technique, is demonstrated.

2. FWBT, weighted with the inverse of the outer factor of the system to be reduced, is shown to be an attractive relative error approximation method, as the computational cost can be reduced significantly with respect to 'vanilla' FWBT.
   The algebraic Riccati equation that needs to be solved for this is also useful when computing a Balanced Stochastic Truncation (BST).

3. A novel iterative algorithm for H_2-optimal model reduction is put forward. An extension of this algorithm, allowing frequency weighting, is given.

4. The same type of algorithm is developed for the Hilbert-Schmidt-Hankel norm, also with frequency weighting.

5. Two time-domain weighted approximation procedures are proposed, one which is optimal with respect to a finite-horizon H_2-norm, and a suboptimal one based on balancing.

Beknopte inhoud

Deze doctoraatsverhandeling levert een aantal bijdragen tot het domein van reductie van regelaars, door enkele nieuwe frequentie- of tijdsgewogen modelreductietechnieken aan te brengen, en door bepaalde equivalenties tussen reductiealgoritmes aan het licht te brengen.

1. Er wordt aangetoond dat Enns' algoritme voor frequentiegewogen balanceren (FWBT) met een weging gericht op behoud van performantie equivalent is met het recent voorgestelde gesloten-lus balanceren (CLBT).
   Voorts wordt duidelijk gemaakt hoe voor regelaars gebaseerd op toestandschatters, CLBT sneller kan opgelost worden.
   Een gelijkaardige aanpak wordt uitgewerkt voor FWBT met een weging gericht op behoud van stabiliteit.
   Voor het bijzondere geval van H_2-optimale regelaars wordt bewezen dat CLBT equivalent is met een ouder algoritme, nl. BCRAM.

2. FWBT met als weging de inverse van de minimum-fase factor van een systeem wordt voorgesteld als een aantrekkelijke relatieve-fout modelreductiemethode.
   Voorts wordt een alternatieve procedure voor Balanced Stochastic Truncation (BST) uitgewerkt, waarbij dezelfde Riccativergelijking dient opgelost te worden.

3. Een nieuw iteratief algoritme wordt aangebracht voor H_2-optimale modelreductie evenals een uitbreiding voor frequentiegewogen H_2 modelreductie.

4. Een algoritme van hetzelfde type, ook met frequentieweging, wordt ontwikkeld voor de Hilbert-Schmidt-Hankel norm.

5. Twee tijdsgewogen modelreductietechnieken worden voorgesteld, één optimaal m.b.t. een kwadratisch foutcriterium met een eindige tijdshorizon, en één gebaseerd op balanceren.

Glossary

Symbols

$\left[\ \cdot\ \right]$	:	general matrix or vector
$\left[\ \cdot\ \right]^T$	:	transpose of a matrix or vector
$\left[\ \cdot\ \right]^*$	:	complex conjugate transpose of a matrix or vector
$\left[\begin{array}{c\|c} A & B \\ \hline C & D \end{array}\right]$	=	$C(sI-A)^{-1}B+D$ or $C(zI-A)^{-1}B+D$
$[A,B,C,D]$	=	$C(sI-A)^{-1}B+D$ or $C(zI-A)^{-1}B+D$
$\\|\cdot\\|_2$	:	H_2 norm of a system or 2-norm of a signal
$\\|\cdot\\|_\infty$	:	H_∞ norm of a system
$\\|\cdot\\|_{HSH}$	:	Hilbert-Schmidt-Hankel norm of a system
$\\|\cdot\\|_H$	:	Hankel norm of a system
$\\|\cdot\\|_F$	:	Frobenius norm of a matrix or vector
$\bar{\sigma}(\cdot)$	:	maximal singular value of a matrix
$\bar{\lambda}(\cdot)$	:	spectral radius of a matrix
x_i	:	ith element of vector x
X_{ij}	:	element of matrix X on row i and column j
$\det(X)$	:	determinant of matrix X
$\mathrm{Tr}\ (X)$	:	trace of matrix X
I_n	:	identity matrix of size $n \times n$
$O_{m\times n}$	:	zero matrix of size $m \times n$
$A \otimes B$	:	Kronecker product
j	=	$\sqrt{-1}$
$\mathcal{B}_T$	:	bilinear transformation
$\mathcal{M}$ ($\mathcal{M}^d$)	:	the set of all FD LTI CT (DT) systems
$\mathcal{M}_r$ ($\mathcal{M}_r^d$)	:	the set of all FD LTI CT (DT) systems of order at most r
$\mathcal{S}$ ($\mathcal{S}^d$)	:	the set of all stable FD LTI CT (DT) systems
$\mathcal{S}_r$ ($\mathcal{S}_r^d$)	:	the set of all stable FD LTI CT (DT) systems of order at most r
T	:	the unit circle in the complex plane

Acronyms

1-DOF	:	one-degree-of-freedom
ARE	:	Algebraic Riccati Equation
BST	:	Balanced Stochastic Truncation
BT	:	Balanced Truncation
CLBT	:	Closed-Loop Balanced Truncation
CT	:	Continuous-Time
DT	:	Discrete-Time
FD	:	Finite-Dimensional
FIR	:	finite impulse response
FWBT	:	Frequency Weighted Balanced Truncation
IIR	:	infinite impulse response
IOWBT	:	Inverse Outer Factor Weighted Balanced Truncation
LMI	:	Linear Matrix Inequality
LTI	:	Linear Time Invariant
MAMS	:	Multiple Actuator Multiple Sensor
MIMO	:	Multiple Input Multiple Output
MISO	:	Multiple Input Single Output
QRD	:	QR Decomposition
SASS	:	Single Actuator Single Sensor
SIMO	:	Single Input Multiple Output
SISO	:	Single Input Single Output
S.P.	:	Standard Plant
STLS	:	Structured Total Least Squares
SVD	:	Singular Value Decomposition

Contents

Nederlandse samenvatting

Hoofdstukken 1 en 2 : inleiding.

De moderne regeltechniek is ver geëvolueerd sedert het pionierswerk van Nyquist en Bode, en biedt de ontwerper een zeer uitgebreid arsenaal aan werktuigen en middelen voor de synthese van regelsystemen. Desalniettemin vinden deze moderne technieken slechts moeilijk ingang in de dagdagelijkse praktijk.

Ten dele is dit te wijten aan het feit dat de 'moderne regelaars', die het resultaat zijn van de moderne methodes, veelal een vrij hoge complexiteit hebben, zodat de implementatie ervan moeilijk of onaantrekkelijk wordt. Moderne regeltechniek gaat uit van een model, dat om accuraat te zijn vaak een hoge orde heeft. Meestal is de resulterende regelaar van minstens even hoge orde : H_2 en H_∞ regelaars hebben dezelfde orde als het gebruikte procesmodel; bij andere technieken, zoals μ-synthese voor ontwerp van robuuste regelaars of convexe optimalisatie over een affiene parametrisatie van de zogenaamde Youla-parameter, is de orde van de regelaar nog hoger. Niet zelden verkrijgt men op die manier regelaars met een orde die in de tientallen loopt.

Benadering van zo'n regelaar door een regelaar van lagere orde is dikwijls noodzakelijk en nog vaker wenselijk : zowel de benodigde rekentijd als de kost van de hardware stijgt snel in functie van de orde van de regelaar, terwijl een overdreven complexiteit van een regelaar slechts de begrijpelijkheid en handelbaarheid ervan vermindert.

Eén mogelijke aanpak bestaat erin één of ander algoritme voor rationele approximatie toe te passen, d.w.z. een modelreductietechniek die het verband tussen ingangen en uitgangen van de regelaar C in open regelkring zo goed mogelijk tracht te benaderen door een systeem C_r van orde r, m.a.w. de approximatiefout $C - C_r$ volgens een of ander criterium zo klein mogelijk probeert te maken, zonder rekening te houden met de gevolgen voor het gedrag van de gesloten regelkring.

Deze naïeve methode heeft het voordeel van de eenvoud, maar een nadeel is het fenomeen *spill-over* : ogenschijnlijk onmerkbare dynamische verschijnselen in open lus kunnen aanleiding geven tot onaanvaardbaar slecht regelgedrag, bv. tot verlies van stabiliteit, in gesloten lus. Dit is logisch : het is bv. een elementair regeltechnisch feit dat het deel van een Nyquist-curve dat dichtst bij het 'kritisch punt' -1 ligt bepalend is voor het uiteindelijk gedrag, en bijgevolg is het van groter belang het gedrag van de regelaar goed te benaderen binnen de bijbehorende frequentie-intervallen dan erbuiten.

Het is dan ook niet verwonderlijk, dat frequentiegewogen modelreductie (die $W_o(C - C_r)W_i$ tracht 'klein' te maken, waarbij W_i een wegingssysteem is aan de ingang en W_o aan de uitgang) onder de meer geavanceerde technieken voor approximatie van regelaars een belangrijke plaats inneemt.

De meest gangbare manier om het 'groot' of 'klein' zijn van de approximatiefout $C - C_r$ te meten is het gebruik van een of andere systeemnorm $\|\cdot\|$, al dan niet met frequentieweging : men probeert $\|W_o(C - C_r)W_i\|$ klein te maken.

Bekende voorbeelden zijn de H_2 norm, gedefinieerd (voor een LTI continu systeem $G(s) \triangleq [A, B, C, 0]$) als

$$\begin{aligned} \|G(s)\|_2 &\triangleq \left(\frac{1}{2\pi}\int_{\omega=-\infty}^{\infty} \text{Tr } (G(-j\omega)^T G(j\omega))d\omega\right)^{\frac{1}{2}} \\ &= \left(\text{Tr } (C\int_0^{\infty} e^{At}BB^Te^{A^Tt}dtC^T)\right)^{\frac{1}{2}} , \end{aligned}$$

alsook de H_∞ norm, gedefinieerd door

$$\|G(s)\|_\infty \triangleq \sup_{\omega\in\mathbb{R}} \bar{\sigma}(G(j\omega))$$

en de Hankel norm, gedefinieerd als

$$\|G(s)\|_H \triangleq \sqrt{\bar{\lambda}(PQ)} ,$$

waarbij $P \triangleq \int_0^\infty e^{At}BB^Te^{A^Tt}dt$ de *controleerbaarheidsgramiaan* is van $[A, B, C, 0]$ en $Q \triangleq \int_0^\infty e^{A^Tt}C^TCe^{At}dt$ de *observeerbaarheidsgramiaan*.

Deze normen hebben welbekende interpretaties : het kwadraat van de H_2 norm is het gemiddeld vermogen in het uitgangssignaal, wanneer het systeem aangestuurd wordt door witte gaussiaanse ruis met variantie 1; het kwadraat van de H_∞ norm is de maximale energie in het uitgangssignaal, wanneer het systeem aangestuurd wordt door een ingangssignaal met energie niet groter dan 1; het kwadraat van de Hankel norm is de maximale energie in het uitgangssignaal na een tijdstip t, wanneer het systeem tot het tijdstipt t aangestuurd wordt door een ingangssignaal met energie niet groter dan 1.

Een vierde norm die in dit doctoraatswerk aan bod komt is de Hilbert-Schmidt-Hankel norm :

$$\|G(s)\|_{HSH} \triangleq \left(\text{Tr } (C \int_0^\infty te^{At}BB^Te^{A^Tt}dtC^T) \right)^{\frac{1}{2}} .$$

Deze is verwant aan de drie voorgaande door zijn interpretatie : het kwadraat van de Hilbert-Schmidt-Hankel norm is de gemiddelde energie in het uitgangssignaal na een tijdstip t, wanneer het systeem tot het tijdstipt t aangestuurd wordt door witte gaussiaanse ruis met variantie 1.

Hoewel de definitie van de Hilbert-Schmidt-Hankel norm en die van de H_2 norm weinig van elkaar verschillen, is de eerste topologisch equivalent met de H_∞ norm en de tweede niet.

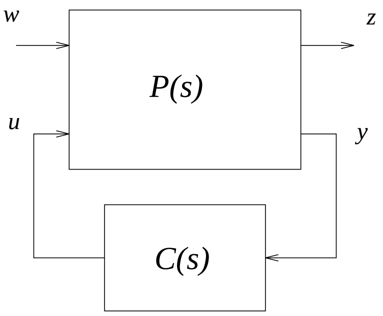
Analoge definities gelden voor discrete systemen.

Zoals gezegd kan frequentieweging belangrijk zijn bij de reductie van regelaars, maar *welke* frequentieweging men moet gebruiken, werd nog niet vermeld.

Een veel gebruikte weging heeft tot doel de stabiliteit van de gesloten regelkring te behouden. Hierbij wordt uitgegaan van een resultaat voor robuuste stabiliteit : als een regelaar C van orde n_c een proces P_{yu} van orde n_p stabiliseert, dan zal elke perturbatie van deze regelaar $C + \delta C$, waarbij $C + \delta C$ evenveel onstabiele polen heeft als C en de onzekerheid δC voldoet aan de voorwaarde $\|(I - P_{yu}C)^{-1}P_{yu}\delta C\|_\infty < 1$, eveneens P_{yu} stabiliseren. Hoewel de approximatiefout $C - C_r$ niet als een onzekerheid dient beschouwd te worden, is $C - C_r$ vóór modelreductie uiteraard onbekend, en daar $\|(I - P_{yu}C)^{-1}P_{yu}(C - C_r)\|_\infty < 1$ het behoud van stabiliteit garandeert, is het duidelijk dat het zo klein mogelijk maken van $\|(I - P_{yu}C)^{-1}P_{yu}(C - C_r)\|_\infty$ de kans bevordert dat C_r P_{yu} stabiliseert. Een stabiliteitsgerichte frequentieweging is dus $W_s \triangleq (I - P_{yu}C)^{-1}P_{yu}$. Dit wegingssysteem heeft orde $n_p + n_c$.

Een andere weging is gericht op het behoud van performantie, waarbij performantie in de ruimste zin van het woord gedefinieerd kan worden als een kwantitatieve maat voor de geschiktheid van het gedrag van de gesloten regelkring. Vaak poogt men performantie uit te drukken als een norm van een overdrachtsmatrix $\|H_{zw}\|$, met $H_{zw}(C) \triangleq L_f(P, C) = P_{zw} + P_{zu}C(I - P_{yu}C)^{-1}P_{yw}$, waarbij $P = \begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix}$ een klassiek 'standaardproces' ('Standard Plant' of S.P.) is (cf. Fig. 0.1).

$$P(s) \triangleq \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$



Figuur 0.1: Voorstelling en toestandsmodel van een standaardproces.

Weze C een regelaar van hoge orde met een goede performantie, die dient vervangen te worden door een regelaar C_r van lagere orde. Aangezien

$$\begin{aligned} H_{zw}(C_r) - H_{zw}(C) &= P_{zu}(I - C_r P_{yu})^{-1}(C_r - C)(I - P_{yu}C)^{-1}P_{yw} \\ &\approx P_{zu}(I - CP_{yu})^{-1}(C_r - C)(I - P_{yu}C)^{-1}P_{yw} \ , \end{aligned}$$

kan men, om de goede performantie te behouden, $\|W_{p,l}(C_r - C)W_{p,r}\|$ klein proberen te maken, waarbij $W_{p,l} \triangleq P_{zu}(I - CP_{yu})^{-1}$ en $W_{p,r} \triangleq (I - P_{yu}C)^{-1}P_{yw}$.

Minimale realisaties voor deze wegingen, met orde $n_p + n_c$, zijn :

$$\begin{aligned} W_{p,l} &= \left[\begin{array}{cc|c} A + B_u D_c C_y & B_u C_c & B_u \\ B_c C_y & A_c & 0 \\ \hline C_z + D_{zu} D_c C_y & D_{zu} C_c & D_{zu} \end{array}\right] \\ \text{en} \qquad W_{p,r} &= \left[\begin{array}{cc|c} A + B_u D_c C_y & B_u C_c & B_w + B_u D_c D_{yw} \\ B_c C_y & A_c & B_c D_{yw} \\ \hline C_y & 0 & D_{yw} \end{array}\right] . \end{aligned}$$

Een relevante vraag i.v.m. modelreductie van MIMO regelaars is steeds, wat het effect is van een verandering van de eenheden, waarin de in- en uitgangen uitgedrukt worden, op de gereduceerde regelaar. Voor frequentiegewogen reductie met de performantie-gerichte wegingen van hierboven vindt men snel dat zo'n verandering van schaal geen invloed heeft. Reductie met de stabiliteits-gerichte weging wordt echter wel beïnvloed : als u vervangen wordt door $D_u^{-1}\bar{u}$ en y door $D_y^{-1}\bar{y}$, waarbij D_u en D_y positief definiete diagonale matrices zijn, krijgt men $\bar{u} = \bar{C}\bar{y} = D_u C D_y^{-1}\bar{y}$, en $\bar{P}_{yu} = D_y P_{yu} D_u^{-1}$, en

$$\|(I - \bar{P}_{yu}\bar{C})^{-1}\bar{P}_{yu}(\bar{C} - \bar{C}_r)\|_\infty = \|D_y(I - P_{yu}C)^{-1}P_{yu}(C - C_r)D_y^{-1}\|_\infty \ .$$

De keuze van D_y kan uiteraard ingrijpende invloed hebben op de verzameling van regelaars C_r waarvoor deze uitdrukking kleiner is dan 1. Een procedure om zowel D_y als C_r te gebruiken om de gewogen norm klein genoeg te maken is 'Algoritme' 2.1 :

$D_0 = I$
$k = 0$
herhaal $\{$

$$
\begin{array}{rcl}
k & \leftarrow & k+1 \\
C_r^{[k]} & = & \operatorname{appr}\min\limits_{C_r} \|D_{k-1}(I - P_{yu}C)^{-1}P_{yu}(C - C_r)D_{k-1}^{-1}\|_\infty \\
D_k & = & \arg\min\limits_{D_y} \|D_y(I - P_{yu}C)^{-1}P_{yu}(C - C_r)D_y^{-1}\|_\infty
\end{array}
$$

$\}$ **tot convergentie bereikt is**

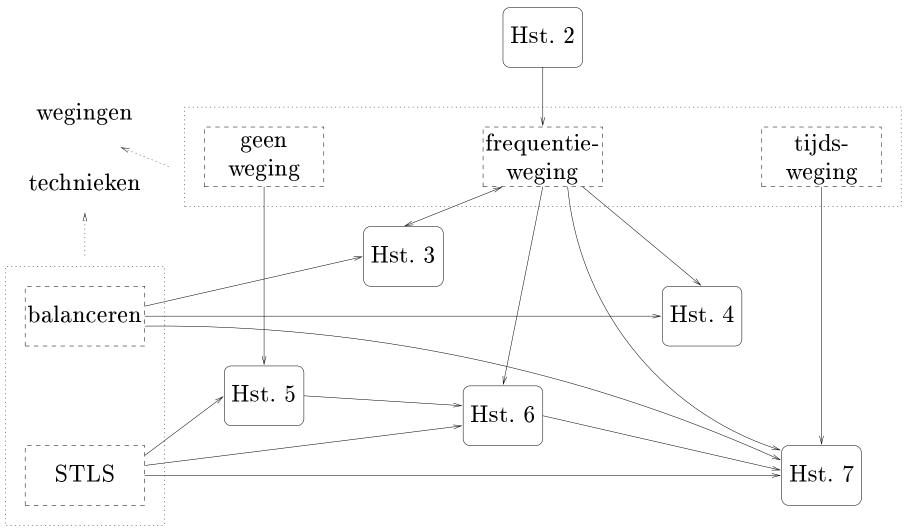
Deze procedure is nauw verwant aan de DK-iteratie voor μ-synthese, waarbij een modelreductie van de regelaar geïncorporeerd wordt in de iteratielus.

De verbanden tussen de gebruikte technieken en wegingsmethodes enerzijds en de verschillende hoofdstukken anderzijds wordt aangegeven door Fig. 0.2.

Hoofdstuk 3 : 'gesloten lus' balanceren.

Een welbekende modelreductietechniek is 'balanceren en afkappen' of 'Balanced Truncation' (BT) geïntroduceerd door Moore [195]. Hierbij zoekt men een realisatie van een systeem G waarvoor de controleerbaarheidsgramiaan P en de observeerbaarheidsgramiaan Q van G diagonaal zijn en gelijk aan elkaar ('balanceren'). De waarden op de diagonaal (de *Hankel singuliere waarden* genoemd) worden gerangschikt van groot naar klein; de toestandsvariabelen corresponderend met de kleinste Hankel singuliere waarden worden dan gewoon weggelaten ('afkappen'). De eenvoud van deze methode, samen met enkele gunstige eigenschappen ervan (zoals bv. een bovengrens op H_∞ norm van de benaderingsfout), maken dit tot één van de meest populaire approximatiemethodes.

Enns [93] heeft een frequentie-gewogen uitbreiding van de methode van Moore voorgesteld, genaamd 'Frequency Weighted Balanced Truncation' (FWBT). Met



Figuur 0.2: Verbanden tussen hoofdstukken, technieken en wegingsmethodes.

in- en uitgangswegingssystemen W_i en W_o van orde n_i resp. n_o, worden hier P en Q vervangen door P_g en Q_g, de submatrices van de controleerbaarheids-gramiaan van GW_i en van de observeerbaarheidsgramiaan van W_oG, die bij de n toestandsvariabelen van G horen. Ook deze methode is in de praktijk zeer nuttig, hoewel men geen bruikbare bovengrens voor de benaderingsfout heeft. Twee Lyapunov vergelijkingen dienen hiervoor opgelost, de ene van dimensie $n + n_i$, de andere van dimensie $n + n_o$.

Voor de stabiliteitsgerichte weging met W_s komt dit neer op een $n_c \times n_c$ en een $(n_p + 2n_c) \times (n_p + 2n_c)$ Lyapunov vergelijking; hier wordt aangetoond dat een $n_c \times n_c$ en een $(n_p + n_c) \times (n_p + n_c)$ Lyapunov vergelijking volstaan.

Voor FWBT met de performantiegerichte wegingen $W_{p,l}$ en $W_{p,r}$ worden de Lyapunov-vergelijkingen op soortgelijke wijze verkleind tot twee $(n_p + n_c) \times (n_p + n_c)$ Lyapunov-vergelijkingen, die dezelfde blijken te zijn als deze die moeten opgelost worden voor 'Closed-Loop Balanced Truncation' (CLBT), een 'balancerende' reductietechniek voorgesteld door Wortelboer e.a. [302]. Hiermee wordt ook een nieuwe interpretatie gegeven voor CLBT.
Voor het veel voorkomende geval $n_c = n_p$ betekent dit dat men ongeveer 3.5 maal minder rekenoperaties nodig heeft dan met de algemene oplossingsmethode.

Voor regelaars, waarbij $u = K\hat{x}$, met $\hat{x}$ een geschatte toestand geproduceerd door een estimator met dynamica $\dot{\hat{x}} = A\hat{x} + B_u u + L(C_y\hat{x} - y)$, kan CLBT met minder rekenoperaties uitgevoerd worden. In dit geval kan men aantonen dat P_g en Q_g submatrices zijn van de controleerbaarheidsgramiaan $\tilde{P}_1$ van $\left[\begin{array}{cc|c} A+K_eC_y & 0 & B_w+K_eD_{yw} \\ -K_eC_y & A+B_uK_r & -K_eD_{yw} \\ \hline C_z & C_z+D_{zu}K_r & D_{zw} \end{array}\right]$, resp. de observeerbaarheidsgramiaan $\tilde{Q}_2$ van $\left[\begin{array}{cc|c} A+B_uK_r & B_uK_r & B_w \\ 0 & A+K_eC_y & -B_w-K_eD_{yw} \\ \hline C_z+D_{zu}K_r & D_{zu}K_r & D_{zw} \end{array}\right]$:

$$\tilde{P}_1 = \begin{bmatrix} \times & \times \\ \times & P_g \end{bmatrix} \text{ en } \tilde{Q}_2 = \begin{bmatrix} \times & \times \\ \times & Q_g \end{bmatrix}.$$

De Lyapunov-vergelijkingen voor de berekening van $\tilde{P}_1$ en $\tilde{Q}_2$ vallen uiteen in zes Sylvestervergelijkingen, die ongeveer viermaal minder flops vergen dan de gewone manier om P_g en Q_g te berekenen. Analoge vergelijkingen voor discrete regelaars zijn eenvoudig te vinden.

Een specifiek voorbeeld van dergelijke regelaars verkrijgt men d.m.v. H_2 synthese. In dit geval kan men bewijzen dat CLBT equivalent is aan een bestaande reductietechniek voor regelaars, nl. 'Balanced Controller Reduction Algorithm Modified' (BCRAM) [312]. Hoewel dit een vereenvoudiging van de formules inhoudt, is het aantal benodigde rekenoperaties ongeveer hetzelfde als voor CLBT.

Voor regelaars gebaseerd op een toestandschatter kan ook FWBT met stabiliteitsgerichte weging W_s op soortgelijke wijze met minder berekeningen gedaan worden; men vindt dat de grootste van de twee Lyapunov-vergelijkingen kan vervangen worden door drie Sylvestervergelijkingen.

Ook voor de reductie van gedecentraliseerde regelaars is de idee die aan de basis ligt van CLBT nuttig. Gedecentraliseerde regelaars hebben een blokdiagonale of ontkoppelde structuur, d.w.z. dat de gemeten uitgangen y en de actuatoringangen u van het te regelen proces ingedeeld worden in groepen y_i en u_i, waarbij terugkoppeling enkel mogelijk is van uitgang y_i naar ingang u_i d.m.v. een (dynamische) regelaar C_i. De reductietechniek komt voor dit soort regelaars neer op het toepassen van CLBT op elke 'sub-regelaar' in aanwezigheid van alle andere sub-regelaars in de terugkoppellus.

Hoofdstuk 4 : relatieve fout modelreductie

Voor vierkante systemen wordt de relatieve fout van een approximatie G_r t.o.v. het originele systeem G gedefinieerd als $G^{-1}(G_r - G)$ of $(G_r - G)G^{-1}$.
Relatieve fout modelreductie probeert een of andere maat van de relatieve fout klein te maken, bv. $\|G^{-1}(G_r - G)\|_{L_\infty}$ of $\|G^{-1}(G_r - G)\|_{L_2}$.

Verscheidene redenen kunnen ingeroepen worden voor dergelijke reductiemethodes; een mogelijke motivatie in het domein van de regeltechniek is dat voor servoregelaars, die een referentie-ingang zo goed mogelijk proberen te volgen, relatieve fout modelreductie bijna equivalent is aan frequentiegewogen modelreductie met stabiliteit-gerichte weging. Omdat relatieve fout modelreductie toch altijd conservatiever is m.b.t. behoud van stabiliteit dan de laatstgenoemde gewogen modelreductie, moet er een bijkomende reden zijn om relatieve fout modelreductie te verkiezen; voor één welbekende methode, 'Balanced Stochastic Truncation' (BST), kan deze reden gevonden worden in het het voorhanden zijn van een vooraf uitrekenbare bovengrens op de H_∞ norm van de relatieve fout. Voor een andere methode, die hier wordt voorgesteld, ligt het voordeel in het kleiner aantal rekenoperaties dat nodig is.

Weze G een systeem van orde n. Relatieve fout modelreductie kan beschouwd worden als een bijzonder geval van frequentiegewogen modelreductie, waarbij als weging G^{-1} genomen wordt. Het ligt voor de hand de hogervermelde methode van Enns (FWBT) op zijn bruikbaarheid hiervoor te onderzoeken. Voor Enns' methode moet de weging echter stabiel zijn; voor minimum-fase systemen is dit zo, en voor dit geval heeft Zhou [320, 321] reeds aangetoond dat FWBT met weging G^{-1} eenvoudiger uit te rekenen is dan met een algemene weging, met name door twee $n \times n$ Lyapunov-vergelijkingen op te lossen. Voorts heeft Zhou voor dit geval de equivalentie van deze methode met BST bewezen.
Voor niet-minimum-fase systemen is de weging G^{-1} onstabiel; hier wordt voorgesteld deze te vervangen door W^{-1}, waarbij W minimum-fase is en $WW^\sim = GG^\sim$. De $2n \times 2n$ gramiaan bij FWBT heeft dan een structuur die toelaat P_g en Q_g sneller te vinden, wat leidt tot

Stelling 4.2
Weze $G(s) \triangleq [A, B, C, D]$ een vierkant continu stabiel inverteerbaar systeem,
Weze $W(s)$ een minimum-fase systeem zodat $W(s)W^T(-s) = G(s)G^T(-s)$,
Weze $G_r(s)$ het systeem van orde r verkregen met FWBT met $W^{-1}(s)$ als weging aan de uitgangszijde,
Definieer $A_i \triangleq A - BD^{-1}C$, $C_i \triangleq -D^{-1}C$, en $A_{wi} \triangleq A_i + XC_i^T C_i$,
Weze X de stabiliserende oplossing van $A_i X + X A_i^T + X C_i^T C_i X = 0 \qquad (4.2),$

Dan wordt $G_r(s)$ gevonden door een realisatie van $G(s)$ waarvoor P_g en Q_g gebalanceerd zijn, waarbij P_g en Q_g voldoen aan

$$\begin{aligned} AP_g + P_gA^T + BB^T &= 0 \qquad (4.1) \\ A_{wi}^TQ_g + Q_gA_{wi} + C_i^TC_i &= 0\,, \qquad (4.3) \end{aligned}$$

af te kappen tot orde r.

Dit algoritme wordt gedoopt 'IOWBT' (Inverse Outer Factor Weighted Balanced Truncation).

De Riccativergelijking (4.2) heeft geen constante term, wat ze eenvoudiger op te lossen maakt. Vooreerst zijn er twee speciale oplossingen, nl. $X = 0$, wat correspondeert met het minimum-fase geval bestudeerd door Zhou, en $X = Q_i^{-1}$, waarbij $A_i^TQ_i + Q_iA_i + C_i^TC_i = 0$; dit laatste geval komt overeen met het 'maximum-fase' geval.
Voor het algemene geval kan men baat halen uit de structuur van de Hamiltoniaan die hoort bij (4.2) : men kan hiervoor direkt een Schur decompositie neerschrijven, die dan d.m.v. orthogonale transformaties opnieuw geordend wordt.

Een analoge stelling en oplossingswijze kan men formuleren voor discrete systemen.

De meest courante relatieve fout reductietechniek, BST, wordt typisch uitgerekend door het oplossen van de vergelijkingen

$$AP + PA^T + BB^T = 0$$

$$B_w \triangleq PC^T + BD^T$$

$$A^TQ_w + Q_wA + (C^T - Q_wB_w)(DD^T)^{-1}(C - B_w^TQ_w) = 0 \qquad (4.9)$$

waarbij Q_w zó gekozen is dat $W(s) \triangleq D_w + C_w(sI - A)^{-1}B_w$, minimum-fase is, met $C_w \triangleq D^{-1}(C - B_w^TQ_w)$ en $D_w \triangleq D^T$. Om deze oplossing Q_w van (4.9) te verkrijgen vindt men de antistabiele eigenruimte van de bij (4.9) horende Hamiltoniaan :

$$H = \begin{bmatrix} (A - B_w(DD^T)^{-1}C)^T & C^T(DD^T)^{-1}C \\ -B_w(DD^T)^{-1}B_w & -(A - B_w(DD^T)^{-1}C) \end{bmatrix}.$$

Riccativergelijking (4.2) blijkt ook nuttig te zijn om Q_w te vinden, zodat een snellere rekenwijze mogelijk wordt; dat dit mogelijk is blijkt reeds uit het werk van Anderson [19], doch binnen de context van BST werd dit nooit opgemerkt.

Dit wordt geformuleerd in

Stelling 4.4
Met $X \stackrel{\triangle}{=} Q_w^{-1} - P$, *waarbij* Q_w *een oplossing is van* (4.9), *geldt*

$$A_i X + X A_i^T + X C_i^T C_i X = 0 \qquad (4.2)$$

Omgekeerd is elke $Q_w = (P + X)^{-1}$ *een oplossing van* (4.9).

Om de gewenste oplossing te vinden heeft men volgend
Lemma 4.1
De transmissienullen van $W(s)$ *zijn eigenwaarden van* $-(A_i + X C_i^T C_i)$.

$X = 0$ en $X = Q_i^{-1}$ zijn uiteraard nog steeds oplossingen voor (4.2), maar nu krijgt men $X = 0$ voor een maximum-fase systeem en $X = Q_i^{-1}$ voor een minimum-fase systeem; voor dit laatste geval blijkt op deze manier duidelijk dat BST equivalent is met IOWBT. Ook voor andere systemen kan men dezelfde oplossingsmethode gebruiken als voor IOWBT, maar dit vergt meer flops voor systemen met relatief weinig niet-minimum-fase nullen. Het is goedkoper de Hamiltoniaan die hoort bij (4.2) blokdiagonaal te maken, hetgeen wel een niet-orthogonale transformatie vereist.

Opnieuw kunnen discrete systemen op analoge wijze behandeld worden.

Een vergelijking leert dat het aantal rekenoperaties dat nodig voor IOWBT, tussen $110n^3$ (voor minimum-fase systemen) en $135n^3$ (voor maximum-fase systemen) ligt, althans wanneer enkel orthogonale transformaties gebruikt worden; met de hogervermelde blokdiagonalisatie blijft dit altijd ongeveer $110n^3$.
Het klassiek algoritme voor BST vergt gemiddeld $250n^3$ flops, de methode van hierboven ongeveer $90n^3$; met enkel orthogonale transformaties varieert dit tussen $110n^3$ voor minimum-fase systemen en $90n^3$ voor maximum-fase systemen.

Een bovengrens op de H_∞ norm van de relatieve fout bij IOWBT is enkel beschikbaar voor minimum-fase systemen; de uitdrukking voor die bovengrens geëxtrapoleerd voor andere systemen wordt echter zelden overschreden; IOWBT biedt dus niet dezelfde garanties als BST, maar is daarom nog niet minderwaardig.
Voor systemen met meer dan r niet-minimum-fase nullen is BST zinloos; hoewel de H_∞ norm van de relatieve fout dan steeds groter is dan 1 voor alle approximanten van orde r, geeft IOWBT meestal een betere benadering voor dergelijke systemen.
Niet-minimumfase nullen blijven bewaard in de BST-approximant; voor IOWBT

is dit niet zo.

Voor niet-vierkante systemen is er geen eenduidige definitie van de relatieve fout. Eén mogelijke definitie is $E_r \triangleq G^\dagger(G-G_r)$ waarbij $G^\dagger$ een pseudo-invers systeem van G is. Voor het gebruik van deze definitie bij modelreductie van regelaars kan men nog steeds een aantal motivaties vinden. Een logische generalisatie lijkt dus, een pseudo-invers $W^\dagger$ te gebruiken als weging bij FWBT. Bij één keuze van $W^\dagger$ verandert enkel vergelijking (4.2), nl. in

$$A_i X + X A_i^T + X C_i^T C_i X - B D^{\perp T} D^{\perp} B^T = 0 \;, \tag{4.19}$$

waarbij $D^\perp$ zo gekozen is dat $D^{\perp T} D^\perp = I - D^\dagger D = I - D^T(DD^T)^{-1}D$. Deze nieuwe vergelijking heeft echter wel een constante term, zodat de voordelen voor de berekening van de oplossing ervan wegvallen.

Voor BST kan ook Stelling 4.4 gegeneraliseerd worden tot het niet-vierkante geval :

Stelling 4.7
Met $X \triangleq Q_w^{-1} - P$, *waarbij* Q_w *een oplossing is van* (4.9), *geldt*

$$A_i X + X A_i^T + X C_i^T C_i X - B D^{\perp T} D^{\perp} B^T = 0 \;.$$

Omgekeerd is elke $Q_w = (P+X)^{-1}$ *een oplossing van* (4.9).

Hoofdstuk 5 : H_2 modelreductie

De H_2-optimale approximant G_r^* van orde r minimaliseert de H_2-norm van de benaderingsfout $G - G_r$, d.w.z.

$$G_r^* = \arg \min_{G_r \in \mathcal{S}_r} \|G - G_r\|_2 \;.$$

Het kwadraat van de H_2-norm is overal afleidbaar, en men kan nodige voorwaarden voor H_2-optimaliteit formuleren d.m.v. de gradiënt. Dit is de grondslag van de meerderheid van de algoritmes voor optimale H_2 modelreductie.

Voor discrete SISO systemen biedt STLS (Structured Total Least Squares) een alternatieve formulering van het probleem. In deze benadering worden de Markov parameters van de approximant als onafhankelijke veranderlijken gekozen, en de eis dat de orde niet groter dan r mag zijn wordt geformuleerd als een voorwaarde op de Hankel matrix gedefinieerd aan de hand van deze parameters. Lagrange vermenigvuldigers worden dan geïntroduceerd om nodige voorwaarden te vinden voor dit probleem van beperkte optimalisatie; deze nodige voorwaarden worden dan omgevormd in een stelsel niet-lineaire vergelijkingen gebracht, die men kan beschouwen als een oplossings-afhankelijke eigenwaardedecompositie. Dit leidt tot een iteratief algoritme dat zeer verschilt van methodes van het gradiënt-type.

Laat $h(z) = \sum_{i=1}^{\infty} h_i z^{-i}$ een discreet, SISO, strict eigenlijk, stabiel systeem voorstellen, en $\hat{h}(z) = \sum_{i=1}^{\infty} \hat{h}_i z^{-i}$ de approximant. De $(p+r)\times 1$ vector $\hat{h}$ bevat de eerste $p+r$ Markov parameters $\hat{h}_i$ van $\hat{h}(z)$. Beschouw eerst een probleem met eindige horizon :

$$\hat{h} = \arg\min_{\hat{h}_i} \frac{1}{2}\sum_{i=1}^{p+r}(h_i - \hat{h}_i)^2 \qquad (5.6)$$

$$\text{zó dat } \exists y \in \mathbb{R}^{(r+1)\times 1} \text{ waarvoor } \hat{H}_p y = 0 \text{ en } y^T y = 1,$$

$$\text{met } \hat{H}_p \triangleq H(\hat{h}, p) \triangleq \begin{bmatrix} \hat{h}_1 & \hat{h}_2 & \hat{h}_3 & \cdots & \hat{h}_{r+1} \\ \hat{h}_2 & \hat{h}_3 & \hat{h}_4 & \cdots & \hat{h}_{r+2} \\ \hat{h}_3 & \hat{h}_4 & \hat{h}_5 & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ \hat{h}_p & \hat{h}_{p+1} & \hat{h}_{p+2} & \cdots & \hat{h}_{p+r} \end{bmatrix}.$$

Als $p \to \infty$, wordt de voorwaarde $\hat{H}_p y = 0$ equivalent aan de eis dat de orde van $\hat{h}(z)$ niet groter is dan r.
D.m.v. p Lagrange vermenigvuldigers l_i en λ voor de voorwaarden $\hat{H}_p y = 0$ en $y^T y = 1$ wordt het probleem hertaald als een onbeperkt optimalisatievraagstuk; afleiding m.b.t. $\hat{h}_i$, y en l_i en λ geeft dan de nodige voorwaarden voor een stationair punt :

$$\begin{aligned} h_i - \hat{h}_i &= (l \star y)_i && (5.8) \\ \hat{H}_p y &= 0 && (5.9) \\ \hat{H}_p^T l &= 0 && (5.10) \\ y^T y &= 1\,. && (5.11) \end{aligned}$$

Hierbij stelt $\star$ de convolutie-operator voor, en $h \in \mathbb{R}^{(p+r)\times 1}$ en l de vectoren met als elementen h_i en l_i.

Als $x \in \mathbb{R}^{m\times 1}$, definieer de $p \times (p+m-1)$ Toeplitz matrix

$$T(x,p) \triangleq \begin{bmatrix} x_1 & x_2 & \cdots & x_m & 0 & \cdots & 0 \\ 0 & x_1 & x_2 & \cdots & x_m & \ddots & \vdots \\ \vdots & \cdots & x_1 & x_2 & \cdots & x_m & 0 \\ 0 & \cdots & 0 & x_1 & x_2 & \cdots & x_m \end{bmatrix} ;$$

definieer verder $H_p \triangleq H(h,p)$, $T_y \triangleq T(y,p)$ en $T_l \triangleq T(l,r+1)$;
dan kan (5.8)-(5.11) herschreven worden als

$$\begin{aligned} \hat{h} &= h - l \star y && (5.12)\\ H_p y &= D_y l && (5.13)\\ H_p^T l &= D_l y && (5.14)\\ \|y\| &= 1\,. && (5.15) \end{aligned}$$

Een meer symmetrische formulering verkrijgt men door l te herschalen : als $\sigma \triangleq \|l\|$ en $x \triangleq l\sigma^{-1}$, wordt dit

$$\begin{aligned} \hat{h} &= h - x \star y\sigma && (5.16)\\ H_p y &= D_y x\sigma && (5.17)\\ H_p^T x &= D_x y\sigma && (5.18)\\ \|y\| = 1 &= \|x\|\,. && (5.19) \end{aligned}$$

De gelijkenis met een singuliere waarde ontbinding wordt hier manifest.
Een uitdrukking voor de objectfunctie is

$$\frac{1}{2}\|h-\hat{h}\|^2 = \frac{1}{2}y^T D_x y\sigma^2 = \frac{1}{2}x^T D_y x\sigma^2 = \frac{1}{2}x^T H y\sigma\,.$$

Verscheidene iteratieve procedures kunnen voorgesteld worden om bovenstaande vergelijkingen op te lossen. Eén methode, Algoritme 5.3, is geïnspireerd door een algoritme om de kleinste eigenwaarde van een matrix te vinden, nl. *inverse power iteration.*

Hoewel deze generalisatie van het algoritme niet noodzakelijk altijd leidt tot de 'singuliere vectoren' horend bij de kleinste σ, blijkt dit meestal toch het geval te zijn; een kleine waarde van σ is meestal ook een goede heuristiek voor een kleine waarde van de objectfunctie $x^T D_y x\sigma^2$.

De vectoren x en y in de nodige voorwaarden kunnen ook op een andere manier genormaliseerd worden, waarbij $y^T D_x y = x^T D_y x = 1$ en bv. $\|x\| = \|y\|$. In

Algoritme 5.3

$k = 0$
Initialiseer y_0, l_0
herhaal {

$$\begin{array}{rcl} k & \leftarrow & k+1 \\ y_k & = & (H^T D_{y_{k-1}}^{-1} H)^{-1} D_{l_{k-1}} y_{k-1} \\ l_k & = & D_{y_{k-1}}^{-1} H y_k \\ \begin{bmatrix} y_k \\ l_k \end{bmatrix} & \leftarrow & \dfrac{1}{\|y_k\|} \begin{bmatrix} y_k \\ l_k \end{bmatrix} \end{array}$$

} zolang $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - T_{l_k}^T y_k$

dit geval is de te minimaliseren objectfunctie gelijk aan σ^2, zodat de singuliere vectoren x en y met de kleinste σ dan ook bij het globale minimum horen. Dit geeft aanleiding tot Algoritme 5.4.

De efficiëntie van deze algoritmes kan nog verbeterd worden.
Eén equivalente maar lichtjes efficiëntere versie wordt voorgesteld in [74].
Ook kan de Toeplitz structuur van D_y en D_l uitgebuit worden.
Voorts kan men D_y en D_l niet iedere iteratie aanpassen, waardoor bepaalde rekenintensieve bewerkingen minder vaak dienen uitgevoerd te worden.

Een generalisatie van deze methode naar MIMO systemen lijkt onmogelijk. Voor SIMO of MISO systemen is het echter wel mogelijk; Algoritme 5.3 wordt dan vertaald tot Algoritme 5.7 (hierbij staat $\otimes$ voor het Kronecker-product).

Als men het probleem met oneindige tijdshorizon beschouwt, d.w.z. de limiet $p \to \infty$ neemt, merkt men in experimenten dat $l^{[k]}(z) \stackrel{\triangle}{=} \lim_{p\to\infty} \sum_{i=1}^{p} l_i z^{-i}$ een rationale functie wordt. Een formulering die afhangt van een eindig aantal parameters lijkt dus in het verschiet. Inderdaad, geformuleerd in het z-domein worden (5.12)-(5.15) :

$$\begin{array}{rcl} \hat{h}(z) & = & h(z) - y(z)l(z) \ , \\ \left[h(z)y(z^{-1})\right]_+ & = & \left[y(z^{-1})y(z)l(z)\right]_+ \ , \end{array}$$

Algoritme 5.4

$k = 0$
Initialiseer u_0, v_0
herhaal {

$$
\begin{array}{rcl}
k & \leftarrow & k+1 \\
v_k & = & (H^T D_{v_{k-1}}^{-1} H)^{-1} D_{u_{k-1}} v_{k-1} \tau_{k-1}^2 \\
u_k & = & D_{v_{k-1}}^{-1} H v_k \tau_{k-1}^{-1} \\
\gamma & \leftarrow & \sqrt{\|T_{u_k}^T v_k\|} \\
\alpha & \leftarrow & \dfrac{\|v_k\|}{\|u_k\|} \\
u_k & \leftarrow & \alpha \gamma^{-1} u_k \\
v_k & \leftarrow & \alpha^{-1} \gamma^{-1} v_k \\
\tau_k & = & u_k^T H v_k
\end{array}
$$

} **zolang** $\|v_k - v_{k-1}\| > \epsilon$
$\hat{h} = h - T_{u_k}^T v_k \tau_k$

$$
\begin{array}{rcl}
\left[h(z) l(z^{-1})\right]_{0\ldots r} & = & \left[l(z^{-1}) l(z) y(z)\right]_{0\ldots r} \ , \\
\|y(z)\| & = & 1 \ ;
\end{array}
$$

hier betekent de notatie $\left[\sum_{i=-\infty}^{\infty} x_i z^{-i}\right]_+ \triangleq \sum_{i=1}^{\infty} x_i z^{-i}$.
De iteratie wordt vertaald in

$$
\begin{array}{rcl}
\left[h(z) y^{[k]}(z^{-1})\right]_+ & = & \left[y^{[k-1]}(z^{-1}) y^{[k-1]}(z) l^{[k]}(z)\right]_+ \ , \\
\left[h(z) l^{[k]}(z^{-1})\right]_{0\ldots r} & = & \left[l^{[k-1]}(z^{-1}) l^{[k-1]}(z) y^{[k-1]}(z)\right]_{0\ldots r} \ .
\end{array}
$$

Hieruit kan men afleiden dat

$$
l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{a(z) \hat{a}^{[k-1]}(z)} \ ,
$$

waarbij $f^{[k]}(z)$ een reële veelterm in z is met $\deg f^{[k]}(z) = n-1$, $a(z)$ de noemerveelterm van $h(z)$, en $\hat{a}^{[k]}(z) \triangleq y^{[k]}(z^{-1})$.

Algoritme 5.7

$k = 0$
Initialiseer y_0, l_0
herhaal {

$$\begin{array}{rcl} k & \leftarrow & k+1 \\ y_k & = & (H^T(D_{y_{k-1}}^{-1} \otimes I_m)H)^{-1} D_{l_{k-1}} y_{k-1} \\ l_k & = & (D_{y_{k-1}}^{-1} \otimes I_m) H y_k \\ \begin{bmatrix} y_k \\ l_k \end{bmatrix} & \leftarrow & \dfrac{1}{\|y_k\|} \begin{bmatrix} y_k \\ l_k \end{bmatrix} \end{array}$$

} **zolang** $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - T_{l_k}^T y_k$

De onbekenden zijn nu niet langer $l_i^{[k]}$, maar de n coëfficiënten van $f^{[k]}(z)$, en natuurlijk ook van $\hat{a}^{[k]}(z)$. Men kan dus komen tot algoritmes waar bij iedere iteratie een eindig lineair stelsel dient opgelost te worden.

Men kan aantonen dat in een optimum $l(z)$ de vorm heeft

$$l(z) = \frac{z^r \hat{a}_r(z) \bar{f}(z)}{a(z)\hat{a}(z)} ,$$

waarbij $\hat{a}(z)$ de noemerveelterm is van $\hat{h}(z)$, en $\hat{a}_r(z) \triangleq z^r \hat{a}(z^{-1})$; vandaar dat in zo'n optimum de benaderingsfout $h(z) - \hat{h}(z) = \frac{\bar{f}(z)\hat{a}(z)}{a(z)}(\frac{\hat{a}_r(z)}{\hat{a}(z)})^2$ het kwadraat van een *all-pass* factor bevat.

Hoofdstuk 6 : frequentiegewogen H_2-approximatie

De resultaten van Hoofdstuk 5 kunnen uitgebreid worden voor SISO frequentiegewogen H_2 modelreductie. Hierbij ligt de nadruk op het ontwikkelen van efficiëntere en meer betrouwbare algoritmes in het z-domein.

Bij frequentiegewogen H_2 modelreductie moet men onderscheid maken tussen strict eigenlijke en inverteerbare approximanten; voor het ongewogen probleem vervalt dit verschil, omdat daar de constante term $\hat{h}_0$ steeds gelijk moet zijn aan h_0. Eerst wordt het algoritme voor strict eigenlijke approximanten behandeld.

Neem een stabiele, minimum-fase en inverteerbare wegingsfunctie $w(z) = \frac{n_w(z)}{d_w(z)}$ van orde w; het beschouwde optimalisatieprobleem is

$$\min_{\hat{h}(z)} \|w(z)(h(z) - \hat{h}(z))\|_2$$

$$\text{met McMillan } \deg \hat{h}(z) \leq r \text{ en } \hat{h}(z) \text{ strict eigenlijk.}$$

Noodzakelijke voorwaarden voor optimaliteit worden hier

$$\left[w(z^{-1})w(z)(h(z) - \hat{h}(z))\right]_+ = l(z)y(z) \ ,$$

$$\left[\hat{h}(z)y(z^{-1})\right]_+ = 0 \ , \ \left[\hat{h}(z)l(z^{-1})\right]_{0\ldots r} = 0 \ , \ \|y\| = 1 \ .$$

Opnieuw kan $\hat{h}(z)$ uit de tweede en derde voorwaarde geëlimineerd worden :

$$\begin{aligned} \left[h(z)y(z^{-1})\right]_+ &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ y(z^{-1})\right]_+ \ , \\ \left[h(z)l(z^{-1})\right]_{0\ldots r} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ l(z^{-1})\right]_{0\ldots r} \ . \end{aligned} \tag{6.2}$$

Een logische generalisatie van de iteratie van Hoofdstuk 5 is dan

$$\begin{aligned} &\left[h(z)y^{[k]}(z^{-1})\right]_{1,\ldots,p} \\ &\quad = \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_+ y^{[k-1]}(z^{-1})\right]_{1,\ldots,p} \ , \end{aligned} \tag{6.6}$$

$$\begin{aligned} &\left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} \\ &\quad = \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ l^{[k-1]}(z^{-1})\right]_{0\ldots r} \ . \end{aligned} \tag{6.7}$$

Definiërend $\xi^{[k]}(z) \triangleq \left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_+$, bewijst men

$$\xi^{[k]}(z) = w(z)\frac{\phi^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)}, \ \phi^{[k]}(z) \text{ een veelterm van graad } n + r - 1 \ . \tag{6.8}$$

Uit de definitie van $\xi^{[k]}(z)$ haalt men dan

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)d_w(z)}, f^{[k]}(z) \text{ een veelterm van graad } n + w - 1 \ .$$

Wanneer men bv. de $n+r+w$ residus $X_i^{[k]}$ van $\xi^{[k]}(z)$ in zijn polen als onbekende veranderlijken kiest, kan men zowel de residus $C_i^{[k]}$ van $l^{[k]}(z)$ als de coëfficiënten van $\hat{a}^{[k]}(z)$ elimineren; men verkrijgt een algoritme waarbij elke iteratie een lineair stelsel van $n+r+w$ vergelijken en $n+r+w$ onbekenden dient opgelost te worden. Dit stelsel is bovendien ook beter geconditioneerd dan datgene, dat men bekomt wanneer men de coëfficiënten van $\phi^{[k]}(z)$ en $f^{[k]}(z)$ als onbekenden neemt.

De voorgestelde algoritmes werken enkel voor discrete systemen.
De bilineaire transformatie bewaart de H_2 norm niet, en kan dus niet gebruikt worden om continue H_2 approximatie problemen om te zetten in discrete. Men kan echter eenvoudig aantonen dat, indien $h_1(z) \triangleq \frac{1}{z+1} h_c(\frac{z-1}{z+1})$ en $w(z) \triangleq w_c(\frac{z-1}{z+1})$ (waarbij $h_c(s)$ het te benaderen systeem voorstelt en $w_c(s)$ de wegingsfunctie), en optimale frequentiegewogen H_2 approximatie van $h_1(z)$ met weging $w(z)$ resulteert in de approximant $\hat{h}_1(z)$, dan $\hat{h}_c(s) \triangleq \frac{2}{1-s}\hat{h}_1(\frac{1+s}{1-s})$ de optimale approximant voor $h_c(s)$ is.

Indien de approximant $\hat{h}(z)$ niet strict eigenlijk hoeft te zijn, m.a.w. als $\hat{h}_0$ verschillend mag zijn van nul, gelden lichtjes verschillende nodige voorwaarden; met de definitie $\left[\sum_{i=-\infty}^{\infty} x_i z^{-i}\right]_{0+} \triangleq \sum_{i=0}^{\infty} x_i z^{-i}$, krijgt men :

$$\begin{aligned} \left[h(z)y(z^{-1})\right]_{+} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_{0+} y(z^{-1})\right]_{+} , \\ \left[h(z)l(z^{-1})\right]_{0\ldots r} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_{0+} l(z^{-1})\right]_{0\ldots r} , \end{aligned} \tag{6.22}$$

en de 'iteratievergelijkingen' worden :

$$\begin{aligned} \left[h(z)y^{[k]}(z^{-1})\right]_{+} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_{0+} y^{[k-1]}(z^{-1})\right]_{+} \\ \left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_{0+} l^{[k-1]}(z^{-1})\right]_{0\ldots r} \end{aligned}$$

Hier wordt $\xi^{[k]}(z)$ gedefinieerd $\xi^{[k]}(z) \triangleq \left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_{0+}$; de uitdrukkingen voor $\xi^{[k]}(z)$ en $l^{[k]}(z)$ blijven geldig, alleen is $\phi^{[k]}(z)$ nu een veelterm van graad $n+r$; dit betekent dat een extra veranderlijke $X_0^k = \xi_0^{[k]}$ moet ingevoerd worden. Uit de definitie van $\xi^{[k]}(z)$ kan men nu een extra relatie afleiden, die toelaat X_0^k te elimineren.

Bepaalde 'niet-generische' gevallen geven aanleiding tot technische complicaties in bovenstaande algoritmes. Wanneer bv. nullen van het wegingsfilter $w(z)$ samenvallen met polen van $h(z)$, gaan bovenstaande werkwijzen ogenschijnlijk niet meer op. Jammer genoeg is dit geen te verwaarlozen uitzondering, maar treedt dit verschijnsel juist op bij de stabiliteit- en performantiegerichte wegingen uit Hoofdstuk 2. De op te lossen vergelijkingen worden hierdoor echter veeleer eenvoudiger dan moeilijker : een aantal variabelen vallen weg.

Een andere complicatie is polen (of nullen van $w(z)$) met multipliciteit groter dan één, polen van $h(z)$ in de oorsprong $z = 0$, strict eigenlijke wegingsfuncties, of wegingsfuncties met polen of nullen op de eenheidscirkel; de machinerieën om deze problemen op te lossen zijn niet moeilijk te vinden.

Het lastigste geval is datgene waar polen van de approximant samenvallen met polen van $h(z)$ en $w(z)$ of nullen van $w(z)$ of multipliciteit groter dan één hebben, omdat men dit niet bij voorbaat kan vaststellen, en men m.a.w. 'dynamisch' moet testen of dit zich voordoet.

Hoofdstuk 7 : modelreductie met tijdsweging

In dit hoofdstuk worden een aantal (deels heuristische) reductiemethodes aangereikt waarbij de approximatiefout gewogen wordt in het tijdsdomein. Als $W_o(t)$ en $W_i(t)$ tijdswegingen zijn, tracht men m.a.w. $W_o(t)(G(t) - G_r(t))W_i(t)$ klein te maken.

Bij tijdsgewogen H_2-optimale modelreductie minimaliseert men

$$\min_{\hat{h}_i} \frac{1}{2} \sum_{i=1}^{p+r} f_i (h_i - \hat{h}_i)^2$$

met de eis dat de orde van $\sum_{i=1}^{\infty} \hat{h}_i z^{-i}$ niet groter is dan r.

De benadering van Hoofstuk 5 is gemakkelijk uit te breiden tot dit probleem en geeft aanleiding tot de volgende nodige voorwaarden :

$$\begin{aligned} h - \hat{h} &= D_f(l \star y) = D_f T_l^T y = D_f T_y^T l \\ Hy &= T_y D_f T_y^T l \\ H^T l &= T_l D_f T_l^T y \end{aligned}$$

waarbij D_f een diagonale matrix is met $[D_f]_{(i,i)} \triangleq \frac{1}{f_i}$.

Algoritme 5.3 wordt dan :

Algoritme 7.1

$k = 0$
Initialiseer y_0, l_0
herhaal {

$$
\begin{aligned}
D_{y_k} &= T_{y_k} D_f T_{y_k}^T \\
D_{l_k} &= T_{l_k} D_f T_{l_k}^T \\
k &\leftarrow k+1 \\
y_k &= (H^T D_{y_{k-1}}^{-1} H)^{-1} D_{l_{k-1}} y_{k-1} \\
l_k &= D_{y_{k-1}}^{-1} H y_k \\
\begin{bmatrix} y_k \\ l_k \end{bmatrix} &\leftarrow \frac{1}{\|y_k\|} \begin{bmatrix} y_k \\ l_k \end{bmatrix}
\end{aligned}
$$

} zolang $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - D_f T_{l_k}^T y_k$

Men kan dit eenvoudig uitbreiden met nog een bijkomende weging in het frequentiedomein.

Een specifiek voorbeeld van een tijdsgewogen H_2 norm is de HSH norm; hierbij is $f_i = i$. De HSH norm is een overal afleidbare functie van de systeemparameters; uitdrukkingen voor de gradiënt van $J \triangleq \| [A, B, C, D] \|_{HSH}^2$ m.b.t. A, B, C en D zijn gemakkelijk af te leiden, en kunnen, zoals bij de H_2 norm, gebruikt worden om nodige voorwaarden te formuleren voor de optimale HSH norm approximant, of als basis voor optimalisatie-algoritmes van het gradiënt-type.

Voor SISO systemen kunnen in dit geval ook de algoritmes van Hoofdstuk 6 aangepast worden. De nodige voorwaarden die op deze manier afgeleid worden zijn

$$
\begin{aligned}
\left[h(z)y(z^{-1})\right]_+ &= -\left[w^{-1}(z)y(z^{-1}) \int z^{-1} \left[w^{-1}(z^{-1})l(z)y(z)\right]_+ dz\right]_+ , \\
\left[h(z)l(z^{-1})\right]_{0\ldots r} &= -\left[w^{-1}(z)l(z^{-1}) \int z^{-1} \left[w^{-1}(z^{-1})l(z)y(z)\right]_+ dz\right]_{0\ldots r} .
\end{aligned}
$$

waarbij de constante term van de onbepaalde integraal nul gekozen wordt, m.a.w. de integraal stelt steeds een strict eigenlijke functie voor.
De iteratie van Algoritme 7.1 ziet er dan als volgt uit :

$$\begin{aligned}
&\left[h(z)y^{[k]}(z^{-1})\right]_+ \\
&\quad = -\left[w^{-1}(z)y^{[k-1]}(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_+ dz\right]_+ , \\
&\left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} \\
&\quad = -\left[w^{-1}(z)l^{[k-1]}(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ dz\right]_{0\ldots r} .
\end{aligned}$$

Met de definitie $\xi^{[k]}(z) \triangleq -\int z^{-1}\left[w^{-1}(z^{-1})\hat{a}^{[k-1]}(z^{-1})l^{[k]}(z)\right]_+ dz$ blijft de uitdrukking voor $\xi^{[k]}(z)$ behouden; wel geldt nu

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{(a(z)\hat{a}^{[k-1]}(z)d_w(z))^2}, f^{[k]}(z) \text{ een veelterm van graad } 2n+2w+r-1 .$$

Soortgelijke, maar ingewikkelder en meer gesofisticeerde methodes kunnen gebruikt worden om de iteratievergelijkingen uit te drukken in functie van de $n+r+w$ residus van $\xi^{[k]}(z)$.

HSH modelreductieproblemen voor continue systemen kunnen gewoon d.m.v. de bilineaire transformatie geconverteerd worden naar een discreet probleem.

Voor een meer heuristische methode, tijdsgewogen balanceren en afkappen, worden eerst tijdsgewogen gramianen gedefinieerd : $P_f \triangleq \int_0^\infty f(t)e^{At}BB^Te^{A^Tt}dt$ en $Q_f \triangleq \int_0^\infty f(t)e^{A^Tt}C^TCe^{At}dt$; in het bijzonder bestuderen we polynomiale functies $f(t)$ en als basis hiervoor $f(t) = t^i$.
Men bewijst dat de matrices

$$X_i \triangleq \int_0^\infty t^i e^{A_1t}De^{A_2t}dt$$

kunnen berekend worden door de volgende recursie uit te voeren :

$$\begin{aligned}
A_1X_0 + X_0A_2 + D &= 0 , \\
A_1X_i + X_iA_2 + iX_{i-1} &= 0 , \qquad i > 0 .
\end{aligned}$$

O.a. met deze recursie kan men P_f en Q_f berekenen voor elke stuksgewijs polynomiale $f(t)$. Voor discrete systemen vindt men analoge uitdrukkingen.

Deze tijdsgewogen gramianen kunnen op voor de hand liggende wijze gebruikt worden om tijdsgewogen normen te berekenen, alsook voor tijdsgewogen balancerende modelreductie. Voor deze reductiemethode kan aangetoond worden, dat voor een bepaalde klasse van systemen het gereduceerde model stabiel blijft als het oorspronkelijk systeem stabiel is.

Besluit

De voornaamste bijdragen van dit doctoraatswerk zijn :

1. Er wordt onderzocht hoe FWBT efficiënter kan worden uitgevoerd bij het gebruik van bepaalde stabiliteit- en performantiegerichte wegingen. We tonen aan dat FWBT equivalent is met de recent voorgestelde techniek CLBT in alle gevallen, en voor het bijzondere geval van H_2-optimale regelaars ook met een ouder algoritme, BCRAM. Deze resultaten houden een aanzienlijke reductie van de rekenkost in. Nog meer besparingen blijken mogelijk voor regelaars die een geschatte toestand statisch terugkoppelen. Een uitbreiding van CLBT voor gedecentraliseerde regelaars wordt voorgesteld.

2. Er wordt onderzocht hoe FWBT kan gebruikt worden als relatieve-fout modelreductietechniek, en efficiënter gemaakt. Ook voor BST wordt een verbetering aangestipt. Dit gebeurt eerst voor vierkante systemen en wordt dan uitgebreid voor niet-vierkante systemen, waar sommige voordelen evenwel verdwijnen.

3. Een iteratief algoritme, steunend op STLS, voor H_2-optimale modelreductie wordt voorgesteld, en uitgebreid voor frequentiegewogen H_2 modelreductie; dit wordt ook gedaan voor de Hilbert-Schmidt-Hankel norm.

4. Een toepassing van STLS voor tijdsgewogen H_2-optimale modelreductie wordt aangestipt.

5. We definiëren tijdsgewogen gramianen, en tonen hoe ze te berekenen. Deze gramianen vinden een toepassing bij het berekenen van tijdsgewogen normen; balanceren van de tijdsgewogen gramianen leidt tot een heuristische tijdsgewogen modelreductietechniek.

Chapter 1

Introduction and motivation

Modern model based control theory is a field which has seen continuous and intensive activity for decades, and consequently has evolved considerably since the work of pioneers such as Bode and Nyquist. An enormous body of knowledge has been collected with respect to analysis and synthesis methods.

Yet in spite of all this diligent work, these results often have not made their way to the 'real world' of practical applications. Assuredly there are many reasons why they have failed to be applied : many have argued that the advances in control and system theory have been driven by mathematical interest and feasibilities rather than engineering needs; yet to say that this is the only reason would be an oversimplification. Conservatism, reluctance to spend time and money on modeling, lack of expertise, are certainly other contributing elements. Every engineer is sure to have his own opinion on this matter, and we shall not go into a detailed analysis of this problem, for fear of starting religious wars. We firmly believe, however, that classical control methods have been preferred because, in spite of their simplicity and constraints, they were most often sufficient. As more and more will be demanded of control systems however, in terms of accuracy, bandwidth, etc., more sophisticated techniques will gradually become unavoidable. Classical control design relies primarily on single input single output controllers, few design parameters to tweak, and basically more analysis than synthesis methodologies. Some processes present difficulties which cannot be overcome with such tools : think of systems with complex unstable dynamics, severely coupled inputs and outputs, non-minimum phase zeroes, etc. Therefore the problems connected with modern control will have to be tackled.

One of these problems is the high complexity of the models and resulting controllers. The order of a model is an important aspect of its complexity, though of course are also other aspects, e.g. the number of inputs and outputs, etc. Many modeling procedures yield models of rather high order, as in general higher order implies better accuracy; this is especially true for spatially discretized systems, obtained via e.g. Finite Element Modeling. Often control design strategies yield controllers of the same order as the process model or higher, depending on the specifications. Classical examples are H_2 [163] and H_∞ control design [86, 164]; a few modern and very powerful control design techniques that result in quite high orders are the convex optimization approach of Boyd et al. [56], and μ-synthesis [88, 30], the most important robust control synthesis method known to date.

That high controller order is a drawback when implementing the controller, is obvious : both the cost of hardware and the time of computation grow steeply when the order is increased. How serious this disadvantage is, clearly depends on the application, e.g. computation time might be unimportant for a slow chemical process. But even if these were of no concern, it can be argued that the implemented order should not be chosen higher than really necessary, since otherwise one only increases the likelihood of failures and obstructs diagnosis; to paraphrase Occam and his razor, 'if you have the choice between two controllers, take the simplest one'.

Fortunately, one often finds that many high-order controllers may be approximated by controllers of considerably lower order without having to relax the specified constraints or impairing the performance too much. Thus model reduction has become an important tool in modern control design.

Closed-loop considerations turn out to be of paramount importance while designing a controller of reduced order. It has been recognized long before now that the control action is critical in some frequency intervals; it is obvious that the controller should be approximated well around those frequencies, while at other less important frequency intervals a larger deviation may be allowed. Dynamics that are almost un-noticeable in open loop may greatly influence the closed-loop behavior, and may cause extremely bad performance when insufficiently captured by a reduced order model.

1.1 Control design

A word or two is appropriate on modern control design methodologies that lead to high order controllers; these are generally used in the context of a standard plant framework.

A standard plant $P(s)$ [56] (Fig. 1.1) is a system where a distinction has been made between inputs and outputs available for control and those not available. The inputs w not available for control are called *exogenous*; they encompass disturbances and signals provided by the user of the control system, rather than by the automatic controller, e.g. a reference signal. The inputs u available for control are labeled *actuator* inputs. The outputs z not available for control are called *performance* outputs; usually the standard plant is set up in such a way that the performance specifications are reflected by a small value of z. The outputs available for control are termed *sensor* or *measurement* outputs.

The transfer matrix and state space representation of a standard plant can be partitioned into four transfer matrices :

$$P(s) = \begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix} = \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$

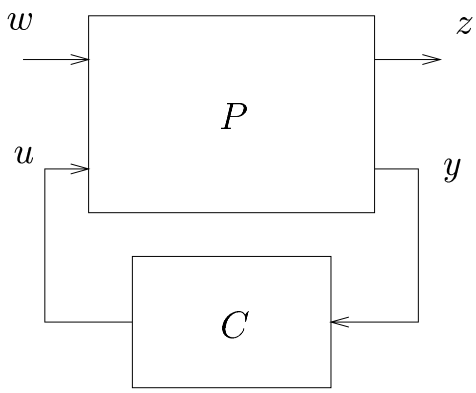


Figure 1.1: Standard plant configuration.

Thus linear control design is the process of finding a linear system $C(s)$ (i.e. the controller), that has y for inputs and u for outputs, in such a way that the closed-loop transfer H_{zw} from the exogenous inputs w to the performance outputs z has certain desirable properties. It is readily found that $H_{zw}(s) = L_f(P(s), C(s)) \triangleq P_{zw}(s) + P_{zu}(s)(I - C(s)P_{yu}(s))^{-1}C(s)P_{yw}(s)$.

Very general control design problems can be formulated as : 'Find a controller $C(s)$ that has y for inputs and u for outputs, such that the resulting closed-loop transfer function $H_{zw}(s) = P_{zw}(s) + P_{zu}(s)(I - C(s)P_{yu}(s))^{-1}C(s)P_{yw}(s)$ is made small.' In which sense the word 'small' is defined here is of course a major part of the specifications; generally it is defined in such a way that smallness includes stability. Well-known modern control design techniques are H_2 control or LQG, minimizing $\|H_{zw}(s)\|_2$, and H_∞ control, minimizing $\|H_{zw}(s)\|_\infty$.

It is easily seen that $P(s)$ may be stabilized by a controller $C(s)$ if and only if $P_{yu}(s) = [A, B_u, C_y, D_{yu}]$ is stabilizable and detectable. A controller $C(s)$ stabilizes such a plant $P(s)$ if and only if it stabilizes $P_{yu}(s)$. Therefore, if one is interested only in a stabilizing controller, there is no advantage in studying the problem in a standard plant framework. Performance specifications, and the effect of disturbance, can only be formulated in a general way by setting up a standard plant that reflects these things.

A landmark result is that the set of all stabilizing controllers may be parameterized as a function of a stable 'parameter system' $Q(s)$, and that the set of all possible closed-loop transfer matrices is affine in this parameter [311, 310, 79]. An important remark, vastly broadening the scope of control problems that can be exactly, though not analytically solved, is that a very broad class of control specifications is convex in any $H_{zw}(s)$ and thus also convex in any affine parameterization of $Q(s)$ [56]. The optimal $Q(s)$ for these problems may be found with any desired degree of accuracy using convex optimization techniques, and the corresponding controller is readily computed. However, one important specification, which concerns us to a high degree, cannot be expressed as a convex constraint : a restriction on the order of $C(s)$, or even $H_{zw}(s)$, is not convex. Thus this design technique certainly does not preclude the necessity of controller reduction, rather the opposite; for the optimal $Q(s)$ for general problems is often infinite-dimensional, and as it is approximated by an affine parameterization, the order of the approximation may have to be rather high for accuracy. Moreover, the order of the controller $C(s)$ is in general the sum of the orders of $Q(s)$ and of $P(s)$, increasing the controller complexity even more.

Last but not least, the μ-synthesis methodology gives rise to high order controllers as well; see also Subsection 2.4.5.

1.2 Controller reduction

As we discussed above, many optimal control design techniques yield controllers of at least the same order as the plant to be reduced; for some methods the order is even likely to be much higher. Since high order is often a serious drawback for a controller, possibly preventing its implementation, controller reduction is a discipline of considerable interest in modern control theory.

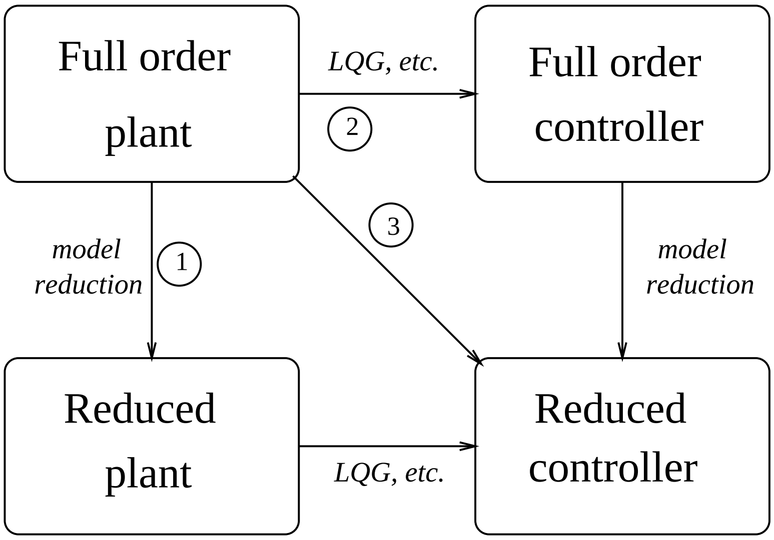


Figure 1.2: The three roads of controller reduction.

A diagram one will often encounter in survey papers on model reduction for control design [23, 132, 143, 302] is shown in Fig. 1.2. It is meant to provide a classification of controller reduction methods into three 'roads'.
First finding a reduced order model $P_r(s)$ for the plant model $P(s)$ and designing a controller for $P_r(s)$, is the first road, labeled *plant reduction*, and perhaps the oldest and most natural one : in the stage of modeling the process to be controlled, one is always inclined to omit dynamic effects that one thinks irrelevant, and to include them in the model only when one reaches the conviction that they are indeed important; this can be regarded as an implicit plant reduction. This approach is now commonly regarded as being fraught with dangers, however : the plant reduction is performed before one has any knowledge about the closed loop, since the controller has not been designed yet. Model reduction on a plant may discard dynamics that seem insignificant in open loop, but are excited by the controller (that was of course designed without regard to these dynamics), sometimes to such an extent that the closed loop becomes unstable; this phenomenon is called *spill-over*.

Following the second road, a high order controller $C(s)$ is first being designed on the basis of the full order model, and a reduced-order controller is then obtained by some model reduction on $C(s)$; this class of methods will be labeled *controller reduction*. If one uses open loop model reduction for this, there is not much reason to suppose that the closed-loop system will be any more satisfactory than with the first 'road'. However, since both the controller and the plant are known at this stage, closed-loop considerations can be taken into account, i.e. one may try to minimize the effect of the controller reduction on the closed-loop system. Many authors have contributed such controller reduction methods, some of which will be sketched below. There is one fundamental disadvantage to this second class of methods, however : sometimes the order n of the plant is too high to make an optimal control design possible. Not only the number of computations needed often goes up with some multiple of n^3, but also, there is the more serious concern that the algorithms become numerically unreliable for very high orders, thereby barring the second road.

Roads 1 and 2 are labeled 'indirect' approaches, in contrast with the third road, labeled the 'direct' approach, which amounts to optimization of a (preferably smooth) closed-loop performance criterion, reflecting also concerns such as stability, robustness, etc., over the set of all possible controllers of order at most r. This approach will be referred to as *fixed order controller design.* The literature offers many algorithms of this type [163, 179, 44, 49, 258, 63, 308]; when looking for relevant contributions, it is good to keep in mind that fixed order control design can trivially be formulated as a *static output feedback* design problem [172, 156, 183, 184, 185, 186, 45, 46, 205, 206, 183, 268, 5, 245]. This road is conceptually the ideal one, since one directly optimizes what one is interested in. However, one is confronted with many difficulties : generally, the optimization problem is non-convex, and it is impossible to verify whether a candidate solution is a global minimum or not; the computational burden is often much higher, and the numerical problems certainly not less; closed-loop stability is sometimes hard to include in the minimization criterion; the choice of the parameterization of the set of fixed-order controllers is important but difficult. The bottom line is that these methods often require sophisticated and computationally expensive algorithms, without guaranteeing that better solutions can be reached in an acceptable time. Moreover, the flexibility and the vast body of knowledge offered by existing control design techniques and software are lost to the designer.

At this point the reader may feel that the diagram in Fig. 1.2 is rather like one of those medieval maps, where next to one road it is marked *'Here be dragons'*, another road is barred by a dangerous wild river, and a third road winds through the enchanted Transsylvanian mountains, that may prove to be a deadly labyrinth. Yet things are altogether not so gloomy : rather than three separate choices, the three roads are often complementary.

Where the order of the original plant is so high that one cannot reliably do the computations for the (optimal) control design technique one wishes to use, some plant reduction will be unavoidable. However, it is not necessary or advisable to reduce the plant immediately to the desired controller order; rather, one should choose some relatively high order, so that the reduction error is kept to an absolute minimum, yet such that the design problem becomes numerically feasible. One should remember that there is no advantage in reducing the order of the plant further than this level, since one will only lose accuracy, while gaining nothing but a little computation time. Keeping this in mind, one will very often find that the plant reduction is not the most critical step. One may also benefit from the existence of a stabilizing and more or less acceptable preliminary low-order controller, e.g. an available PID controller or one obtained using heuristics, to take into account closed-loop considerations as in the second road.

This reduced order plant of intermediate order is then used to design an optimal controller, which may then be reduced to the desired order with the methods of the second approach.

Presumably the resulting low-order controller will be more desirable than the preliminary (or absent) controller one started with; thus it may be helpful to replace this preliminary controller by the newly found one, and repeat the whole procedure, possibly more than once or until convergence. Whether such an iteration converges, and if so whether it converges to a minimum, depends of course on the actual design and reduction techniques chosen.

Optionally, the resulting low-order controller may be improved using a direct method; often, it will be close enough to the optimum, so that the direct method does not suffer from some of its pitfalls, and does not need as much sophistication as when starting without this initial guess.

So the traveler to the low order countries will often have to start out on the first road, there being no other choice; but will be well advised to leave it well before he meets any dragons, and cross the river at the first available ford, to reach the second road, which will hopefully take him to the foothills near his destination; and there he may then quietly stroll to the valley along the third road, on a stretch where the evil mountains no longer cast their spell.

The work presented in this thesis is mainly contributing to the ongoing labour of paving the second road. Frequency weighted model reduction methods are of special interest here. As was mentioned above, the controller doesn't need to be approximated equally well in the entire frequency range. As a rule of thumb, the error should be small around the critical or *cross-over* frequency, the frequency interval where the Nyquist plot nears the point -1, etc., but there are better ways to construct weighting systems than simply to design a bandpass filter that

stresses the critical interval(s) (see Section 2.4).
Of course, nothing prevents using the frequency weighted model reduction algorithms presented in this text for plant reduction, if some good weights of moderate order are available; but they are in general not quite as indispensable there.

Before proceeding to an overview of the work in this dissertation, we offer one more general remark on controller reduction, which it is important to keep in mind when reading this text. The controller reduction more often than not is very iterative in nature, not in the sense that it may make use of iterative algorithms to do one reduction, but that the reduction itself may have to be repeated several times over; this was hinted at already in this section, and further examples will pop up in this thesis. On top of this, the layer above controller reduction, i.e. controller design, is rarely a one-shot process, but may have to be redone many times. It is not uncommon, therefore, that for one control system design, tens or even hundreds of controller reduction are to be performed. As a consequence, savings in the operation count of a reduction technique gain importance accordingly.

1.3 Chapter by chapter overview

The contributions of this thesis fall in two quite different categories : balancing-type model reduction techniques, mainly to be found in Chapters 3 and 4, and H_2-optimal model reduction algorithms, treated in Chapters 5 and 6; Chapter 7 has a somewhat hybrid character, and contains both types of techniques.
Another classification may be done according to the type of weightings : unweighted, frequency weighted, and time weighted approximation methods.

These relationships between chapters and concepts are depicted in a graphical way by the almost-matrix-like diagram of Fig. 1.3 ; the direction of the graph arrows represent the logical or motivating precedence of these relationships.

This busy structure should not deter the reader; Fig. 1.4 shows in which order the chapters may be read.

Chapter 2 contains some preliminaries, a great part of which may be safely skipped by more knowledgeable readers; special attention is given to norms and a section is devoted to frequency weightings for controller reduction. This section also shows of how controller reduction may be useful in some D-K iteration-like control design schemes, and how frequency weight design may be an iterative process itself.

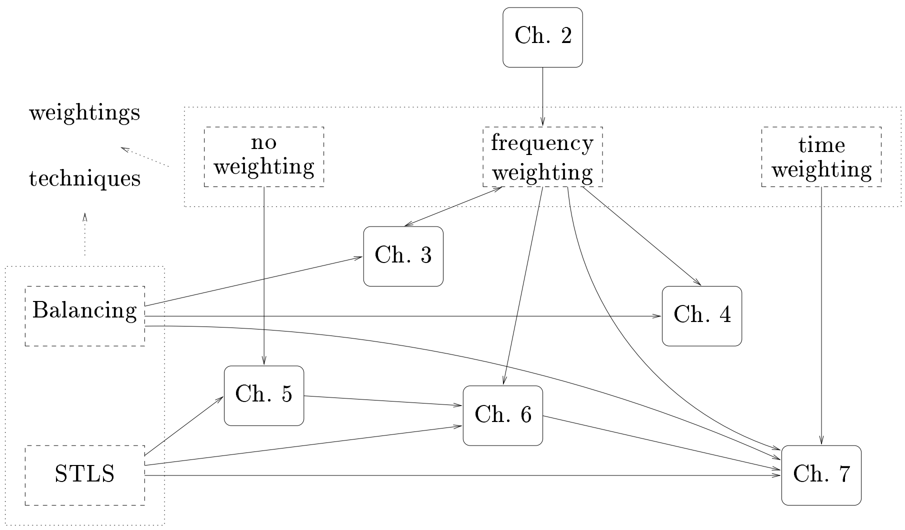


Figure 1.3: Connections between chapters, techniques and weighting methods.

The contributions of **Chapter 3** consist mainly in showing how the structure of specific weightings or controllers may be exploited to reduce the size of the computations. The popular Frequency Weighted Balanced Truncation (FWBT) [93] is applied to some of the frequency weights put forward in Chapter 2. It is shown that using FWBT while trying to keep the same closed-loop performance, one is lead in a natural way to an existing model reduction technique with a lower flop count, i.e. the recently proposed Closed-Loop Balanced Truncation (CLBT) [302]. For observer-based controllers, more savings are achieved, and another connection to an existing model reduction technique is revealed. Examples are included, to illustrate how these methods may be useful in iterative designs, e.g. for decentralized control or μ-synthesis.

In **Chapter 4**, another specific frequency weighted approximation problem is studied : relative error model reduction, where the system to be approximated is weighted with its own inverse. For this, a well-known balancing-type method is available : Balanced Stochastic Truncation [78]. Here, another balancing-type model reduction method is proposed that can equally be regarded as an instance of Frequency Weighted Balanced Truncation; again, the structure can be used to perform the computations more efficiently, yielding an algorithm quite competitive with Balanced Stochastic Truncation, and indeed very related to it, as for some special cases both methods are equivalent. Furthermore, it is shown how one algebraic Riccati equation is instrumental in the computations for both

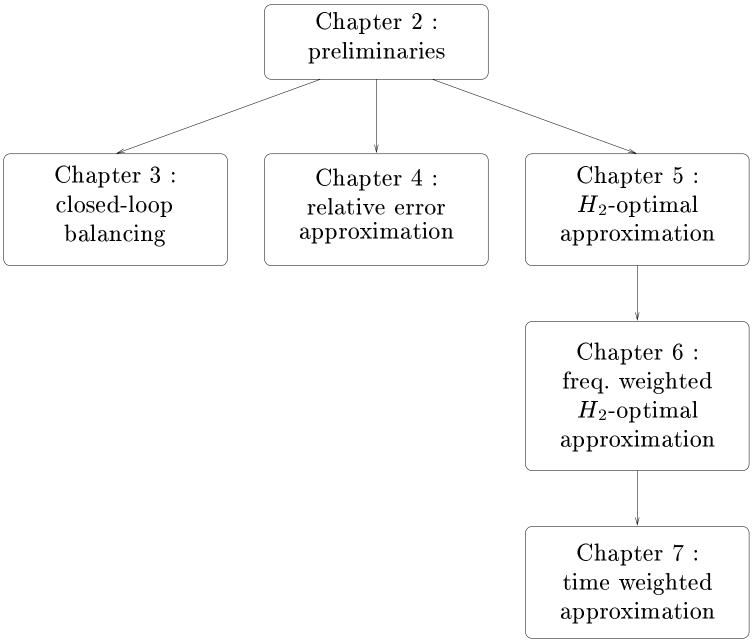


Figure 1.4: Readability flow chart of the thesis.

algorithms; thus a more efficient computational method for Balanced Stochastic Truncation is found.

In **Chapter 5**, the discrete-time H_2-optimal model reduction problem is treated, i.e. the approximation problem where the H_2-norm of the modeling error is minimized. Some properties of the optimal approximant are first listed. The problem is then cast in the Structured Total Least Squares framework recently proposed in [74]. This framework is useful for many other problems as well, but the generic iterative algorithms for its solution are not very efficient for this problem, as it is in fact cast as a limit case of Structured Total Least Squares problems of ever larger size. This chapter reveals the structure of the quantities figuring in this 'limit case algorithm'; thus a more efficient iterative algorithm is arrived at, tailored to this specific problem.

The algorithm proposed in Chapter 5 is plagued with the fact that an ill-conditioned linear system is to be solved each iteration; in **Chapter 6**, a new formulation is introduced, which greatly alleviates this problem. At the same time, the results from Chapter 5 are generalized to the *frequency weighted* H_2-

optimal model reduction problem. Some special but common cases of frequency weighting are discussed, and it is shown how continuous-time reduction problems may be solved with this method as well.

In **Chapter 7**, another generalization of H_2-optimal model reduction is treated, where the modeling error is weighted in the time domain. Again Structured Total Least Squares is shown to be a useful framework; however, only for a special case, i.e. Hilbert-Schmidt-Hankel model reduction, the limit case can be worked out, and an algorithm similar to the one of Chapter 6 is presented.
Finally, a heuristic balancing-type method is put forward incorporating time-domain weighting, that can serve as a provider of initial candidates of the latter algorithms.

Chapter 8 offers some conclusions and briefly discusses open questions and possibilities for future research. Furthermore, some of the proofs and derivations are gathered in an **appendix**, to enhance the readability of the text; the very last pages contain an extensive **bibliography** on controller reduction and frequency weighted model reduction.

All numerical evidence concerning the contributions of this thesis, presented in tables and plots, was produced by Matlab routines. This software is available upon request from the author, of course with the usual caveats and disclaimers with respect to possible imperfections or lack of documentation.

Chapter 2

Preliminaries

In this chapter, some notation and fundamental concepts that will be used throughout the sequel, are collected. All of this material, except maybe a couple of facts on norms in Subsection 2.2.7, and the treatment of frequency weightings for controller reduction in Section 2.4, is standard and can be found in most textbooks on modern linear control theory, and is added here only to enhance the readability of the thesis, so the treatment given here will be very brief.

General notation

The notation $A \stackrel{\triangle}{=} B$ generally means that A is defined by the expression B, but sometimes it will be clear from the context that B or the parts that make up B are defined by the left hand side.

The abbreviation *iff* means 'if and only if' and may also be denoted by the symbol $\iff$.

2.1 Linear algebra

The set of real numbers is denoted as $\mathbb{R}$, $\mathbb{C}$ denotes the set of complex numbers. The imaginary number $\sqrt{-1}$ is represented by j; it will be clear from the context when this symbol is reused in a different meaning. The complex conjugate of a number $c \in \mathbb{C}$ is written as $\bar{c}$. The set of $m \times n$ real (complex) matrices is denoted as $\mathbb{R}^{m\times n}$ ($\mathbb{C}^{m\times n}$).
The element on the ith row and the jth column of a matrix $M \in \mathbb{C}^{m\times n}$ may be written as M_{ij} or as $[M]_{(i,j)}$. $M_{:j}$ or $[M]_{(:,j)}$ is the jth column of M, $M_{i:}$ or $[M]_{(i,:)}$ is the ith row.

The *Kronecker product* $A \otimes B$ where A and B are matrices, $A \in \mathbb{C}^{m\times n}$, is the matrix

$$A \otimes B \triangleq \begin{bmatrix} A_{11}B & A_{12}B & \cdots & A_{1n}B \\ A_{21}B & A_{22}B & \cdots & A_{2n}B \\ \vdots & \vdots & \vdots & \vdots \\ A_{m1}B & A_{m2}B & \cdots & A_{mn}B \end{bmatrix}.$$

The *transpose* of a matrix $M \in \mathbb{C}^{m\times n}$, denoted as M^T, is a matrix defined by $[M^T]_{(i,j)} \triangleq M_{ji}$; the *complex conjugate transpose* of a matrix $M \in \mathbb{C}^{m\times n}$, denoted as M^*, is defined by $[M^*]_{(i,j)} \triangleq \bar{M}_{ji}$.
A matrix $M \in \mathbb{R}^{n\times n}$ is *symmetric* iff $M^T = M$; a matrix $M \in \mathbb{C}^{n\times n}$ is *Hermitian* iff $M^* = M$.

The unit matrix of size $n \times n$ is denoted by I_n and often I where its dimension is clear from the context; likewise, the zero matrix of size $m \times n$ is denoted by $0_{m\times n}$ and often just 0.

A matrix $U \in \mathbb{R}^{n\times n}$ is *orthogonal* iff $U^TU = UU^T = I$; $U \in \mathbb{C}^{n\times n}$ is *unitary* iff $U^*U = UU^* = I$.

The *inverse* of a matrix $M \in \mathbb{R}^{n\times n}$, denoted M^{-1}, is a matrix such that $MM^{-1} = M^{-1}M = I$; if it exists, it is unique.
A *right inverse* of a matrix $M \in \mathbb{R}^{n\times m}$, denoted $M^\dagger$, is a matrix such that $MM^\dagger = I$; if $M^\dagger$ exists, $n \leq m$; if $RankM = n \leq m$, then $M^\dagger$ exist. Analogously, a *left inverse* of a matrix $M \in \mathbb{R}^{n\times m}$, also denoted $M^\dagger$, is such that $M^\dagger M = I$, and exists only if $n \geq m$; again, $n \geq m = RankM$ is sufficient for existence. A right or left inverse of a square regular matrix is the inverse of that matrix; hence, the use of the same notation for left and right inverse causes no ambiguity.

If $M_i \in \mathbb{C}^{m_i\times n_i}$, $i = 1, \ldots, N$, then

$$\operatorname{diag}(M_1, M_2, \ldots, M_N) \triangleq \begin{bmatrix} M_1 & 0 & & \cdots & 0 \\ 0 & M_2 & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & & 0 & M_N \end{bmatrix}.$$

A matrix $M \in \mathbb{C}^{n\times n}$ is *positive definite* iff

$$\forall\, x \in \mathbb{C}^{n\times 1}\backslash\{0\} : x^* M x > 0\ .$$

A matrix $M \in \mathbb{C}^{n\times n}$ is *positive semidefinite* iff

$$\forall\, x \in \mathbb{C}^{n\times 1} : x^* M x \geq 0\ .$$

$\lambda \in \mathbb{C}$ is an *eigenvalue* of a matrix $M \in \mathbb{C}^{n\times n}$ if $Mx = \lambda x$ for some $x \in \mathbb{C}^{n\times 1}$.

For every matrix $M \in \mathbb{C}^{n\times n}$, there is a nonsingular matrix $T_J \in \mathbb{C}^{n\times n}$ and a matrix $J = \mathrm{diag}(J_1, J_2, \ldots, J_t) \in \mathbb{C}^{n\times n}$, where the matrices J_i have the form

$$J_i = \begin{bmatrix} \lambda_i & 1 & & \cdots & 0 \\ 0 & \lambda_i & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & & 0 & \lambda_i \end{bmatrix},$$

such that $M = T_J^{-1} J T_J$ [116]. The numbers λ_i are easily seen to be eigenvalues of M. For a proof of this statement see [128]. The J_i are called Jordan blocks. Matrices for which the Jordan blocks J_i are all 1×1 blocks are called *non-defective* or *diagonalizable*; in this case $T_J^{-1} J T_J$ is an *eigenvalue decomposition* of M. Matrices that are not diagonalizable are called *defective*, and the eigenvalue λ_i associated with any Jordan block of size greater than one is labeled a *defective eigenvalue*. Defectiveness of matrices is non-generic [298] i.e. a randomly small perturbation may turn any defective matrix into a non-defective one, whereas the set of non-defective matrices is an open set. A matrix $M \in \mathbb{C}^{n\times n}$ is *simple* if it has n distinct eigenvalues; simple matrices are again generic in the set of diagonalizable matrices.

For every $M \in \mathbb{C}^{n\times n}$, there is a unitary $Q \in \mathbb{C}^{n\times n}$ and an upper triangular $S \in \mathbb{C}^{n\times n}$ such that $M = QSQ^*$; this is called the *complex Schur decomposition*. The diagonal elements S_{ii} of the *complex Schur matrix* S are easily seen to be the eigenvalues of the matrix M.
For every $M \in \mathbb{R}^{n\times n}$, there is an orthogonal $Q \in \mathbb{R}^{n\times n}$ and a matrix $S \in \mathbb{R}^{n\times n}$,

$$S = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1m} \\ 0 & S_{22} & \cdots & S_{2m} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & S_{mm} \end{bmatrix},$$

where the S_{ii} are either 1×1 matrices or 2×2 matrices with complex conjugate eigenvalues, such that $M = QSQ^T$; this is called the *real Schur decomposition*. Matrices like S in the real Schur decomposition are called *upper quasi-triangular*.

The Schur decomposition is computed using (an improved version of) the famous QR algorithm, which requires approximately $25n^3$ operations if both the matrices S and Q are computed [116].

The Schur decomposition turns out to be of paramount importance for reliable computations in modern control theory, including controller reduction. The reason for this is that it is computed using orthogonal (or unitary) transformations only, which makes it numerically dependable. It is also the first and numerically most reliable step in the computation of an eigenvalue decomposition; this is then followed by a diagonalization procedure of the Schur factor S, which has a lower operation count, but may be very unreliable.

The *trace* of a matrix M, denoted as Tr M, is the sum of its diagonal elements. The *Frobenius norm* of M is $\|M\|_F \triangleq \sqrt{\mathrm{Tr}\ (M^T M)} = \sqrt{\mathrm{Tr}\ (M M^T)}$.

2.2 System theory

In the most broad sense, a system can be defined without distinguishing between its inputs and its outputs [293]; this is often useful e.g. in circuit theory, but in a control design context, the distinction between inputs and outputs is always clear. Determining which control inputs are used, which parameters can be measured, and which one should be affected by the control, etc. is usually not the control designer's discretion. This is why, in this thesis, a system is considered as an operator mapping input signals, represented by a time-dependent vector $u(t)$ (u_k), to output signals $y(t)$ (y_k).

We shall restrict ourselves in this thesis to linear systems which are *finite-dimensional* or *lumped*, i.e. have a finite state vector. While the most accurate models of physical phenomena are most often infinite-dimensional, at some stage these models are generally approximated by a finite-dimensional one, for practical purposes. Furthermore we shall only consider time-invariant systems.

2.2.1 Realizations

A continuous-time (CT) linear system is described by linear differential state space equations and linear output equations

$$\dot{x}(t) = Ax(t) + Bu(t) \ , \tag{2.1}$$

$$y(t) \;=\; Cx(t) + Du(t) \;, \tag{2.2}$$

where $A \in \mathrm{I\!R}^{n\times n}$, $B \in \mathrm{I\!R}^{n\times n_u}$, $C \in \mathrm{I\!R}^{n_y\times n}$, $D \in \mathrm{I\!R}^{n_y\times n_u}$.
The vector x is referred to as the *state vector* or just *state*; with some abuse of terminology, the i-th component of the state vector x_i is sometimes called the *i-th state*. The future state $x(t)$ and output $y(t)$, $t > t_0$, are uniquely determined if the initial state $x(t_0) = x_0$ and future input $u(t)$, $t > t_0$, are specified. The transfer matrix $G(s) = C(sI - A)^{-1}B + D$ maps the Laplace transform $u(s)$ of $u(t)$ onto the Laplace transform $y(s) = G(s)u(s)$ of $y(t)$, if $x_0 = 0$.

A discrete-time (DT) system is described by difference equations

$$x(k+1) \;=\; Ax(k) + Bu(k) \;, \tag{2.3}$$
$$y(k) \;=\; Cx(k) + Du(k) \;. \tag{2.4}$$

Again, given the initial state and future input the future state and output is uniquely determined. The transfer matrix $G(z) = C(zI - A)^{-1}B + D$ maps the z-transform $u(z)$ of $u(k)$ onto the z-transform $y(z) = G(z)u(z)$ of $y(k)$, if $x_0 = 0$.

Though equations (2.1) (2.2) uniquely determine the system above, they do not give the only state space description or state space *realization* of the system; substitution $x = T\bar{x}$, where T is nonsingular square matrix, obviously yields another realization

$$\begin{aligned} \dot{\bar{x}} &= T^{-1}AT\bar{x} + T^{-1}Bu \;, \\ y &= TCx + Du \;. \end{aligned}$$

We say this realization was obtained from (2.1) (2.2) by a *similarity transformation* T. The *order* of the realization (2.1) (2.2) is n (recall $A \in \mathrm{I\!R}^{n\times n}$).
We adopt the notation $\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ or $[A, B, C, D]$ for the realization(2.1) (2.2), but often we shall slightly abuse this notation by designating the underlying system by it. The same remarks and notation apply to discrete-time systems.

Note that only real systems, i.e. systems which can be represented by a realization with real matrices A, B, C and D, are dealt with, such that real inputs u yield real outputs y. This point may seem obvious, yet one occasionally encounters model reduction methods in the literature yielding complex reduced-order models, which in our opinion makes these methods quite useless, at least within the context of control theory.

2.2.2 Stability

The *poles* of a system $G = [A, B, C, D]$ are the eigenvalues of A. A continuous-time system will be called *stable* iff all its poles have strictly negative real parts[1]; a discrete-time system will be *stable* iff all its poles have an absolute value strictly smaller than one.

2.2.3 Gramians

The *controllability gramian* P and the *observability gramian* Q of a realization $[A, B, C, D]$ of a stable continuous-time system are defined as

$$P \triangleq \int_0^\infty e^{At} BB^T e^{A^T t} dt \qquad \text{and} \qquad Q \triangleq \int_0^\infty e^{A^T t} C^T C e^{At} dt.$$

These matrices satisfy the well-known Lyapunov equations

$$AP + PA^T + BB^T = 0 \qquad \text{and} \qquad A^T Q + QA + C^T C = 0.$$

For a stable discrete-time system one has

$$P \triangleq \sum_{k=0}^{\infty} A^k BB^T (A^T)^k \qquad \text{and} \qquad Q \triangleq \sum_{k=0}^{\infty} (A^T)^k C^T C A^k,$$

and the discrete-time Lyapunov equations become :

$$APA^T - P + BB^T = 0 \qquad \text{and} \qquad A^T QA - Q + C^T C = 0.$$

Note that the gramians P and Q are not system invariant : a similarity transformation T yields a realization with controllability gramian TPT^T and observability gramian $T^{-T} Q T^{-1}$. The gramians are always positive semidefinite.

Parseval's identity [230] yields another formula for the gramians :

$$\begin{aligned} P &= \frac{1}{2\pi} \int_{-\infty}^{\infty} (j\omega I - A)^{-1} BB^T (-j\omega I - A^T)^{-1} d\omega \\ Q &= \frac{1}{2\pi} \int_{-\infty}^{\infty} (-j\omega I - A^T)^{-1} C^T C (j\omega I - A)^{-1} d\omega \ . \end{aligned}$$

The gramians have various interpretations [50, 107, 195, 216], we mention only a few. The controllability gramian P may be interpreted as the expected value

[1] This definition is usually labeled *asymptotically* or *strictly* stable, but since we are not concerned here with the distinction from other kinds of stability, we drop the epithet.

of $x(t)x(t)^T$, if the system has been driven with white noise of unit variance up to time t. Indeed,

$$\begin{aligned}
E(x(t)x(t)^T) &= E(\int_{-\infty}^{t}\int_{-\infty}^{t} e^{A(t-\tau_1)}Bu(\tau_1)u^T(\tau_2)B^Te^{A^T(t-\tau_2)}d\tau_1 d\tau_2) \\
&= \int_{-\infty}^{t}\int_{-\infty}^{t} e^{A(t-\tau_1)}BE(u(\tau_1)u^T(\tau_2))B^Te^{A^T(t-\tau_2)}d\tau_1 d\tau_2 \\
&= \int_{-\infty}^{t}\int_{-\infty}^{t} e^{A(t-\tau_1)}B\delta(\tau_1-\tau_2)B^Te^{A^T(t-\tau_2)}d\tau_1 d\tau_2 \\
&= \int_{-\infty}^{t} e^{A(t-\tau)}BB^Te^{A^T(t-\tau)}d\tau \\
&= \int_{0}^{\infty} e^{A\tau}BB^Te^{A^T\tau}d\tau = P \ .
\end{aligned}$$

Another well-known interpretation is that the minimal input energy required to reach $x(0) = x_0$ is $\int_{-\infty}^{0} u^T u dt = x_0^T P^{-1} x_0$. The observability gramian Q gives the relation between the initial state x_0 and the energy in the transient response $y(t) = Ce^{At}x_0$:

$$\int_0^\infty y^T y dt = \int_0^\infty x_0^T e^{A^T t} C^T C e^{At} x_0 dt = x_0^T Q x_0 \ .$$

For discrete-time systems, these interpretations still hold, and can be derived in an analogous way.

2.2.4 System properties

A system $G(s) = [A, B, C, D]$ is *strictly proper* if $D = 0$; the *strictly proper part* of $G(s)$ is $[A, B, C, 0]$.
$G(s)$ is *square* if $n_u = n_y$; a square system $[A, B, C, D]$ is *causal invertible* if D is invertible.

The *inverse* of a causal invertible $n \times n$ system G, denoted G^{-1}, is a system such that $GG^{-1} = G^{-1}G = I$; it is easily found that if $G = [A, B, C, D]$, then

$$G^{-1} = \left[\begin{array}{c|c} A - BD^{-1}C & BD^{-1} \\ \hline -D^{-1}C & D^{-1} \end{array}\right] \ .$$

Note that non-proper inverses may be defined for systems which are not causal invertible, but these are not considered here. By analogy, a *right inverse* of an $n \times m$ system G, denoted $G^\dagger$, is a system such that $GG^\dagger = I$; again if $G^\dagger$ exists, $n \leq m$. Again, we define G to be right (causal) invertible if and only if D has a right inverse. The definition for left inverses is analogous, and the notation is reused as in the matrix case, as again there can be no ambiguity.

For a CT system $G(s) \triangleq [A, B, C, 0]$, $g(t) \triangleq Ce^{At}B$ is the *impulse response matrix*; the output y is the sum of the transient response and the convolution of the impulse response matrix with the input, i.e. $y(t) = Ce^{At}x_0 + g(t) \star u(t)$ where the convolution is defined as $g(t) \star u(t) \triangleq \int_0^t g(t-\tau)u(\tau)d\tau$. For a DT system, the *impulse response matrix* is $G(k) \triangleq CA^kB + D\delta(k)$, where $\delta(k) = 1$ for $k = 1$, and zero elsewhere; an expression for the output is $y(k) = CA^kx_0 + G(k) \star u(k)$ where $G(k) \star u(k) \triangleq \sum_{l=0}^{k} G(k-l)u(l)$.

A quantity defined by operations on the system matrices is a *system invariant* if its value does not depend on the realization; the poles and the transfer matrix $G(s)$ are examples of such system invariants. Since the transfer matrix uniquely determines the system, we shall often regard it as the system itself.

The *transmission zeroes* (often simply called *zeroes*) are the complex numbers where $G(s)$ drops rank. Zeroes are system invariants.
A system is *minimum phase* if its zeroes have strictly negative real parts for continuous-time systems, or if its zeroes have an absolute value strictly smaller than one for discrete-time systems. A system is *inverse stable* if it is causal invertible and minimum phase. A finite-dimensional system is *right outer* (*left outer*) if it has full row (column) rank in the open right half complex plane for continuous-time systems, or outside the closed unit disk for discrete-time systems.

A square system $G(s)$ is *all-pass* if $G^T(-s)G(s) = G(s)G^T(-s) = I$. A system $G(s)$ is *right inner* if it is stable and $G^T(-s)G(s) = I$ (*left inner* if it is stable and $G(s)G^T(-s) = I$). Any full rank stable $G(s)G^T(-s) = I$). Any full rank stable $n \times m$ system $G(s)$ can be factored as $G_i(s)G_o(s)$, where $G_i(s)$ is right inner and $G_o(s)$ is right outer; if $n \geq m$, $G_o(s)$ is square $(m \times m)$, if $m \geq n$, $G_i(s)$ is square $(n \times n)$ [286]. If $G(s)$ is finite-dimensional, it has a finite-dimensional inner-outer factorization. Evidently one may formulate a dual version of this proposition with left inner and left outer factors.

The transfer matrix may be expanded into a power series :

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \sum_{i=0}^{\infty} G_i s^{-i}$$

for a continuous-time system, and

$$G(z) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \sum_{i=0}^{\infty} G_i z^{-i}$$

for a discrete-time system, where $G_0 \triangleq D$ and $G_i \triangleq CA^{i-1}B$. The matrix G_i is coined the i-th *Markov parameter*[2]; Markov parameters are system invariants. Note that in the discrete-time case, the impulse response matrix is the series of Markov parameters.

For a strictly proper stable continuous-time system $G(s) = C(sI - A)^{-1}B$, the *power spectrum* is defined as $G(s)G^T(-s)$. It is easily shown that $G(s)G^T(-s) = R(s) + R^T(-s)$, where $R(s) \triangleq C(sI - A)^{-1}PC^T$. The *output covariance parameters* are the Markov parameters R_i of the causal part $R(s)$ of the power spectrum, i.e. $R_i \triangleq CA^{i-1}PC^T$; the power spectrum may be expressed as $G(s)G^T(-s) = \sum_{i=1}^{\infty} R_i s^{-i} + R_i^T(-s)^{-i}$.
For discrete-time systems, the output covariance parameters may be defined by the same expression $R_i \triangleq CA^{i-1}PC^T$.
A number of model reduction algorithms focus on matching some of the Markov and output covariance parameters (see Subsection 2.3.3).

A realization $G = [A, B, C, D]$ of order n is *controllable* if the *controllability matrix* $\left[\begin{array}{cccc} B & AB & A^2B & \cdots A^{n-1}B \end{array}\right]$ is full rank and *observable* if G^T is controllable. If a realization is both controllable and observable, it is *minimal*. For an uncontrollable realization, a left eigenvector of the A-matrix may be found which is orthogonal to the range of the B matrix; the corresponding eigenvalue is called an uncontrollable pole. For an unobservable realization, a right eigenvector of the A-matrix may be found which is in the null space of the C matrix; the corresponding eigenvalue is called an unobservable pole.
For a non-minimal realization, a realization of lower order may be found that describes the same system; for a minimal system this is not possible. The order of a minimal realization of a system is its *McMillan degree*. By similarity transformation of a minimal realization one may find all possible realizations of the same order. The set of finite-dimensional linear CT systems will be denoted by $\mathcal{M}$. The set of CT systems with McMillan degree at most r is written as $\mathcal{M}_r$; this can also be regarded as the set of systems that can be represented by a realization of order r. The set of stable systems with McMillan degree at most r is denoted by $\mathcal{S}_r$.

[2]Some authors define the i-th Markov parameter as CA^iB; here the definition of [153] is adopted.

A system is *stabilizable* if its uncontrollable poles are stable. A system is *detectable* if its unobservable poles are stable.

2.2.5 Bilinear transform

The *bilinear transformation* $\mathcal{B}_T$ maps a CT system $G(s)$ on a DT system

$$G_d(z) = \mathcal{B}_T(G(s)) \stackrel{\triangle}{=} G(\frac{1-\frac{Ts}{2}}{1+\frac{Ts}{2}}) \, ,$$

where T may be interpreted as a sampling time. The inverse of the bilinear transformation, $\mathcal{B}_T^{-1}$, maps a DT system $G(z)$ on a CT system $G_c(s) = \mathcal{B}_T^{-1}(G(z)) \stackrel{\triangle}{=} G(\frac{2}{T}\frac{z-1}{z+1})$. When T is not mentioned it will be assumed to equal 2; the corresponding bilinear transformation is denoted $\mathcal{B}$.
The bilinear transformation preserves stability. If $G(s) = [A, B, C, D]$, then

$$\mathcal{B}_T(G(s)) = \left[\begin{array}{c|c} (I - A\frac{T}{2})^{-1}(I + A\frac{T}{2}) & (I - A\frac{T}{2})^{-1}B\sqrt{T} \\ \hline \sqrt{T}C(I - A\frac{T}{2})^{-1} & D - C(2I - AT)^{-1}BT \end{array}\right].$$

When $T = 2$, it is readily verified that these realizations have identical controllability and observability gramians; for other values of T, the controllability gramian P changes into $2T^{-1}P$ and the observability gramian Q into $\frac{1}{2}TQ$. Hence, the product PQ is invariant under bilinear transformation.

2.2.6 Special realizations

Some realizations of a system are of special interest. Observer, controller, observability and controllability form are well known for SISO (single input single output) systems, but rather cumbersome to generalize to the MIMO (multiple input multiple output) case [153]; moreover, these forms suffer from numerical conditioning problems, which make them unsuitable for representation of systems of anything but low orders, such that they are rather irrelevant in a model reduction context. Other forms, of which a few are briefly mentioned below, are more suited.

It was mentioned above that the controllability and observability gramians are not system invariant, but PQ will be transformed into $TPQT^{-1}$ by a similarity transformation T, so the eigenvalues of PQ are system invariant. As both P and Q are positive definite, these eigenvalues are nonnegative. For minimal realizations, it is possible to find a so-called *balancing* similarity transformation T_B such that $T_B P T_B^T = T_B^{-T} Q T_B^{-1} = \Sigma$, where $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$, $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$ (for minimal systems $\sigma_n > 0$). The numbers σ_i, are called the *Hankel singular values*, and give a precise measure of how observable and controllable the corresponding states are, in view of the interpretations of the gramians given above [195]; Hankel singular values are invariant under bilinear transformation. $\left[T_B^{-1}AT_B, T_B^{-1}B, CT_B, D\right]$ is labeled a *balanced* realization. This realization has advantageous properties with respect to roundoff errors [197], but is mainly important in model reduction [195], as will be seen in a later chapter.

It is a misconception that balancing the gramians is always a well-conditioned operation : clearly, balancing is in general impossible for non-minimal realizations, and is generally ill-conditioned for realizations close to non-minimality.

As there is always a nonsingular $T_J \in \mathbb{C}^{n\times n}$ such that $A = T_J^{-1}JT_J$, where J is in Jordan form, every system with realization $[A, B, C, D]$ can also be represented by $\left[J, T_J^{-1}B, CT_J, D\right]$. Most often, the eigenvalues of A are non-defective, and J will actually be diagonal, and have the system's poles on the diagonal; we say we have a *modal* realization :

$$G(s) = \left[\begin{array}{cccc|c} \lambda_1 & 0 & \cdots & 0 & B_1 \\ 0 & \lambda_2 & \ddots & \vdots & B_2 \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & \lambda_n & B_n \\ \hline C_1 & C_2 & \cdots & C_n & D \end{array}\right] = \sum_{i=1}^{n} \frac{C_iB_i}{s-\lambda_i} + D \; .$$

The matrices C_iB_i are called the *residues*. It is often advantageous to restrict oneself to real A, B, C, D matrices; a realization where A is in real quasi-diagonal form will also be termed a modal realization.

Other realizations of particular interest are those where the A-matrix is put in Schur form : let $A = USU^T$ be a real Schur decomposition, then $\left[S, U^TB, CU, D\right]$ is a real Schur form realization of $G(s)$. Since U is an orthogonal matrix, putting a system into Schur form is a numerically very reliable operation. Schur form realizations make the poles of the system apparent, but not the residues. Several computations are facilitated by the Schur form of the A-matrix, e.g. computing the gramians, system inversion, frequency response computation.

2.2.7 Norms

The properties of a norm $\| \ \|$ on any vector space S over a field such as the real or complex numbers are well known :

1. $\forall \ x \in S : \|x\| \geq 0$

2. $\forall\, x \in S : \|x\| = 0 \iff x = 0$
3. $\forall\, x \in S, \forall\, a (\in \mathbb{R} \text{ or } \in \mathbb{C}) : \|ax\| = |a|\|x\|$
4. $\forall\, x, y \in S : \|x + y\| \leq \|x\| + \|y\|$

We are concerned here with norms on signals $s(t)$ and on systems $G(s)$.

2-norm

The *2-norm* of an $n \times 1$ vector signal $s(t)$ is defined as

$$\|s(t)\|_2 \triangleq \left(\int_{-\infty}^{\infty} s^*(t)s(t)dt \right)^{\frac{1}{2}} .$$

H_2 norm

The H_2 norm of a stable continuous-time system $G(s) = [A, B, C, 0]$ is defined by the relation :

$$\|G(s)\|_2 \triangleq \left(\int_{0}^{\infty} \text{Tr } (g(t)^T g(t))dt \right)^{\frac{1}{2}} .$$

where $g(t) \triangleq Ce^{At}B$ is the impulse response matrix. Note that continuous-time systems need to be strictly proper for the definition to make sense. Parseval's identity [230] yields an alternative definition :

$$\|G(s)\|_2 \triangleq \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{Tr } (G(j\omega)^* G(j\omega))d\omega \right)^{\frac{1}{2}}$$

This latter expression may be evaluated for unstable systems as well; for such systems we shall call this the L_2 *norm* $\|G(s)\|_{L_2}$.

For a discrete-time system $G(z) = [A, B, C, D]$,

$$\|G(z)\|_2^2 \triangleq \sum_{k=0}^{\infty} \text{Tr } (G_k^T G_k)dt$$

where G_k are the Markov parameters; here an alternative definition is

$$\|G(z)\|_2^2 \triangleq \frac{1}{2\pi j} \oint_T z^{-1} \text{Tr } (G(z^{-1})^T G(z))dz,$$

where the integral is taken over the unit circle T in the complex plane.
It is easily seen that $\|G\|_2^2 = \text{Tr } (B^T Q B + D^T D) = \text{Tr } (CPC^T + DD^T)$.

Note that the H_2 norm is not preserved after bilinear transformation.

For continuous-time (possibly matrix-valued) signals $x(t)$ define :

$$\begin{aligned}
\mathcal{L}_2 &\triangleq \{x(t) \mid \|x(t)\|_2 < \infty\} \\
\mathcal{L}_2^- &\triangleq \{x(t) \in \mathcal{L}_2 \mid x(t) = 0 \quad \forall\, t > 0\} \\
\mathcal{L}_2^+ &\triangleq \{x(t) \in \mathcal{L}_2 \mid x(t) = 0 \quad \forall\, t \leq 0\}
\end{aligned}$$

$$\begin{aligned}
\forall\ x(t) \in \mathcal{L}_2 : [x(t)]_+ &\triangleq x(t) \qquad t > 0 \\
&\triangleq 0 \qquad\quad t \leq 0
\end{aligned}$$

and for discrete-time (possibly matrix-valued) signals $x(z)$

$$\begin{aligned}
L_2 &\triangleq \{x(z) = \sum_{i=-\infty}^{\infty} x_i z^{-i} \mid \|x(z)\|_2 < \infty\} \\
L_2^- &\triangleq \{x(z) \in L_2 \mid x_i = 0 \quad \forall\, i > 0\} \\
L_2^+ &\triangleq \{x(z) \in L_2 \mid x_i = 0 \quad \forall\, i \leq 0\}
\end{aligned}$$

$$\forall\, x(z) = \sum_{i=-\infty}^{\infty} x_i z^{-i} : [x(z)]_+ \triangleq \sum_{i=1}^{\infty} x_i z^{-i}$$

HSH norm

The *Hilbert-Schmidt-Hankel* norm (or short the HSH norm) of a linear system can be defined, both in the continuous-time and the discrete-time case, as the Frobenius norm of the Hankel operator associated with this linear system, i.e. the operator mapping past inputs to future outputs.
For a discrete-time system $G(z) = D + C(zI - A)^{-1}B = \sum_{i=0}^{\infty} G_i z^{-i}$ the Hankel operator is the infinite-size Hankel matrix

$$\mathcal{H} \triangleq \begin{bmatrix} G_1 & G_2 & G_3 & \cdots \\ G_2 & G_3 & G_4 & \cdots \\ G_3 & G_4 & G_5 & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix} = \mathcal{O}\mathcal{C}\ ,$$

where $\mathcal{O}$ is the (infinite) observability matrix $\begin{bmatrix} C^T & A^T C^T & A^{2T} C^T & \cdots \end{bmatrix}^T$ and $\mathcal{C}$ is the controllability matrix $\begin{bmatrix} B & AB & A^2 B & \cdots \end{bmatrix}$. Thus the HSH norm is defined as $\|G(z)\|_{HSH} \triangleq \|\mathcal{H}\|_{\mathcal{F}} = \sqrt{\text{Tr }(\mathcal{H}\mathcal{H}^*)} = \sqrt{\text{Tr }(\mathcal{C}\mathcal{C}^*\mathcal{O}^*\mathcal{O})}$ and hence $\|G(z)\|_{HSH} = \sqrt{\text{Tr } PQ}$, where P and Q are the controllability and observability gramian. Furthermore it is also easily seen that

$$\text{Tr }(\mathcal{H}\mathcal{H}^T) = \sum_{i=1}^{\infty}\sum_{j=1}^{\infty} \text{Tr }(G_i G_j^T) = \sum_{i=0}^{\infty} i \text{Tr }(G_i^T G_i) \ .$$

For continuous-time systems, consider the linear system $G(s) = [A, B, C, 0]$ with impulse response matrix $g(t) = Ce^{At}B$. Then the Hankel operator is the integral operator $\mathcal{H}$ with kernel $k(t,s) = g(t+s)$, i.e. the operator mapping past inputs $u(t)$ (with $u(t) = 0$ for $t > 0$), to future outputs $y(t) = \int_{-\infty}^{0} g(t-s)u(s)ds = \int_0^{\infty} g(t+s)u(-s)ds$. Thus, the HSH norm is defined in this case as

$$\|G(s)\|_{HSH}^2 \triangleq \int_0^{\infty}\int_0^{\infty} \text{Tr }\left(g^T(t+s)g(t+s)\right) dtds \ .$$

Transforming to new independent variables $s' \triangleq t - s$ and $t' \triangleq t + s$,

$$\|G(s)\|_{HSH}^2 = \frac{1}{2}\int_0^{\infty}\int_{-t'}^{t'} \text{Tr } g^T(t')g(t')ds'dt' = \int_0^{\infty} t\text{Tr } g^T(t)g(t)dt \ .$$

It is also possible to rewrite the defining equation as

$$\begin{aligned} \|G(s)\|_{HSH}^2 &= \int_0^{\infty}\int_0^{\infty} \text{Tr }(B^T e^{A^T t} e^{A^T s} C^T C e^{At} e^{As} B) dtds \\ &= \text{Tr }\left(\int_0^{\infty} e^{As}BB^T e^{A^T s} ds \int_0^{\infty} e^{A^T t} C^T C e^{At} dt\right) \\ &= \text{Tr } PQ \ . \end{aligned}$$

Applying $\|G(s)\|_{HSH}^2 = \text{Tr } PQ$ to a balanced realization yields the expression $\|G(s)\|_{HSH}^2 = \sum_{i=1}^{n} \sigma_i^2$, where σ_i are the Hankel singular values of the system $G(s)$ (or $G(z)$).

A scaling of the frequency axis $s \to as$ does not affect the Hankel singular values : $P \to a^{-1}P$, $Q \to aQ$, hence, the HSH norm is unchanged as well.

Furthermore, the fact that the HSH norm depends solely on the gramians P and Q has a nice consequence : the HSH norm is preserved under the bilinear transform $\mathcal{B}$, since this mapping does not affect the gramians. In combination with the invariance under frequency scaling, this also means that the HSH norm is unaltered by the more general bilinear transform $\mathcal{B}_T$, where s is replaced by $\frac{2}{T}\frac{z-1}{z+1}$. A number of interesting properties of the HSH norm is given in [130].

H_∞ norm

A prominent tool in modern control theory is the H_∞ norm : this is defined as

$$\|G(s)\|_\infty \triangleq \sup_{\Re s > 0} \bar{\sigma}(G(s))$$

$$\|G(z)\|_\infty \triangleq \sup_{|z| > 1} \bar{\sigma}(G(z)) \ .$$

The H_∞ norm only makes sense for stable systems; for unstable systems one can define the L_∞ norm

$$\|G(s)\|_{L_\infty} \triangleq \sup_{\omega \in \mathbb{R}} \bar{\sigma}(G(j\omega))$$

$$\|G(z)\|_{L_\infty} \triangleq \sup_{\omega \in \mathbb{R}} \bar{\sigma}(G(e^{j\omega})) \ .$$

As for stable systems the L_∞ norm is identical to the H_∞ norm, and for unstable systems the H_∞ norm is undefined, one can use the notation $\|G(s)\|_\infty$ for the L_∞ norm without introducing any ambiguity.

The H_∞ norm (as well as the L_∞ norm) is *sub-multiplicative*, i.e. $\|G_1 G_2\|_\infty \leq \|G_1\|_\infty \|G_2\|_\infty$, and can be interpreted as an RMS gain :

$$\|G(s)\|_\infty = \sup_{\|u(t)\|_2 \leq 1} \|G \star u(t)\|_2 \ .$$

The H_∞ norm is readily seen to be unaffected by frequency scaling and bilinear transformation.
The computation of the H_∞ norm involves an iteration [55], and is rather expensive : several tens of n^3 flops.

Hankel norm

The *Hankel norm* $\|G\|_H$ is the largest Hankel singular value of a system.

Energy gain interpretations

Most popular norms encountered in control and system theory have an interpretation in terms of energy gain. It is well known that the Hankel norm squared has an interpretation as the maximal energy gain from past inputs to future outputs [107], i.e.

$$\|G\|_H = \sup_{u \in \mathcal{L}_2^-} \frac{\| \left[G * u\right]_+ \|}{\|u\|} \ ;$$

in the discrete-time case this formula is still valid if one replaces $\mathcal{L}_2$ by L_2. The H_∞ norm squared is the maximal energy gain from all inputs to all outputs [56], i.e.

$$\|G\|_\infty = \sup_{u\in\mathcal{L}_2} \frac{\|G*u\|}{\|u\|} ;$$

again replace $\mathcal{L}_2$ by L_2 in discrete-time. The H_2 norm squared is the expected power $E(y^T y)$ in the output if the system is driven by white noise of unit variance.

It is immediately seen from the expressions above that $\|G\|_H \leq \|G\|_\infty$. It is also a standard result [107] that for strictly proper systems, $\|G\|_\infty \leq 2\sum_{i=1}^n \sigma_i$, hence $\|G\|_\infty \leq 2n\|G\|_H$, i.e. the Hankel norm and the H_∞ norm are equivalent for these systems.

For discrete-time systems, it is easy to prove that $\|G\|_2 \leq \sqrt{\min(m,p)}\|G\|_\infty$ if $G(z)$ is a $p\times m$ system; in particular, for SISO (or MISO, or SIMO) systems one has $\|G\|_2 \leq \|G\|_\infty$. This is not the case for continuous-time systems : e.g. for the system $G(s) \triangleq \frac{1}{s+\alpha}$, $\alpha > 0$, $\|G\|_2 = \frac{1}{\sqrt{2\alpha}}$ and $\|G\|_\infty = \frac{1}{\alpha}$; hence, the ratio of the norms $\frac{\|G\|_2}{\|G\|_\infty} = \sqrt{\frac{\alpha}{2}}$ can be given any value by choosing α.

There is no upper bound for the H_∞ norm in terms of the H_2 norm even in the discrete-time case. As a consequence these two norms are not *equivalent*, in other words they are topologically different. This is important in the context of robustness results : e.g. a bound on the H_2 norm of a perturbation cannot guarantee robust stability through the small gain theorem.
The HSH norm, however, does induce the same topology on the set of strictly proper systems; indeed, since $\|G\|_{HSH}^2 = \sum_{i=1}^n \sigma_i^2$, $\|G\|_H \leq \|G\|_{HSH} \leq \sqrt{n}\|G\|_H$, and hence, the Hankel, HSH and H_∞ norms are equivalent. One also has $\frac{1}{2n}\|G\|_\infty \leq \|G\|_{HSH} \leq \sqrt{n}\|G\|_\infty$.

The HSH norm squared may be interpreted as the energy storage capacity of the system (see [197]) : it is the expected energy stored in the system at a certain time, where the system has been driven by white noise of unit variance until that time, i.e. $E(\int_0^\infty y^t y dt)$ (resp. $E(\sum_{i=1}^\infty y_i^T y_i)$) when $u(t)$ (u_i) is unit variance white noise for $t < 0$ $(i < 0)$ and is zero thereafter. Indeed, $E(x(0)x(0)^T) = P$, and

$$\begin{aligned} E(\int_0^\infty y^t y dt) &= E(\int_0^\infty x(0)^T e^{A^T t} C^T C e^{At} x(0)) \\ &= \text{Tr } (\int_0^\infty E(x(0)x(0)^T) e^{A^T t} C^T C e^{At}) = \text{Tr } PQ . \end{aligned}$$

	from all inputs to all outputs	from past inputs to future outputs
worst-case	H_∞	Hankel
average	H_2	HSH

Table 2.1: 'Classification of four system norms.'

A rough and intuitive classification of these norms may thus be attempted on the basis of their character as a worst-case gain or an 'average gain', from either past inputs to future outputs, or all inputs to all outputs, as indicated in Table 2.1.

2.3 Model reduction

Model reduction techniques aim at approximating a model by a model of smaller, usually predetermined, order, such that some measure of the difference between the reduced order model $G_r(s)$ and the original model $G(s)$ is made small.

There is huge body of literature (see [3, 228, 217] and the surveys [53, 105, 276, 60, 23]) on approximation and model reduction; to give a complete overview is not within the scope of this thesis (note that a lot of useful references may be found in the bibliography on p. 237). Only a selection of methods and results relevant to control design are highlighted here.

2.3.1 Sparse and dense

Model reduction methods for control may be classified into two different families, i.e. so-called *dense methods* and *sparse methods*. These methods differ in their range of applicability : sparse methods are used for very large order systems (with hundreds of states or more) whose A-matrix is *sparse*, i.e. the matrix-vector multiplication Ax, whose flop count is normally about $2n^2$, may be computed in $O(n)$ operations. Typically state space models resulting from Finite Element Modeling exhibit sparsity.
The terminology *'sparse methods'* is somewhat deceptive : in fact these techniques can be used for systems with dense state space matrices too, but of course at the cost of higher computational complexity. An equally adequate nomencla-

ture is perhaps *large-scale methods*, since they are to be used when the order is so large that the quantities that are generally needed for model reduction, e.g. Schur matrices, gramians, cannot be computed reliably. These quantities then must be approximated using iterative techniques. Thus, in general, sparse methods are not used because they are preferred, but because there is no other choice.

In principle large-scale problems do not necessarily exhibit sparsity; there are however rather few examples of large-scale problems that are not sparse [91, 92]. We shall deal only with dense methods here; for surveys of sparse methods see [232, 51, 52, 136, 274].

2.3.2 Projection of dynamics

One very crude way of reducing the order of a system is simply and bluntly to discard some components of the state vector; this process is called *truncation*. If the state vector is reordered in such a way that the components which are retained are the first r components, and this reordered realization is partitioned as follows :

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & D \end{array}\right]$$

where $A_{11} \in \mathbb{R}^{r\times r}$, $A_{22} \in \mathbb{R}^{r\times r}$, and the other matrices have compatible dimensions, then the reduced order model is $[A_{11}, B_1, C_1, D]$.
In general there is no indication whatsoever that this brute-force method will yield an acceptable reduced-order model, of which the dynamical behavior will in any way resemble that of the full order system; yet for some realizations, the contribution of some states to the input-output behavior can be shown to be negligible or at least small, so that truncation is justified.

The procedure of transforming a realization by a similarity transformation and subsequently truncating is also known as *projection of dynamics*.
This nomenclature comes from the fact that this process may be regarded as a projection in the state-space : let the system $G(s)$ be described by the equations

$$\begin{aligned} \dot{x} &= Ax + Bu \ , \\ y &= Cx + Du \ , \end{aligned}$$

and let $\tau \stackrel{\triangle}{=} RL$, where $R, L^T \in \mathbb{R}^{n\times r}$ and $LR = I_r$; τ is a projection onto $\mathcal{R}(R)$, the range of R, along $\mathcal{N}(L)$, the null space of L. If this projection can be chosen in such a way that at each time, $x(t) \approx \tau x(t) = RLx(t)$, then

$$RL\dot{x} \approx ARLx + Bu \ ,$$

$$y \approx CRLx + Du \ ,$$

or, defining $x_r \triangleq Lx$, the system is approximately described by

$$\begin{aligned} \dot{x_r} &= LARx_r + LBu \ , \\ y &= CRx + Du \ . \end{aligned}$$

Since $LR = I_r$, it is always possible to find a T such that $L = \begin{bmatrix} I_r & 0 \end{bmatrix} T$ and $R = T^{-1}\begin{bmatrix} I_r \\ 0 \end{bmatrix}$, so that the system $G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ is approximated by

$$\left[\begin{array}{c|c} LAR & LB \\ \hline CR & D \end{array}\right] = \left[\begin{array}{c|c} \begin{bmatrix} I_r & 0 \end{bmatrix} TAT^{-1}\begin{bmatrix} I_r \\ 0 \end{bmatrix} & \begin{bmatrix} I_r & 0 \end{bmatrix} TB \\ \hline CT^{-1}\begin{bmatrix} I_r \\ 0 \end{bmatrix} & D \end{array}\right] .$$

This class encompasses a surprising number of model reduction methods.

2.3.3 Some classes of model reduction algorithms

Among the oldest and most natural model reduction techniques are *modal reduction* [69, 117, 300, 102, 181, 1, 54, 64] and aggregation methods [25, 166], where the poles of the reduced order system are taken to be a subset of the poles of the original system, or, more generally, where the poles of the reduced order system are fixed in advance. The main advantage of these methods is their simplicity. *Modal truncation* is one such technique, where truncation is performed on a modal form. When one seeks to minimize a norm, modal reduction present convex optimization problems for SISO, SIMO and MISO systems, but not for MIMO systems[3].

Another class of algorithms relies on truncation of realizations where certain gramians (not necessarily always the observability and controllability gramians of the system itself) are balanced; these algorithms shall be labeled *balancing reduction methods*. Many of the model reduction methods dealt with or proposed in this thesis fall in this category. The most famous member of the class is *balanced truncation* [195, 216, 94], i.e. truncation of the balanced realization. It has some nice properties, e.g. the H_∞ error is bounded above by the sum of the discarded Hankel singular values multiplied by two.

[3]This may easily be seen by looking at a pole-residue representation of the system : modal reduction only tries to determine optimal residues. For SIMO or MISO systems, these residues are full rank, unrestricted matrices, whereas for MIMO systems, the residues are generically rank-one matrices, i.e. products of unrestricted matrices.

We shall come back to balanced truncation and its cousins in later chapters.

Optimal reduction algorithms are those that minimize the error with respect to some norm, e.g. the H_∞ or the H_2 norm. In general these objectives give rise to problems that are not analytically solvable, and iterative methods must be resorted to. Moreover, the problems are most often non-convex and global minimization cannot be guaranteed within polynomial time. The most notable exception is Hankel norm approximation [4, 59, 161, 106, 107, 235, 246, 127, 307]. This has a similar error bound, and the static gain can be manipulated in such a way that the error bound is half the bound of balanced truncation.
It is trivial that the norm of the approximation error is monotonically non-increasing as a function of the reduced order r for optimal reduction algorithms, as the search space is growing with the order; for other methods, this is generally a desirable feature, but cannot always be guaranteed.

Yet another kind of methods are Padé approximation [212, 62, 10], Routh approximation [138], and q-COVER [315, 314, 81, 250, 22, 255, 256, 174, 289]. A Padé approximant of order r matches the first $2r$ Markov parameters of the original system. A Padé approximation is not necessarily stable where the original system is stable, which is a possible disadvantage; modifications have been proposed [247]. For an overview of Padé approximations see [60].
A Routh approximation [138] matches all coefficients of the Routh table up to a certain order. This approach does preserve stability, while still matching the first r Markov parameters.
Another Padé-type approach is q-COVER, matching the first r Markov parameters and the first r output covariances R_i (see Subsection 2.2.4).

2.3.4 Shaping the approximation error

A feature many methods mentioned above, more specifically those obtained by projection of dynamics, have in common is that the original and the reduced order model have the same static gain D, i.e. the approximation is perfect for $s \to \infty$. To have a perfect approximation at some frequency is never undesirable, but unfortunately the response of plants or controllers at very high frequencies is often unimportant for the closed-loop performance. One is often more interested in a perfect match at a more critical frequency, or to capture the behavior at steady-state. The frequency with perfect matching can be shifted from $s = \infty$ to $s = 0$ by first applying the transformation $s \to s^{-1}$, reducing the transformed model $G(s^{-1})$, and re-transforming $s \to s^{-1}$. This is called a *singular perturbation approximation* [95, 96, 97, 98, 159, 176, 218, 236, 253].

A broader class of model reduction algorithms are the frequency weighted variants, i.e. where a weighted modeling error $W_o(s)(G(s) - G_r(s))W_i(s)$ is made small in some sense, with $W_o(s)$ and $W_i(s)$ an output and input weighting system. These weighting systems stress frequencies where, and directions in which, the error $G(s) - G_r(s)$ should be small. Frequency weighted extensions exist for H_2-optimal model reduction [125], Hankel norm approximation [167, 137, 322], Balanced Truncation [93, 14, 15], and others [203].

An important special case of frequency weighted model reduction is *relative error* model reduction. The *relative error* of a right invertible system G and its approximant G_r may be defined as a system Δ_r satisfying $G_r - G = G\Delta_r$; a dual definition holds for left invertible systems. Alternative, not equivalent, definitions may also be thought of (see [124]). Relative error model reduction attempts to make some measure of the relative error small. An important algorithm for relative error model reduction is Balanced Stochastic Truncation (BST); it shares with Balanced Truncation the attractive feature of a pre-computable H_∞ error bound.

A rather trivial, but sometimes useful connection between controller reduction and frequency weighted model reduction is that the latter is a special case of the former. Indeed, as shown in Fig. 2.1, finding a system $G_r(s)$ that makes $W_o(s)(G(s) - G_r(s))W_i(s)$ small amounts to the same thing as finding a controller $G_r(s)$ for the standard plant $\begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix} \triangleq \begin{bmatrix} W_oGW_i & -W_o \\ W_i & 0 \end{bmatrix}$ that makes $P_{zw} + P_{zu}G_r(I - P_{yu}G_r)^{-1}P_{yw}$ small.

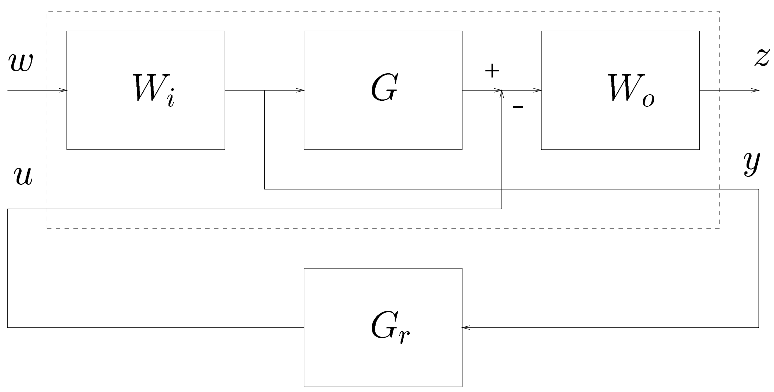


Figure 2.1: Standard plant for weighted model reduction.

2.3.5 Some other issues

A perhaps equally trivial, but important remark to be made is that scaling the inputs and outputs of the system to be reduced generally changes the result of a model reduction procedure, i.e. let $G(s)$ be approximated by $G_r(s)$ and let $\bar{G}_r(s)$ approximate $D_1 G(s) D_2$, where the same model reduction technique is employed and D_1 and D_2 are diagonal matrices, then in general $G_r(s)$ will be different from $D_1^{-1}\bar{G}_r(s)D_2^{-1}$. Taking e.g. $G(s) = \begin{bmatrix} \frac{100}{s+1} & 0 \\ 0 & \frac{1}{s+2} \end{bmatrix}$, $D_1 = I$, and $D_2 = \begin{bmatrix} 1 & 0 \\ 0 & 10000 \end{bmatrix}$, and reducing to order 1, $G_r(s)$ will be close to $\begin{bmatrix} \frac{100}{s+1} & 0 \\ 0 & 0 \end{bmatrix}$, whereas $D_1^{-1}\bar{G}_r(s)D_2^{-1}$ will be close to $\begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{s+2} \end{bmatrix}$.

Therefore, when one is reducing a MIMO plant or controller, one should always ask oneself whether a preliminary scaling of inputs or outputs is needed.

Most model reduction techniques require a stable system to be reduced. Moreover they yield stable reduced order systems. For general model reduction purposes, the first feature may seem a natural constraint, and the second one a desirable property, but especially for controller reduction this is not necessarily so. A controller need not be stable, and to restrict oneself to the class of stable controllers may be a silly if not infeasible constraint. Therefore, one needs a way of reducing unstable systems.

One possible solution is to split the controller into a stable and an unstable part using partial fraction decomposition, leave the unstable part unchanged and reduce only the stable part. While this may be acceptable in many cases (which will often have but few unstable poles), this is certainly *ad hoc* and suboptimal. Obviously anti-stable systems cannot be reduced with this artifice.

Another possibility is to reduce a coprime factorization of G as follows [204] : an unstable system $G \triangleq [A, B, C, D]$ can be factorized as $G = NM^{-1} = \tilde{M}^{-1}\tilde{N}$,

$$\text{where } N \triangleq \left[\begin{array}{c|c} A + BK_r & B \\ \hline C + DK_r & D \end{array}\right] \;,\quad M \triangleq \left[\begin{array}{c|c} A + BK_r & B \\ \hline K_r & I \end{array}\right] \;,$$

K_r being a matrix s.t. $A + BK_r$ is stable,

$$\text{and } \tilde{N} \triangleq \left[\begin{array}{c|c} A + K_eC & B + K_eD \\ \hline C & D \end{array}\right] \;,\quad \tilde{M} \triangleq \left[\begin{array}{c|c} A + K_eC & K_e \\ \hline C & I \end{array}\right] \;,$$

where K_e stabilizes $A + K_eC$.
NM^{-1} is called a *right coprime factorization* of G, $\tilde{M}^{-1}\tilde{N}$ a *left coprime factorization*. The pairs of systems (N, M) and $(\tilde{N}, \tilde{M})$ completely determine G; therefore, a possible way of reducing G is to reduce either $\begin{bmatrix} N^T & M^T \end{bmatrix}^T$ or $\begin{bmatrix} \tilde{N} & \tilde{M} \end{bmatrix}$. If a method employing projection of dynamics is used to approximate $\begin{bmatrix} \tilde{N} & \tilde{M} \end{bmatrix}$ e.g., the r-th order reduced system $\begin{bmatrix} \tilde{N}_r & \tilde{M}_r \end{bmatrix}$ is also a left coprime factorization of a system G_r of order r, in fact obtained by the same projection of dynamics from G.

This technique has been coined graph model reduction [284, 285, 286], [287], fractional model reduction [192], and coprime factor model reduction [112, 189].

Note that coprime factorizations are not unique, and that in general different factorizations result in different reduced order controllers; of special interest are normalized coprime factorizations [286, 288, 115, 144, 188].

2.4 Frequency weightings

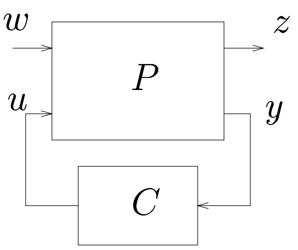
There are various properties that one may wish to retain after controller reduction, stability being the most important one, but also performance, robust stability, robust performance, etc.; any reduced order controller will not only have to be stabilizing, but must also have an acceptable level of performance and robustness against plant variations.

It was stated above that for model reduction to be relevant in a control context, it should be possible to put a frequency weight on the modeling error obtained. Supposing that one has a frequency weighted generalization of a model reduction technique, one should know how to translate one's wishes of preserving certain properties into a suitable frequency weight. Some answers to such questions will be provided in this section.

2.4.1 Stability

Consider a standard plant $P(s)$ of order n_p :

$$P(s) \triangleq \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$



Suppose a controller $C(s) = [A_c, B_c, C_c, D_c]$ (of order n_c) stabilizes $P(s)$, i.e. it internally stabilizes $P_{yu}(s)$. The question is, which frequency weight one should choose so that model reduction with this frequency weight has a better chance of of preserving closed-loop stability.

A qualitative statement one can make about this matter is that in general, the quality of the model around the *cross-over* or *critical* frequency ω_c (the frequency where the sensitivity matrix $(I + P_{yu}C)^{-1}$ peaks) should be higher than in high and low frequency intervals. Weighting with a simple filter that stresses a frequency interval around ω_c, while attenuating other frequencies, already has a better chance of yielding a stabilizing low-order controller than an unweighted approximation; but there are more specific ways of designing stability-oriented weights.

If $C(s)$ is perturbed with an additive perturbation $\delta C(s)$, then stability is guaranteed if $\delta C(s)$ satisfies

$$\|(I - P_{yu}C)^{-1}P_{yu}\delta C\|_\infty < 1 \;\text{ or }\; \|\delta C(I - P_{yu}C)^{-1}P_{yu}\|_\infty < 1 \;.$$

Therefore a good controller reduction method with respect to preserving stability is one that produces a low-order controller $C_r(s)$ with $\|W_s(s)(C(s) - C_r(s))\|_\infty$ or $\|(C(s) - C_r(s))W_s(s)\|_\infty$ small, where $W_s(s) \triangleq (I - P_{yu}(s)C(s))^{-1}P_{yu}(s)$. Note that this weighting system has the poles of $C(s)$ for zeroes.

It is interesting to look at the effect a scaling of the inputs and outputs has on the result of the controller reduction here. Let $\mathcal{D}$ be the set of positive definite diagonal matrices. Replacing u by $D_u^{-1}\bar{u}$ and y by $D_y^{-1}\bar{y}$, where D_u, $D_y \in \mathcal{D}$, one has $\bar{u} = \bar{C}(s)\bar{y} = D_u C(s) D_y^{-1}\bar{y}$, and $\bar{P}_{yu}(s) = D_y P_{yu}(s) D_u^{-1}$.
It is found that

$$\|(I - \bar{P}_{yu}\bar{C})^{-1}\bar{P}_{yu}(\bar{C} - \bar{C}_r)\|_\infty = \|D_y(I - P_{yu}C)^{-1}P_{yu}(C - C_r)D_y^{-1}\|_\infty \;.$$

This expression is very reminiscent of the upper bound for the μ-norm (see Subsection 2.4.5), indeed the latter expression is an upper bound for the quantity

$\mu((I - P_{yu}(s)C(s))^{-1}P_{yu}(s)(C(s) - C_r(s)))$, where the scaling matrices are constant and restricted to $\mathcal{D}$.

Remark that $\|D_y(I - P_{yu}(s)C(s))^{-1}P_{yu}(s)(C(s) - C_r(s))D_y^{-1}\|_\infty < 1$ is also a sufficient condition for closed-loop stability; whenever one has used the weighting above, and finds that $\|(I - P_{yu}(s)C(s))^{-1}P_{yu}(s)(C(s) - C_r(s))\|_\infty > 1$ and therefore stability is not guaranteed, one may consider finding an optimal output scaling $D_y \in \mathcal{D}$, i.e.

$$D_y = \arg\min_{D_y \in \mathcal{D}} \|D_y(I - P_{yu}(s)C(s))^{-1}P_{yu}(s)(C(s) - C_r(s))D_y^{-1}\|_\infty .$$

Actually, the condition $\|D_y(I - P_{yu}(s)C(s))^{-1}P_{yu}(s)(C(s) - C_r(s))D_y^{-1}\|_\infty < \gamma$, with γ a positive real number, may be formulated as a Linear Matrix Inequality using an extension of the Bounded Real Lemma, and thus can be solved using convex programming methods; see [26, 57] for details. Arguably, this is an overkill if one merely wants to verify closed-loop stability, since it is much faster to simply compute the eigenvalues of $(I - P_{yu}(s)C_r(s))^{-1}P_{yu}(s)$. But one may consider going further than this and optimize over the scaling and reduced order controller simultaneously, i.e.

$$\min_{D_y \in \mathcal{D}, C_r(s) \in \mathcal{S}_r} \|D_y(I - P_{yu}(s)C(s))^{-1}P_{yu}(s)(C(s) - C_r(s))D_y^{-1}\|_\infty .$$

A naive way to do this is to keep D_y fixed at first, while finding a reduced-order $C_r(s)$ minimizing the scaled H_∞ norm, and then to keep $C_r(s)$ fixed while minimizing over D_y using convex optimization, yielding 'Algorithm' 2.1.

Here the notation

$$\text{appr} \min_{C_r \in \mathcal{S}_r} f(C_r)$$

is used to denote the approximant of order r obtained with a reduction method seeking to make $f(C_r)$ small. *Which* reduction method is to be used is not specified here, although this may make all the difference, of course.

While it is true that there are efficient (polynomial-time) and globally convergent algorithms for solving the Linear Matrix Inequalities for the minimization over D_y [26, 57], the computational complexity of these algorithms still does increase rather fast, making this scheme impractical for all but rather small problems. Therefore, it will not be dealt with in detail, but we will come back to the idea in Subsection 2.4.5.

2.4.2 Performance

In the case that loss of stabilization by reducing the controller need not be feared, one may consider weighting systems that only aim at conserving the per-

Algorithm 2.1 : stability-oriented reduction with optimal scaling

$D_0 = I$
$k = 0$
do $\{$

$$\begin{array}{rcl} k & \leftarrow & k+1 \\ C_r^{[k]} & = & \text{appr} \min_{C_r \in \mathcal{S}_r} \| D_{k-1}(I - P_{yu}C)^{-1} P_{yu}(C - C_r) D_{k-1}^{-1} \|_\infty \\ D_k & = & \arg \min_{D_y \in \mathcal{D}} \| D_y (I - P_{yu}C)^{-1} P_{yu}(C - C_r) D_y^{-1} \|_\infty \end{array}$$

$\}$ **until convergence**

formance.

Consider again the standard plant $P(s)$, for which a satisfactory controller $C(s)$ has been found. Let $C_r(s) = C(s) + \delta C(s)$ be a candidate reduced-order controller, and define

$$\begin{array}{rcl} H_{zw}(C) & \triangleq & L_f(P, C) = P_{zw} + P_{zu}C(I - P_{yu}C)^{-1} P_{yw} \ , \\ H_{zw}(C_r) & \triangleq & L_f(P, C_r) = P_{zw} + P_{zu}C_r(I - P_{yu}C_r)^{-1} P_{yw} \ . \end{array}$$

Often, the standard plant is drawn up in such a way that a smaller norm or measure of H_{zw} (*which* measure we do not define here), expresses better performance of the control system. One may argue, therefore, that the aim of a performance oriented controller reduction method should be to make $\|H_{zw}(C_r)\|$ small, rather than $\|H_{zw}(C_r) - H_{zw}(C)\|$. Of course such a scheme would be a controller design method rather than a controller reduction method, but that does not make the idea more or less advisable. However, there may be good reasons to resort to controller reduction, seeking to keep the latter norm small; often, the 'full order' controller is designed in an optimal way, and contains a lot of information about the dynamics one should mimic to make $\|H_{zw}(C_r)\|$ small.

It is easily found that

$$\begin{array}{rcl} H_{zw}(C_r) - H_{zw}(C) & = & P_{zu}(I - C_r P_{yu})^{-1} \delta C (I - P_{yu}C)^{-1} P_{yw} \ , \quad (2.5) \\ & \approx & P_{zu}(I - C P_{yu})^{-1} \delta C (I - P_{yu}C)^{-1} P_{yw} \ ; \end{array}$$

the last expression is a first order approximation. Therefore, natural left and right weighting systems for the preservation of performance properties are :

$$W_{p,l} \triangleq P_{zu}(I - CP_{yu})^{-1} \quad \text{and} \quad W_{p,r} \triangleq (I - P_{yu}C)^{-1}P_{yw} \ .$$

Again these weighting systems have the poles of $C(s)$ for zeroes.

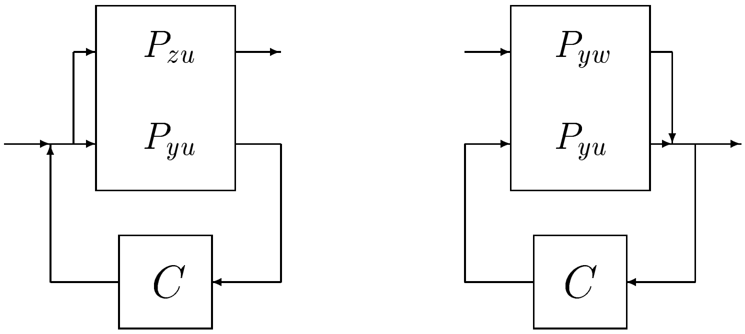


Figure 2.2: $W_{p,l}(s)$ (left) and $W_{p,r}(s)$ (right)

Note that when e.g. $W_{p,l}$ is implemented as the series connection of P_{zu} and $(I-CP_{yu})^{-1}$, the order of this realization will be $2n_p+n_c$, which is non-minimal. A more careful implementation yields a minimal realization; it is easily seen that $W_{p,l}$ and $W_{p,r}$ are represented by the block diagrams depicted in Fig. 2.2, where

$$\left[\begin{array}{c} P_{zu} \\ P_{yu} \end{array}\right] = \left[\begin{array}{c|c} A & B_u \\ \hline C_z & D_{zu} \\ C_y & D_{yu} \end{array}\right] \quad \text{and} \quad \left[\begin{array}{c} P_{yw}P_{yu} \end{array}\right] = \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_y & D_{yw} & D_{yu} \end{array}\right] ,$$

and one finds :

$$W_{p,l} = \left[\begin{array}{cc|c} A + B_uD_cC_y & B_uC_c & B_u \\ B_cC_y & A_c & 0 \\ \hline C_z + D_{zu}D_cC_y & D_{zu}C_c & D_{zu} \end{array}\right]$$

and

$$W_{p,r} = \left[\begin{array}{cc|c} A + B_uD_cC_y & B_uC_c & B_w + B_uD_cD_{yw} \\ B_cC_y & A_c & B_cD_{yw} \\ \hline C_y & 0 & D_{yw} \end{array}\right] .$$

These realizations have order $n_p + n_c$.

In the case of a SISO controller $C(s)$ (i.e. the plant has only one measured output and only one control input available, also called a *SASS* (single actuator single sensor) plant), one may apply the following one-sided weighting :

$W_p(s) \triangleq P_{zw}(I - CP_{yu})^{-2}P_{yw}$. Note that this weighting function has the poles of $C(s)$ for zeroes, with twice the multiplicity.

For MAMS plants, the question of the effect of a scaling of inputs u or outputs y on the result of the controller reduction should be asked again here; remark that one cannot re-scale the inputs w or the outputs z, without changing the performance object function. Replacing u by $D_u^{-1}\bar{u}$ and y by $D_y^{-1}\bar{y}$ as in Subsection 2.4.1, one has $z = P_{zu}u = P_{zu}D_u^{-1}\bar{u} = \bar{P}_{zu}\bar{u}$, and similarly $\bar{P}_{yw} = D_y P_{yw}$; hence, the weights $W_{p,l}$ and $W_{p,r}$ become $\bar{P}_{zu}(I - \bar{C}\bar{P}_{yu})^{-1} = P_{zu}(I - CP_{yu})^{-1}D_u^{-1}$ and $(I - \bar{P}_{yu}\bar{C})^{-1}\bar{P}_{yw} = D_y(I - P_{yu}C)^{-1}P_{yw}$, respectively;

$$\begin{aligned} \text{thus}\ \ & \bar{P}_{zu}(I - \bar{C}\bar{P}_{yu})^{-1}(\bar{C} - \bar{C}_r)(I - \bar{P}_{yu}\bar{C})^{-1}\bar{P}_{yw} \\ &= P_{zu}(I - CP_{yu})^{-1}(C - C_r)(I - P_{yu}C)^{-1}P_{yw}\ , \end{aligned}$$

i.e. the object function of the frequency weighted model reduction problem is insensitive to the scaling.

Recall that this weighted modeling error is only a first-order approximation of the mismatch on the closed-loop transfer; a weighting that gives an exact expression is given by Expression 2.5, but there one of the weighting systems depends on the reduced-order controller, which is of course unknown before reduction. Iteration is a possible way out of this, see Alg. 2.2.

Algorithm 2.2 : iterative re-design of performance-oriented weights

$C_r^{[0]}(s) = C(s)$
$W_{p,r}(s) \triangleq (I - P_{yu}(s)C(s))^{-1}P_{yw}(s)$
$k = 0$
do {

$$\begin{aligned} k &\leftarrow k+1 \\ W_{p,l}^{[k]}(s) &\triangleq P_{zu}(s)(I - C_r^{[k-1]}(s)P_{yu}(s))^{-1} \\ C_r^{[k]} &= \text{appr} \min_{C_r \in \mathcal{S}_r} \|W_{p,l}^{[k]}(C - C_r)W_{p,r}\| \end{aligned}$$

} **ad nauseam**

A word of caution is in order here : whether it makes sense to perform this iter-

ation until convergence is reached depends wholly on the nature of the weighted approximation algorithm used. If this algorithm has no claim towards optimality with respect to a certain measure of performance of the reduced order controller, as is e.g. the case with the methods based on balancing, to be described hereafter, then it is not to be supposed that any such measure would steadily decrease during this iteration. In such a case, the reduced order controller reached after the first iteration may well be the best one. Nevertheless, even for suboptimal algorithms, a modest improvement of performance may often be obtained with a relatively small effort, by application of just one iteration or so.
Furthermore, even if the weighted approximation algorithm is optimal with respect to a certain norm, there is no guarantee that this iteration converges; the only thing one can say in this case is that *if* the iteration converges, it does so to a local minimum of the problem

$$\min_{C_r} \|H_{zw}(C_r) - H_{zw}(C)\| \ .$$

For these reasons, *ad nauseam* was written here in lieu of a more rigorous stopping criterion.

It should be noted that this discussion of weightings for performance preservation is by no means complete. More sophisticated weightings may be employed to get satisfactory performance. To mention just one, the method of [113, 114] gives weights $W_1(s)$ and $W_1(s)$ such that if $\|W_2^{-1}(C - C_r)W_1^{-1}\|_\infty < 1$, the same H_∞ performance bound is guaranteed for the reduced order controller C_r as for the original controller $C(s)$.

2.4.3 Combined stability and performance

We may minimize two frequency weighted norms simultaneously, one that guarantees robust stability and another that guarantees robust performance. Note that expression 2.5 may be rewritten as

$$\tilde{H}_{zw} = P_{zu}(I - CP_{yu})^{-1}(I - \delta CP_{yu}(I - P_{yu}C)^{-1})^{-1}\delta C(I - P_{yu}C)^{-1}P_{yw} \ ,$$

and that in many norms $\|(.)\|_N$, e.g. the H_2 and the H_∞ norm, one has

$$\|H_1(s)H_2(s)\|_N < \|H_1(s)\|_\infty \|H_2(s)\|_N \ .$$

In the (frequently occurring) special cases where $P_{zu} = P_{yu}$ or $P_{yw} = P_{yu}$, and also in the case where P_{yu} is SISO, if one guarantees that

$$\|\delta CP_{yu}(I - P_{yu}C)^{-1}\|_\infty = \|\delta CW_s\|_\infty < \alpha < 1$$

(which means that a sufficient condition for stability is fulfilled), then the performance degradation $\|\tilde{H}_{zw}\|_N$ is less than A if

$$\|P_{zu}(I - CP_{yu})^{-1}\delta C(I - P_{yu}C)^{-1}P_{yw}\|_N < (1-\alpha)A \ .$$

We shall consider only the H_∞ norm. In these cases, the stability and performance weighting functions may be combined into one weighting function (at the price of being more conservative) by specifying (e.g. for the case $P_{zu} = P_{yu}$) :

$$\|P_{yu}(I - CP_{yu})^{-1}\delta C \begin{bmatrix} I\alpha^{-1} \\ \frac{(I-P_{yu}C)^{-1}P_{yw}}{(1-\alpha)A} \end{bmatrix} \|_\infty \leq 1 \ .$$

The design parameter α can be used to trade off stability and performance against each other : α smaller means more stability-oriented frequency weighting, as $\alpha \to 0$, the weighting will become equivalent with the stability-oriented weighting W_s, whereas $\alpha \to 1$ corresponds to only performance-oriented weighting.

In the general case, a more conservative bound

$$\|\delta C(I - P_{yu}C)^{-1}P_{yw}\|_\infty \leq (1-\alpha)A\|P_{zu}(I - CP_{yu})^{-1}\|_\infty^{-1}$$

or

$$\|P_{zu}(I - CP_{yu})^{-1}\delta C\|_\infty \leq (1-\alpha)A\|(I - P_{yu}C)^{-1}P_{yw}\|_\infty^{-1}$$

guarantees the same bound on performance degradation, which can again be combined with the stability weighting function since both weightings are only one-sided :

$$\| \begin{bmatrix} P_{yu}\alpha^{-1} \\ P_{zu}BA^{-1}(1-\alpha)^{-1} \end{bmatrix} (I - CP_{yu})^{-1}\delta C\|_\infty \leq 1$$

where $B \triangleq \|(I - P_{yu}C)^{-1}P_{yw}\|_\infty$.

2.4.4 Robust stability

Controller reduction is closely connected to robustness issues, and the frequency weights introduced above, aiming at preservation of stability or other desirable properties after reduction, are based using on robustness results as we have seen. Retaining stability after reduction does not imply, however, that robustness against plant variations will also be retained, and different weighting functions will have to be designed for assuring this.

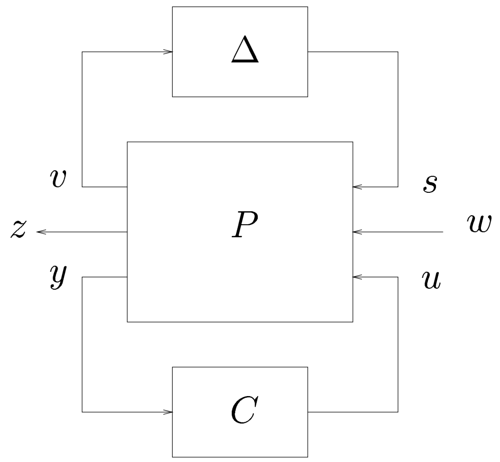


Figure 2.3: Standard plant including perturbations.

Suppose one has a generalized plant as in Fig. 2.3, where all possible perturbations $\Delta(s)$ may be modeled by $\|\Delta(s)\|_\infty \le 1$. H is the transfer matrix of this generalized plant with its feedback loop closed using a controller C but with its perturbation loop open, G_{zw} is the transfer matrix of the plant with both loops closed. A controller C robustly stabilizes this family of possible plants if

$$\|H_{vs}\Delta(s)\|_\infty < 1,\ \forall\ \|\Delta(s)\|_\infty \le 1\ , \qquad \text{i.e. if } \ \|H_{vs}\|_\infty < 1\ .$$

Note that $H_{vs} = P_{vs} + P_{vu}C(I - P_{yu}C)^{-1}P_{ys}$. Suppose one has found a (high-order) controller C that is robustly stable under these perturbations, and $\|H_{vs}\|_\infty = \gamma < 1$. Then robust stability will still be maintained by the reduced-order controller C_r if $\|H_{vs}(C_r) - H_{vs}(C)\|_\infty < 1 - \gamma$.

Now $$H_{vs}(C_r) - H_{vs}(C) = P_{vu}(I - C_r P_{yu})^{-1}\delta C(I - P_{yu}C)^{-1}P_{ys}\ ,$$

therefore frequency weighted model reduction with weighting functions

$$W_i(s) \triangleq (I - P_{yu}C)^{-1}P_{ys} \ \text{ and } \ W_o(s) \triangleq P_{vu}(I - CP_{yu})^{-1} \text{ is well suited.}$$

Again, this can be simplified in the special cases $P_{vu} = P_{yu}$ and $P_{ys} = P_{yu}$, and when P_{yu} is SISO, in a similar way as before.

2.4.5 Robustness using μ-synthesis

To date, the most powerful way of dealing with robustness issues is μ-synthesis. To expose the theory of the structured singular value (SSV or μ-norm) and μ-synthesis is beyond the scope of this thesis, but some of the basics are roughly sketched below. For details we refer to [84, 211, 30] and the references found therein.

The μ-norm or SSV is defined as follows :

$$\mu_\Delta(M) \triangleq (\min\{\bar{\sigma}(\Delta)|\Delta \in S_\Delta, \det(I - M\Delta) = 0\})^{-1} ,$$

where S_Δ is the set of complex matrices with a certain block diagonal structure. The popular nomenclature μ-norm is deceiving, since this quantity is in fact not a norm (it does not satisfy the triangle inequality). The optimization problem underlying this definition is suffering from local minima and is not analytically solvable, even though very satisfactory algorithms exist for this minimization [70, 210].

However, an excellent upper bound for this quantity turns out to be given by the minimum of a convex optimization problem :

$$\mu_\Delta(M) \leq \inf_{D \in S_D} \bar{\sigma}(DMD^{-1})$$

where S_D is the set of matrices D such that $D\Delta = \Delta D$ for all $\Delta \in S_\Delta$. For the worst known example, the upper bound is only 15 % larger than the actual structured singular value[4].

A broad range of robust stability and performance analysis problems, where structured perturbations on the plant are allowed, can be formulated as

$$\sup_{\omega \in \mathbb{R}} \mu_\Delta\left(G(j\omega)\right) \leq 1$$

for some system $G(s)$, the definition of which is of course dependent on the actual robustness problem [84, 30, 88].
Using the upper bound for μ given above, this condition is found to be fulfilled if the following holds :

$$\sup_{D(j\omega) \in S_D, \omega \in \mathbb{R}} \bar{\sigma}(D(j\omega)G(j\omega)D(j\omega)^{-1}) \leq 1$$

or if

$$\sup_{D(s) \in S_D} \|D(s)G(s)D(s)^{-1}\|_\infty \leq 1$$

where $S_{D,s}$ is the set of stable and inverse stable scaling systems $D(s)$ such that $D(j\omega) \in S_D$, $\forall\, \omega$.

[4]This statement is valid only for non-repeated uncertainty blocks.

Likewise, many robustness optimization problems may be posed as :

$$\min_{C(s)\in S_c,\omega\in\mathbb{R}} \mu_\Delta(F_l(P(j\omega),C(j\omega)))$$

where S_c is the set of all controllers stabilizing plant $P(s)$. This may again be approximated by

$$\min_{C(s)\in S_c,D(s)\in S_{D,s}} \|D(s)F_l(P(s),C(s))D(s)^{-1}\|_\infty \; .$$

A simple-minded but useful attempt at solving the latter problem is alternatively to hold $D(s)$ fixed, finding $C(s)$ by solving a standard H_∞ optimization problem, and to hold $C(s)$ fixed, solving the remaining convex optimization problem for the scaling $D(s)$. This is known as the D-K iteration (Alg. 2.3).

Algorithm 2.3 : D-K iteration

$D_0 = I$
$k = 0$
do {

$$\begin{aligned} k &\leftarrow k+1 \\ C^{[k]}(s) &= \arg\min_C \|D_{k-1}F_l(P,C)D_{k-1}^{-1}\|_\infty \\ D_k(s) &= \arg\min_{D,D^{-1}\in S_D} \|DF_l(P,C^{[k]})D^{-1}\|_\infty \end{aligned}$$

} **until convergence**

Convergence of this procedure to the global minimum is not guaranteed, but it often works well. It amounts to a kind of cyclic coordinate optimization method, where one optimizes now over one set of coordinates in the search space, holding the other coordinates fixed, and then over the set of the remaining coordinates, holding the coordinates of the first set fixed, and repeats this procedure until convergence. If the optimization steps over the first set of coordinates and over the second set of coordinates are globally optimal, then under mild conditions, this process converges to a stationary point of the optimization problem over the entire search space [39].

To find the scalings $D(s)$, one should in principle find the minimizing scaling $D(j\omega)$ in every frequency, and find a realization $D(s)$ interpolating all these points; if this were possible, then indeed the D-K iteration could be proven to be globally convergent. In practice, this is of course only done for a finite number of frequency points, and on top of this, $D(s)$ is usually approximated by a low-order system; for this reason, the D-K iteration may sometimes suffer slightly from limit-cycling. In general however, the D-K iteration converges in only a few steps, but nevertheless most often results in a controller of rather high order, which have to be approximated by a lower order controller.

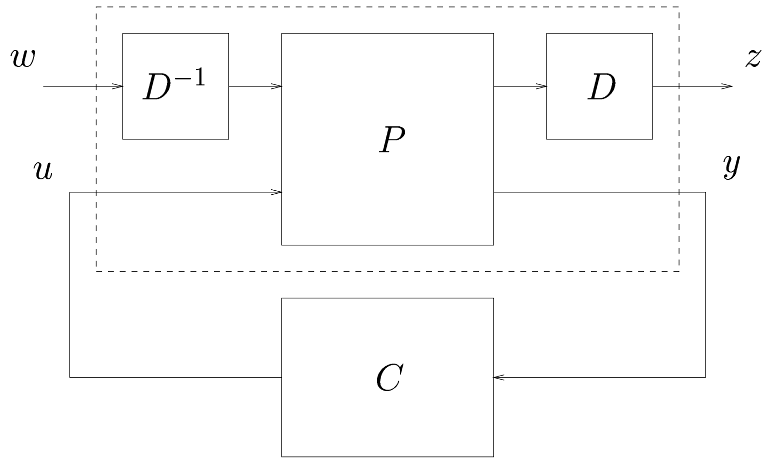


Figure 2.4: Standard plant including the μ-norm scalings.

If one wants to keep the robustness properties that the controller was designed for, one should seek to preserve the μ-norm $\|D(s)F_l(P(s), C(s))D(s)^{-1}\|_\infty$; hence, it is natural to try to perturb the closed-loop performance of the plant of Fig. 2.4, i.e. $D(s)F_l(P(s), C(s))D(s)^{-1}$, as little as possible, with the closed-loop error measured in the H_∞ norm, and thus

$$\begin{aligned} W_{\mu,r}(s) &\triangleq D(s)P_{zu}(s)(I - C(s)P_{yu}(s))^{-1} \\ W_{\mu,l}(s) &\triangleq (I - P_{yu}(s)C(s))^{-1}P_{yw}(s)D(s)^{-1} \end{aligned}$$

are good frequency weights for robustness with respect to structured perturbations. Note that the weighting of Subsection 2.4.4 fits seamlessly into this framework : this corresponds to the case of unstructured perturbations, which means that the scalings are just a scalar multiple of the unit matrix, and can therefore be omitted.

To obtain an optimal reduced-order controller with respect to the μ-norm, one should replace the H_∞ design step in the D-K iteration with an optimal fixed-order H_∞ controller design. Unfortunately no satisfactory algorithms are available for direct fixed-order H_∞-optimal control design; as a substitute, one can approximate this step by an H_∞ design and a subsequent weighted controller reduction with performance-oriented weights, see Alg. 2.4.

Algorithm 2.4 : modified D-K iteration (with reduction in the loop)

$D_0 = I$
$k = 0$
do {

$$\begin{aligned}
k &\leftarrow k+1 \\
C(s) &\leftarrow \arg\min_C \|D_{k-1}F_l(P,C)D_{k-1}^{-1}\|_\infty \\
W_{p,l}(s) &\leftarrow P_{zu}(I - CP_{yu})^{-1} \\
W_{p,r}(s) &\leftarrow (I - P_{yu}C)^{-1}P_{yw} \\
C_r^{[k]}(s) &= \text{appr} \min_{C_r \in \mathcal{M}_r} \|D_{k-1}W_{p,l}(C - C_r)W_{p,r}D_{k-1}^{-1}\|_\infty \\
D_k(s) &= \arg \min_{D, D^{-1} \in S_D} \|DW_{p,l}(C - C_r^{[k]})W_{p,r}D^{-1}\|_\infty
\end{aligned}$$

} **until convergence**

Of course there are even more possibilities here for the algorithm to fail to converge, since there is no guarantee anymore that any object function decreases. Yet if the combination of the H_∞ design and model reduction steps were replaced by

$$C_r^{[k]}(s) = \arg \min_{C_r(s) \in \mathcal{M}_r} \|D_{k-1}(s)F_l(P(s), C_r(s))D_{k-1}^{-1}(s)\|_\infty \ ,$$

then the iteration would converge under the same (unrealistic) assumptions as the D-K iteration. Therefore one can expect that Alg. 2.4 will also converge if these two combined steps are a good enough approximation of the fixed-order H_∞-optimal design above. Obviously, the algorithm becomes more hazardous as a lower order r is chosen.

Recall that considerations about input and output scaling led to Alg. 2.1, and the similarity of that algorithm to our modified D-K iteration Alg. 2.4 is remarkable.

Indeed, consider the standard plant $\tilde{P}$ depicted in Fig. 2.5, with transfer matrix
$\tilde{P}(s) = \begin{bmatrix} (I - P_{yu}C)^{-1}P_{yu}C & -(I - P_{yu}C)^{-1}P_{yu} \\ I & 0 \end{bmatrix}$.

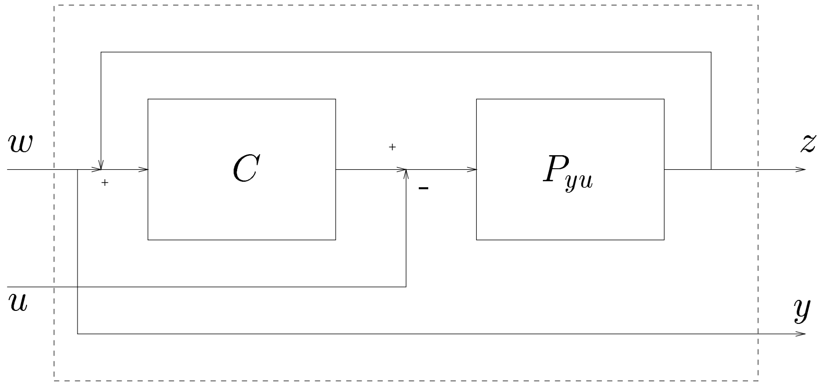


Figure 2.5: Standard plant $\tilde{P}$ generating stability-oriented weighting.

For this plant, the performance-oriented weighting is equivalent to the stability-oriented weighting for plant P_{yu} and controller C : the weights are found to be $W_{p,r}(s) = I$ and $W_{p,l}(s) = (I - P_{yu}C)^{-1}P_{yu}C)$.

Alg. 2.4 for this standard plant is simplified by the fact that the H_∞ design step

$$C(s) \leftarrow \arg\min_C \|D_{k-1}F_l(\tilde{P}, C)D_{k-1}^{-1}\|_\infty$$

is superfluous; indeed, since $F_l(\tilde{P}, C) = 0$, the minimizer of this H_∞ problem is always $C(s)$, the initial controller. Thus Alg. 2.4 becomes Alg. 2.5.

Note that if $\operatorname{appr\,min}_{C_r \in \mathcal{S}_r}$ is replaced by $\arg\min_{C_r \in \mathcal{S}_r}$, the convergence of this algorithm is guaranteed under the same assumptions as for the D-K iteration.

If S_D consists of diagonal scaling systems, this algorithm is almost identical to Alg. 2.1, the only difference being that there the scalings $D_k \in \mathcal{D}$ are restricted to *constant* diagonal matrices. Indeed, there is but little reason to restrict oneself to constant scalings : this is merely more conservative, whereas the optimization problem, admittedly more complex, is barely harder, as more efficient algorithms are available to compute dynamic scalings [210]. In fact, one could even relax the restriction of diagonality, and allow full scaling systems $D(s)$; as long as these are stable and inverse stable, $\|D(I - P_{yu}C)^{-1}P_{yu}(C - C_r)D^{-1}\|_\infty < 1$ still guarantees closed-loop stability.

Algorithm 2.5 : modified D-K iteration applied to plant $\tilde{P}$

$D_0 = I$
$k = 0$
do {

$$\begin{aligned} k &\leftarrow k+1 \\ C_r^{[k]} &= \text{appr} \min_{C_r \in \mathcal{S}_r} \|D_{k-1}(I - P_{yu}C)^{-1} P_{yu}(C - C_r) D_{k-1}^{-1}\|_\infty \\ D_k(s) &= \arg \min_{D(s) \in S_D} \|D(I - P_{yu}C)^{-1} P_{yu}(C - C_r) D^{-1}\|_\infty \end{aligned}$$

} **until convergence**

These procedures will not be illustrated at this point, since no frequency weighted model reduction techniques have been dealt with so far; but in Chapter 3, it will be shown that the methods discussed there may be used in such an iteration.

2.4.6 Approximating the weights?

A final remark should be made about the order of the weighting systems : as shown above, the order of these weights is often higher than that of the controller, and in many cases the amount of computation is a multiple of that of unweighted approximation, whereas a low order weighting would hardly be make any difference in the flop count. A more serious side-effect of this high order is that numerical reliability of the weighted model reduction may deteriorate, sometimes severely.

The question arises whether the weight itself can be approximated. Although this may seem far-fetched, this may often reduce the total operation count. Of course, this preliminary approximation may result in an unsuitable low-order weight, if the order is chosen too low; but it is not uncommon that the weight may be reduced to an order 5 or 10 times lower, without noticeable changes to the final result. One may argue that such a preliminary approximation deprives the user of the guarantees for stability or robust performance that were the starting point for the derivation of those weights; but to obtain guarantees with the original weights, one needs bounds on the frequency weighted modeling H_∞ error, and precisely these error bounds are most often lacking for frequency weighted model reduction methods, or at least more impractical to compute than the property of interest (such as closed-loop stability) itself.

Chapter 3

Closed-loop balancing

In this chapter, we study a well-known model reduction technique, i.e. Enns's Frequency Weighted Balanced Truncation (FWBT) [93], combined with some of the frequency weights presented in Section 2.4.

We show that FWBT with the performance-oriented frequency weights of Subsection 2.4.2, is mathematically equivalent to Closed-Loop Balanced Truncation (CLBT), a model reduction method recently proposed by Wortelboer [302, 305, 65]. CLBT aims at perturbing the closed-loop transfer function H_{zw} of a standard plant as little as possible, by balancing the parts of the closed-loop observability and controllability gramians relevant to the controller states. In contrast with FWBT, it is not restricted to controllers that are themselves stable.
Furthermore this method is superior to FWBT with respect to computational load; it should therefore always be used instead of FWBT with these weightings.

Even better results may be obtained for observer-based controllers : a significant amount of computations can be saved because of the observer–state feedback structure of the controller. In the special case of H_2-optimal control, CLBT is shown to be equivalent to the controller reduction technique BCRAM (Balanced Controller Reduction Algorithm Modified) ([312], see also [68]).

Furthermore, we show how similar savings may be made for the stability-oriented frequency weight of Subsection 2.4.1, both for general and for observer-based controllers.

3.1 Background : balancing methods

3.1.1 Balanced Truncation

One of the success stories of model reduction is Balanced Truncation, introduced by Moore early in the last decade [195]. As some of the algorithms that will be presented later lean heavily on the concept of balancing, a brief and intuitive outline of the algorithm and its main properties is given here. Let $G(s)$ be a C.T. system of McMillan degree n that we wish to approximate by a system of McMillan degree at most r, and let $[A, B, C, D]$ be a balanced realization of $G(s)$, with $A\Sigma + \Sigma A^T + BB^T = 0$ and $A^T\Sigma + \Sigma A + C^TC = 0$. The energy interpretations of the gramians (see Subsection 2.2.3) show that the diagonal elements σ_i of Σ are a measure of how much energy is injected into the i-th state component x_i, and of how much energy x_i contributes to the output energy. Furthermore, the fact that Σ is diagonal indicates that in this realization, the different state components are uncorrelated. It seems reasonable, therefore, to discard the states corresponding to the smallest σ_i first, when one desires to reduce the order of the model, while preserving as much as possible the input-output behavior. Since the Hankel singular values σ_i are ordered in descending order, the proposed reduction method amounts to discarding state components x_i, $i = r + 1, \ldots, n$, i.e. if the realization is partitioned as follows :

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & D \end{array}\right]$$

where $A_{11} \in \mathbb{R}^{r\times r}$, $A_{22} \in \mathbb{R}^{r\times r}$, and the other matrices have compatible dimensions, then the reduced order model is $[A_{11}, B_1, C_1, D]$.

Balanced Truncation, like all model reduction methods that leave the D-matrix of the realization unchanged, gives a perfect match at infinite frequencies.

As mentioned before, finding a balanced realization may be a very ill-conditioned operation, especially for nearly non-minimal systems. This is bad news, since this means that precisely the dynamics that one should be able to discard without noticeable change to the input-output behavior, are hard to truncate away. The problem is alleviated but not solved by so-called *square-root* balancing algorithms [129, 277]. In *balancing free* algorithms the reduced order model resulting from Balanced Truncation is found without actually performing the balancing similarity transformation [169]. Safonov proposed the following so-called Schur method [234] :

- Compute matrices V_r and V_l, whose r columns form bases for the right and left eigenspaces of PQ associated with the r biggest eigenvalues.

- Compute a singular value decomposition $U_e \Sigma_e V_e^T = V_l^T V_r$.
- $L \triangleq \Sigma_e^{-\frac{1}{2}} U_e^T V_l^T$ and $R \triangleq V_r V_e \Sigma_e^{-\frac{1}{2}}$
- $G_r(s) \triangleq [LAR, LB, CR, D]$.

The proof that this reduced-order model is identical to the one obtained by Balanced Truncation may be found in [234]. This procedure may be used wherever two gramian-like matrices need to be balanced. A combination of balancing free and square-root ideas is advanced by Varga [278].

One of the main attractive features of Balanced Truncation is that there is a bound on the H_∞ norm of the error in terms of the Hankel singular values σ_i [107, 93]. In order to give the most general formulation of this bound, the balanced gramians are rewritten as follows :

$$\Sigma = \operatorname{diag}\left(\sigma^{(1)} I_{s_1}, \ldots, \sigma^{(k)} I_{s_k}, \ldots, \sigma^{(m)} I_{s_m}\right),$$

where $\sigma^{(1)} > \sigma^{(2)} > \ldots > \sigma^{(k)} > \sigma^{(k+1)} > \ldots > \sigma^{(m)} \geq 0$, $\sum_{i=1}^m s_i = n$ and $\sum_{i=1}^k s_i = r$. Then

$$\|G(s) - G_r(s)\|_\infty \leq 2 \sum_{i=k+1}^{m} \sigma^{(i)} .$$

Note that the $\sigma^{(i)}$ are only different from the Hankel singular values σ_i if there are repeated Hankel singular values. If all Hankel singular values are distinct one may write the bound as $2\sum_{i=r+1}^n \sigma_i$; in fact this latter bound is clearly always valid, but more conservative in the case of repeated singular values.

Note that even if the realization is not brought into balanced form, the Hankel singular values may be computed at little extra cost. This is why this bound is called *pre-computable*, in contrast with a bound that would require a number of operations in the same order of magnitude as computing the actual error. Such a bound is not of much use, since calculation of the real error $\|G(s) - G_r(s)\|_\infty$, which is of course the tightest conceivable bound, would be preferable then. The concept pre-computability is of course somewhat vague, but is important to keep in mind that a bound on the error is only useful to the extent that it can be found significantly faster than the real error.

If the original system $G(s)$ is stable and minimal, and if $\sigma_r > \sigma_{r+1}$, then the reduced-order system $G_r(s)$ of order r is also stable and minimal.

When there are repeated Hankel singular values, the balanced realization is non-unique, and this non-uniqueness can be exploited to construct a realization that yields stable reduced order systems when truncated [216].

Balanced Truncation is often criticized, as it does not optimize any known norm of the error[1]. For control design however, where the optimization is often a complicated function reflecting all the specifications, it is not always obvious which norm of the modeling error should be minimized, so it can be argued that this is not a great disadvantage. On the other hand, the pre-computable H_∞ bound is useful for stability and performance guarantees, using the robustness results of Chapter 2. Moreover, it has been noted that BT achieves a 'happy balance' between H_2 and H_∞ [302].

A myriad of algorithms differ from BT only in the choice of the matrices P and Q to be balanced; we call them balancing algorithms. Some of them are treated or proposed in this dissertation : BCRA is merely BT applied to a controller, BCRAM is a popular modification [312] (see Section 3.5); the next section explains a frequency weighted balancing method, Chapter 4 contains two balancing techniques suited for relative error model reduction, and Chapter 7 offers a balancing scheme tailored for time domain weighting. Other methods elaborate balancing techniques for coprime factorizations [175, 177, 178]. Verriest proposed balancing the H_2 Riccatians (i.e. the solutions of the ARE's in LQG), which is labeled LQG-balancing [279, 280, 282, 283]; see also Jonckheere [146, 147]. The same idea was also pursued for the H_∞ Riccatians [198, 199]. Many other possibilities are found in [302, 102, 103].

3.1.2 Frequency Weighted Balanced Truncation

An approach that tries to match as closely as possible the closed-loop transfer matrix $H_{zw}(s)$, using frequency weighted model reduction with frequency weights $W_o(s) = P_{zu}(I-CP_{yu})^{-1}$ and $W_i(s) = (I-P_{yu}C)^{-1}P_{yw}$, was mentioned in Subsection 2.4.2; it was not specified there how one can do this frequency weighted model reduction. One method, of which the appeal lies primarily in its simplicity and effectiveness in practice, is Frequency Weighted Balanced Truncation (FWBT), proposed by Enns [93, 94, 20]. We briefly recapitulate this scheme for continuous-time systems; the procedure is actually identical for discrete-time systems, except for the computation of the gramians, which is of course done using discrete Lyapunov equations in that case.

[1] Note that in some cases, Balanced Truncation can be shown to be Hankel-norm optimal, see [200, 201, 202].

We wish to approximate a stable system $G(s) \stackrel{\triangle}{=} [A, B, C, D]$ by a reduced order system $G_r(s) = [A_r, B_r, C_r, D_r]$ so that $W_i(s)(G(s)-G_r(s))W_o(s)$ is made small in some sense, where $W_i(s) \stackrel{\triangle}{=} [A_i, B_i, C_i, D_i]$ and $W_o(s) \stackrel{\triangle}{=} [A_o, B_o, C_o, D_o]$ are stable and causal invertible frequency weights of order n_i and n_o. Often one is interested in a small frequency weighted H_∞ norm $\|W_i(s)(G(s)-G_r(s))W_o(s)\|_\infty$, but other norms may be considered as well, e.g. the H_2 norm; like ordinary Balanced Truncation, the method about to be presented does not minimize such a norm, but uses some heuristics to come up with a small frequency weighted error.

If one balances the controllability gramian $\tilde{P}$ and observability gramian $\tilde{Q}$ of

$$W_o(s)G(s)W_i(s) = \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right] = \left[\begin{array}{ccc|c} A_o & B_oC & B_oDC_i & B_oDD_i \\ 0 & A & BC_i & BD_i \\ 0 & 0 & A_i & B_i \\ \hline C_o & D_oC & D_oDC_i & D_oDD_i \end{array}\right],$$

each component of the resulting state vector is a linear combination of the state vectors x_o, x and x_i of $W_o(s)$, $G(s)$ and $W_i(s)$; for model reduction, however, a realization is needed with which one can evaluate how observable and controllable the states of $G(s)$ are in that realization. Now let

$$\tilde{P} = \begin{bmatrix} P_o & P_{og} & P_{oi} \\ P_{og}^T & P_g & P_{gi} \\ P_{oi}^T & P_{gi}^T & P_i \end{bmatrix} \quad \text{and} \quad \tilde{Q} = \begin{bmatrix} Q_o & Q_{og} & Q_{oi} \\ Q_{og}^T & Q_g & Q_{gi} \\ Q_{oi}^T & Q_{gi}^T & Q_i \end{bmatrix}.$$

Restricting the admissible transformation matrices to those of the form

$$T = \begin{bmatrix} I_{n_o} & 0 & 0 \\ 0 & T_g & 0 \\ 0 & 0 & I_{n_i} \end{bmatrix},$$

where $T_g \in \mathbb{R}^{n\times n}$, one can no longer make $\tilde{P}$ and $\tilde{Q}$ diagonal or even equal, but T_g may be chosen such that $T_gP_gT_g^T = T_g^{-T}Q_gT_g^{-1} = \Sigma_g = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ (where of course σ_i is no longer a Hankel singular value of $G(s)$).
It may be argued that the 'frequency weighted Hankel singular values' σ_i are again a measure of how controllable the i-th state component is from the filter input, since still $E(x(t)x(t)^T) = P_g$ if the input of $W_i(s)$ is driven with unit variance white noise. Also, σ_i indicates how observable the corresponding state is at the filtered output, since the energy of an initial condition response $y(t)$, for $x(0) = x_0$, is $\int_0^\infty y^Tydt = x_0^TQ_gx_0$. Therefore Enns proposed to discard the

states having the smallest σ_i, i.e. to truncate the frequency weighted balanced realization of $G(s)$.

There is no need to compute $\tilde{P}$ and $\tilde{Q}$ completely, since the controllability gramian of $\left[\begin{array}{c|c} \tilde{A}_i & \tilde{B}_i \\ \hline \tilde{C}_i & \tilde{D}_i \end{array}\right] = \left[\begin{array}{cc|c} A & BC_i & BD_i \\ 0 & A_i & B_i \\ \hline C & DC_i & DD_i \end{array}\right]$ is $\left[\begin{array}{cc} P & P_{gi} \\ P_{gi}^T & P_i \end{array}\right]$,
and $\left[\begin{array}{c|c} \tilde{A}_o & \tilde{B}_o \\ \hline \tilde{C}_o & \tilde{D}_o \end{array}\right] = \left[\begin{array}{cc|c} A_o & B_oC & B_oD \\ 0 & A & B \\ \hline C_o & D_oC & D_oD \end{array}\right]$ has $\left[\begin{array}{cc} Q_o & Q_{og} \\ Q_{og}^T & Q \end{array}\right]$ for observability gramian.

A frequency domain integral for P_g and Q_g may be derived using

$$\begin{aligned}
(sI - \tilde{A}_i)^{-1}\tilde{B}_i &= \left[\begin{array}{cc} sI - A & -BC_i \\ 0 & sI - A_i \end{array}\right]^{-1} \left[\begin{array}{c} BD_i \\ B_i \end{array}\right] \\
&= \left[\begin{array}{cc} (sI - A)^{-1} & (sI - A)^{-1}BC_i(sI - A_i)^{-1} \\ 0 & (sI - A_i)^{-1} \end{array}\right] \left[\begin{array}{c} BD_i \\ B_i \end{array}\right] \\
&= \left[\begin{array}{c} (sI - A)^{-1}B(D_i + C_i(sI - A_i)^{-1}B_i) \\ (sI - A_i)^{-1}B_i \end{array}\right] :
\end{aligned}$$

$$P_g = \int (sI - A)^{-1}BW_i(s)W_i^T(-s)B^T(-sI - A^T)^{-1}ds \ ,$$

where the integral is taken over the imaginary axis. Analogously one shows that

$$Q_g = \int (-sI - A^T)^{-1}C^TW_o^T(-s)W_o(s)C(sI - A)^{-1}ds \ .$$

It follows that P_g and Q_g, and consequently also the reduced order system $G_r(s)$ resulting from the procedure, do not depend on the actual realization of $W_i(s)$ and $W_o(s)$. This property is of course essential for any sensible frequency weighted model reduction method.

The H_∞ norm and the H_2 norm of a stable system $G(s)$ do not change if the system is multiplied on the left and/or right by an inner system, (i.e. $\|G(s)\|_\infty = \|G_1(s)G(s)G_2(s)\|_\infty$, if $G_1(s)$, $G_2(s)$ are inner). Therefore a natural feature of a frequency weighted model reduction procedure should be that the input weight $W_i(s)$ may be replaced by any stable right spectral factor $W_{if}(s)$ of $W_i(s)W_i^T(-s)$, i.e. such that $W_{if}(s)W_{if}^T(-s) = W_i(s)W_i^T(-s)$, and analogously that the output weight $W_o(s)$ may be replaced with any left spectral factor of $W_o^T(-s)W_o(s)$, without changing the resulting reduced-order system. It is easily seen that FWBT has this property, by inspection of the integrals above.

A simple H_∞ error bound as for unweighted balanced truncation is not available. Enns proposed a bound [93], which contained an unknown and not pre-computable factor though, so that it is of little use. Recently Kim and Anderson [155] published a bound for the weighted H_∞ error $\|W_o(s)(G(s)-G_r(s))W_i(s)\|_\infty$. Evaluation of this bound involves the computation of several other H_∞ norms, but of systems of lower order. As computing an H_∞ norm $\|H(s)\|_\infty$ is an iterative process requiring several tens of n^3 flops, where n is the order of the system $H(s)$, this bound may represent a valuable tool. However, the complexity of the pre-computation is not lower than that of computing the error itself, and it is not sufficiently known at present how tight this bound is.

Enns showed that the reduced model is generically (i.e. when the reduced-order model is minimal) stable when $W_o(s) = I$ or $W_i(s) = I$, i.e. the case with one-sided weighting. Zhou et al. [320] proved that *'generically'* may be replaced with *'always'* in the discrete-time case. Unfortunately, when the weighting is two-sided, stability of the reduced-order model can no longer be guaranteed.

3.2 Closed-Loop Balanced Truncation

That closed-loop considerations have to be taken into account while designing a controller of reduced order has been pointed out before. A reduced order model may mimic the input-output behavior of a plant rather well, but a control law based on this model may cause unsatisfactory closed-loop behavior because the seemingly insignificant discarded dynamics have greater impact on the closed loop than on the open-loop transfer; this phenomenon is called *spillover*. A similar thing may happen when the controller is obtained by approximating a controller, stabilizing the full order plant in a satisfactory way, by a low-order one. Many authors have contributed model reduction methods more suitable with respect to control design [20, 177, 226, 227, 147, 199, 93, 94, 301].

De Villemagne and Skelton [80] proposed an approach focused on putting the closed-loop controllability gramian in a form that enables one to see which combinations of states are insignificant, but does not employ balancing.
This inspired the Closed-Loop Balanced Truncation (CLBT) approach, an intuitive but remarkably useful method of controller reduction proposed recently by Wortelboer e.a. [65, 302, 305]. In Section 3.3, we show that this method can also be viewed as a frequency weighted model reduction method, where the

weights are chosen so as to retain closed-loop performance. In Section 3.4, it is shown how significant savings can be made in the number of computations in the important special case of observer-based controllers. In this section, we recapitulate the idea of CLBT in the framework of the standard plant configuration.

Suppose one has a standard plant

$$P(s) = \begin{bmatrix} P_{zw}(s) & P_{zu}(s) \\ P_{yw}(s) & P_{yu}(s) \end{bmatrix} \triangleq \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$

and one has a controller $C(s) \triangleq [A_c, B_c, C_c, D_c]$ that yields a satisfactory closed-loop transfer matrix $H_{zw} = P_{zw} + P_{zu}C(I - P_{yu}C)^{-1}P_{yw}$. Let n be the order of the standard plant $P(s)$, n_c the order of $C(s)$, and r the desired reduced order. Note that we may assume without loss of generality that $D_{yu} = 0$, since if $P_{yu}(s)$ is not strictly proper, the problem may always be transformed into another one where $P_{yu}(s)$ is strictly proper by means of a loop transformation [111]. In such a case, a state space representation for H_{zw} is

$$H_{zw}(s) \triangleq \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right] = \left[\begin{array}{cc|c} A + B_u D_c C_y & B_u C_c & B_w + B_u D_c D_{yw} \\ B_c C_y & A_c & B_c D_{yw} \\ \hline C_z + D_{zu} D_c C_y & D_{zu} C_c & D_{zw} + D_{zu} D_c D_{yw} \end{array}\right] . \tag{3.1}$$

For this realization of the closed-loop system, the controllability gramian $\tilde{P} \triangleq \begin{bmatrix} P_p & P_{pc} \\ P_{pc}^T & P_c \end{bmatrix}$ and observability gramian $\tilde{Q} \triangleq \begin{bmatrix} Q_p & Q_{pc} \\ Q_{pc}^T & Q_c \end{bmatrix}$ are computed from $\tilde{A}\tilde{P} + \tilde{P}\tilde{A}^T + \tilde{B}\tilde{B}^T = 0$ and $\tilde{A}^T\tilde{Q} + \tilde{Q}\tilde{A} + \tilde{C}^T\tilde{C} = 0$.

In order to reduce the order of the controller, while changing the closed-loop transfer function as little as possible, one could think of balancing $\tilde{P}$ and $\tilde{Q}$; but as for FWBT, this does not make sense as it mixes up controller and plant states. Instead, we may perform balancing on the controller $[A_c, B_c, C_c, D_c]$ using P_c and Q_c, the parts of these gramians relevant to the controller states; i.e. we apply a similarity transformation T_c to $[A_c, B_c, C_c, D_c]$ such that $T_c P_c T_c^T = T_c^{-T} Q_c T_c^{-1} = \Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_{n_c})$. This is the idea of CLBT [65, 302]. In this realization, σ_i may be interpreted both as the average power injected in the i-th controller state component by a white Gaussian noise disturbance w of unit variance, and as the contribution of the i-th controller state to the output z. Thus, the 'closed-loop Hankel singular values' σ_i are a measure of how controllable and observable these controller states are, or in other words, how much they contribute to the closed-loop input-output behavior of the closed-loop transfer matrix. An obvious idea is to discard those controller states corresponding to the smallest closed-loop Hankel singular values first. On the other hand, one should not forget that since the closed-loop gramians $\tilde{P}$ and $\tilde{Q}$ are not made diagonal, but are transformed into $\begin{bmatrix} P_p & P_{pc}T_c^T \\ T_cP_{pc}^T & \Sigma \end{bmatrix}$ and $\begin{bmatrix} Q_p & Q_{pc}T_c^{-1} \\ T_c^{-T}Q_{pc}^T & \Sigma \end{bmatrix}$; i.e. the controller states and the plant states are still correlated, and thus, truncation of the controller states is harder to justify than in ordinary Balanced Truncation.

While the procedure may be rather intuitive and ad hoc, and cannot guarantee stability of the reduced controller, which is not what we are after anyway, or of the closed-loop system with the reduced order controller, which would be a result too beautiful to be true, it provides a useful reduction method in practice, and compares very favorably with many well-known controller reduction methods with respect to preservation of closed-loop performance. Of course these statements are hard to substantiate in general; we refer to the examples for support of them.

Again, this method works for discrete-time systems as well, with a modification only to the Lyapunov equations.

Note that a somewhat similar, more sophisticated, but not necessarily superior technique is one recently put forward by Chang e.a. [66, 67].

3.3 Equivalence with Frequency Weighted Balanced Truncation

To provide some further motivation for CLBT, we proceed to show in this section that it is mathematically equivalent to the approach of Section 3.1.2, but numerically less costly.

Recall $W_o(s) \triangleq P_{zu}(I - CP_{yu})^{-1}$ and $W_i(s) \triangleq (I - P_{yu}C)^{-1}P_{yw}$ have realizations :

$$W_o(s) = \left[\begin{array}{cc|c} A + B_uD_cC_y & B_uC_c & B_u \\ B_cC_y & A_c & 0 \\ \hline C_z + D_{zu}D_cC_y & D_{zu}C_c & D_{zu} \end{array}\right],$$

$$W_i(s) = \left[\begin{array}{cc|c} A + B_u D_c C_y & B_u C_c & B_w + B_u D_c D_{yw} \\ B_c C_y & A_c & B_c D_{yw} \\ \hline C_y & 0 & D_{yw} \end{array}\right].$$

Theorem 3.1
Let $C(s)$ *be a stable stabilizing controller of order* n_c *for standard plant* $P(s)$*, let* $r \leq n_c$ *and* $W_o(s)$ *and* $W_i(s)$ *as given above;*
Let $C_r^{(w)}(s)$ *be the reduced order controller of order* r *obtained by Frequency Weighted Balanced Truncation with* $W_o(s)$ *as output weight and* $W_i(s)$ *as input weight;*
Let $C_r^{(c)}(s)$ *be the reduced order controller of order* r *obtained by Closed-Loop Balanced Truncation;*

Then $$C_r^{(w)}(s) = C_r^{(c)}(s) .$$

Proof : FWBT with weights W_o and W_i first computes the controllability gramian $\bar{P}$ of

$$C(s)W_i(s) = \left[\begin{array}{ccc|c} A + B_u D_c C_y & B_u C_c & 0 & B_w + B_u D_c D_{yw} \\ B_c C_y & A_c & 0 & B_c D_{yw} \\ B_c C_y & 0 & A_c & B_c D_{yw} \\ \hline D_c C_y & 0 & C_c & D_c D_{yw} \end{array}\right]$$

and the observability gramian $\bar{Q}$ of

$$W_o(s)C(s) = \left[\begin{array}{ccc|c} A + B_u D_c C_y & B_u C_c & B_u C_c & B_u D_c \\ B_c C_y & A_c & 0 & 0 \\ 0 & 0 & A_c & B_c \\ \hline C_z + D_{zu} D_c C_y & D_{zu} C_c & D_{zu} C_c & D_{zu} D_c \end{array}\right],$$

balances the lower right $n_c \times n_c$ blocks of $\bar{P}$ and $\bar{Q}$, and truncates the controller accordingly. Replacing the state x_i of the realization of $C(s)W_i(s)$ by Tx_i, where $T \triangleq \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & I & I \end{bmatrix}$, and the state x_o of the realization of $W_o(s)C(s)$ by $(T^{-1})^T x_o$, yields two new realizations :

$$C(s)W_i(s) = \left[\begin{array}{ccc|c} A + B_u D_c C_y & B_u C_c & 0 & B_w + B_u D_c D_{yw} \\ B_c C_y & A_c & 0 & B_c D_{yw} \\ 0 & 0 & A_c & 0 \\ \hline D_c C_y & C_c & C_c & D_c D_{yw} \end{array}\right],$$

which is seen, by inspection of (3.1), to have $\begin{bmatrix} \tilde{P} & 0 \\ 0 & 0 \end{bmatrix}$ for controllability gramian, and

$$W_o(s)C(s) = \left[\begin{array}{ccc|c} A + B_u D_c C_y & B_u C_c & 0 & B_u D_c \\ B_c C_y & A_c & 0 & B_c \\ 0 & 0 & A_c & B_c \\ \hline C_z + D_{zu} D_c C_y & D_{zu} C_c & 0 & D_{zu} D_c \end{array}\right],$$

which has $\begin{bmatrix} \tilde{Q} & 0 \\ 0 & 0 \end{bmatrix}$ for observability gramian. Therefore

$$\begin{aligned} \bar{P} &= T^T \begin{bmatrix} \tilde{P} & 0 \\ 0 & 0 \end{bmatrix} T = \begin{bmatrix} P_p & P_{pc} & P_{pc} \\ P_{pc}^T & P_c & P_c \\ P_{pc}^T & P_c & P_c \end{bmatrix}, \\ \text{and} \quad \bar{Q} &= T^T \begin{bmatrix} \tilde{Q} & 0 \\ 0 & 0 \end{bmatrix} T = \begin{bmatrix} Q_p & Q_{pc} & Q_{pc} \\ Q_{pc}^T & Q_c & Q_c \\ Q_{pc}^T & Q_c & Q_c \end{bmatrix}. \end{aligned}$$

Hence, both in FWBT with the weightings above and in CLBT, the same gramians P_c and Q_c are balanced, and the reduced order controller is obtained in both cases by applying the same similarity transformation to $[A_c, B_c, C_c, D_c]$ and subsequent truncation. Therefore, $C_r^{(w)}(s) = C_r^{(c)}(s)$. □

Note that this theorem and its proof still hold for discrete-time systems, as in that case a similarity transformation gives rise to the same contragredient transformation on the gramians.

The theorem shows that the two methods are mathematically equivalent; but as the size of the gramians to be computed in CLBT is only $(n + n_c) \times (n + n_c)$ compared to $(n+2n_c) \times (n+2n_c)$ in FWBT, the former is more efficient, as well as numerically more reliable. Moreover, with CLBT there are no restrictions on the poles of the controller : these may be unstable or even lie on the imaginary axis.

We have stated above that this equivalence is a further motivation for CLBT, for it shows that CLBT aims at preserving a first order approximation of the closed-loop performance error. But this interpretation also shows the shortcomings of the method : as the frequency weighted modeling error is a first-order approximation of the perturbation of the closed-loop transfer H_{zw}, precisely for the very low orders, this approximation may deviate considerably from the real error, indicating that the method may not be the very best one whenever the lowest stabilizing order is desired. On the other hand, for orders somewhat higher than this lowest admissible order, the method is likely to be a very good reduction method with respect to closed-loop performance. These suspicions are corroborated by the examples below.

3.4 CLBT for observer-based controllers

In many control design techniques, a controller is found as the combination of a state feedback controller and a state observer, e.g. when using pole placement or in LQG design [86]. For this type of controllers, the order of the controller equals the order of the plant n, hence the procedure above would normally require the solution of two $2n \times 2n$ Lyapunov equations. However, in this case, the computation of the gramians P_c and Q_c may be simplified, as shown below.

3.4.1 Continuous-time systems

Suppose the state equation of the state estimator is given by $\dot{x}_c = Ax_c + B_u u + K_e(C_y x_c - y)$ and the state feedback gain is K_r, then the controller has state space representation $C(s) = [A + B_u K_r + K_e C_y, -K_e, K_r, 0]$, and for the closed-loop system one finds :

$$H_{zw}(s) = \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right] = \left[\begin{array}{cc|c} A & B_u K_r & B_w \\ -K_e C_y & A + B_u K_r + K_e C_y & -K_e D_{yw} \\ \hline C_z & D_{zu} K_r & D_{zw} \end{array}\right] .$$

Defining $T \triangleq \begin{bmatrix} I & I \\ 0 & I \end{bmatrix}$, one has

$$\begin{aligned} H_{zw}(s) &= \left[\begin{array}{c|c} \tilde{A}_1 & \tilde{B}_1 \\ \hline \tilde{C}_1 & \tilde{D}_1 \end{array}\right] \triangleq \left[\begin{array}{c|c} T^{-1}\tilde{A}T & T^{-1}\tilde{B} \\ \hline \tilde{C}T & D_{zw} \end{array}\right] \\ &= \left[\begin{array}{cc|c} A + K_e C_y & 0 & B_w + K_e D_{yw} \\ -K_e C_y & A + B_u K_r & -K_e D_{yw} \\ \hline C_z & C_z + D_{zu} K_r & D_{zw} \end{array}\right] \\ &= \left[\begin{array}{c|c} \tilde{A}_2 & \tilde{B}_2 \\ \hline \tilde{C}_2 & \tilde{D}_2 \end{array}\right] \triangleq \left[\begin{array}{c|c} T^{-T}\tilde{A}T^T & T^{-T}\tilde{B} \\ \hline \tilde{C}T^T & D_{zw} \end{array}\right] \end{aligned}$$

$$= \left[\begin{array}{cc|c} A + B_u K_r & B_u K_r & B_w \\ 0 & A + K_e C_y & -B_w - K_e D_{yw} \\ \hline C_z + D_{zu} K_r & D_{zu} K_r & D_{zw} \end{array}\right] .$$

Now if $\tilde{A}_1 \tilde{P}_1 + \tilde{P}_1 \tilde{A}_1^T + \tilde{B}_1 \tilde{B}_1^T = 0$ and $\tilde{A}_2^T \tilde{Q}_2 + \tilde{Q}_2 \tilde{A}_2 + \tilde{C}_2^T \tilde{C}_2 = 0$, where $\tilde{P}_1$ and $\tilde{Q}_2$ are partitioned in the same manner as before :

$$\tilde{P}_1 \triangleq \begin{bmatrix} P_{p1} & P_{pc1} \\ P_{pc1}^T & P_{c1} \end{bmatrix} \text{ and } \tilde{Q}_2 \triangleq \begin{bmatrix} Q_{p2} & Q_{pc2} \\ Q_{pc2}^T & Q_{c2} \end{bmatrix} ;$$

then $\tilde{P} = T \tilde{P}_1 T^T$ and $\tilde{Q} = T^{-1} \tilde{Q}_2 T^{-T}$, and it is easily seen that $P_c = P_{c1}$ and $Q_c = Q_{c2}$. Therefore CLBT is performed by solving the following Sylvester equations consecutively for P_{p1}, P_{pc1} and P_{c1} :

$$\begin{aligned} &(A + K_e C_y) P_{p1} + P_{p1} (A + K_e C_y)^T \\ &\qquad + (B_w + K_e D_{yw})(B_w + K_e D_{yw})^T = 0 \end{aligned} \tag{3.2}$$

$$\begin{aligned} &(A + K_e C_y) P_{pc1} + P_{pc1} (A + B_u K_r)^T \\ &\qquad - P_{p1} C_y^T K_e^T - (B_w + K_e D_{yw})(K_e D_{yw})^T = 0 \end{aligned} \tag{3.3}$$

$$\begin{aligned} &(A + B_u K_r) P_{c1} + P_{c1} (A + B_u K_r)^T \\ &\qquad - K_e C_y P_{pc1} - P_{pc1}^T C_y^T K_e^T + K_e D_{yw} (K_e D_{yw})^T = 0 \end{aligned} \tag{3.4}$$

and the dual Sylvester equations consecutively for Q_{p2}, Q_{pc2} and Q_{c2} :

$$\begin{aligned} &(A + B_u K_r)^T Q_{p2} + Q_{p2} (A + B_u K_r) \\ &\qquad + (C_z + D_{zu} K_r)^T (C_z + D_{zu} K_r) = 0 \end{aligned} \tag{3.5}$$

$$\begin{aligned} &(A + B_u K_r)^T Q_{pc2} + Q_{pc2} (A + K_e C_y) \\ &\qquad + Q_{p2} B_u K_r + (C_z + D_{zu} K_r)^T D_{zu} K_r = 0 \end{aligned} \tag{3.6}$$

$$\begin{aligned} &(A + K_e C_y)^T Q_{c2} + Q_{c2} (A + K_e C_y) \\ &\qquad + K_r^T B_u^T Q_{pc2} + Q_{pc2}^T B_u K_r + K_r^T D_{zu}^T D_{zu} K_r = 0 \, , \end{aligned} \tag{3.7}$$

which yields $P_{c1} = P_c$ and $Q_{c2} = Q_c$.

Remark that these computations may be done most efficiently if $A + B_u K_r$ and $A + K_e C_y$ are first put in Schur form, since then, solving the six Sylvester equations above requires only two Schur decompositions instead of six. The operation counts below are based on the assumption that the computation of a Schur decomposition costs $25n^3$ flops; this value is of course only approximate [116, p. 380]. $2n^3$ flops are needed to solve a Sylvester equation once the coefficient matrices have been put in Schur form (n^3 for a Lyapunov equation). Counting this way, and noting that four of the six Sylvester equations are actually Lyapunov equations, P_{c1} and Q_{c2} may be computed using about $2(25n^3 + 2n^3) + 4n^3 = 58n^3$ flops, whereas the 'vanilla' approach of computing the two $2n \times 2n$ closed-loop gramians $\tilde{P}$ and $\tilde{Q}$ requires $25(2n)^3 + 2(2n)^3 = 216n^3$ flops.

Note furthermore that the computation is even less costly if the eigenvalue decompositions $A + B_u K_r = T_r^{-1} \Lambda_r T_r$ and $A + K_e C_y = T_e^{-1} \Lambda_e T_e$ are available, as will often be the case when K_r and K_e have been obtained using a pole placement technique; then solving each of the six equations above is done in $O(n^2)$ flops. Indeed, these represent similarity transformations that bring $A + B_u K_r$ and $A + K_e C_y$ into diagonal form, so that the Sylvester equations are easily seen to be solved in $O(n^2)$ flops (note that it is trivial to bring $A + B_u K_r$ and $A + K_e C_y$ into real quasi-diagonal form if one wishes to stick to real computations). The price to be paid for this, however, is a loss of numerical reliability, since these transformations are no longer orthogonal.

A word of caution is in order with respect to the fact that only two Schur decompositions are needed to solve equations (3.2), (3.3), (3.4), (3.5), (3.6), (3.7). In principle any Schur decomposition of $A + B_u K_r$ and $A + K_e C_y$ may be used for solving these equations, but for large Sylvester or Lyapunov equations the eigenvalue ordering of the Schur matrices may have an important impact on the magnitude of the numerical error. The QR algorithm naturally yields an ordering with the eigenvalues of the Schur matrix going from large absolute values in the left upper corner to small ones in the right lower corner; this is numerically beneficial for solving eqs. (3.5), (3.6) and (3.7), but in equal degree numerically adverse for eqs. (3.2), (3.3) and(3.4), if the same Schur decompositions are used without any reordering. Therefore, an inverse ordering, which is done using orthogonal transformations only, may be advisable for eqs. (3.2), (3.3) and (3.4), but requires about extra $6n^3$ operations [116, 259], taking the cost to compute P_c and Q_c to $64n^3$ operations.
Note, however, that the same remark applies to the vanilla approach of computing $\tilde{P}$ and $\tilde{Q}$, where also only the Schur decomposition of $\tilde{A}$ were computed and used for solving both Lyapunov equations. A reordering is equally advisable here, but requires $6(2n)^3 = 48n^3$ operations, adding up to a total cost of $264n^3$ operations.
To avoid the trouble of reordering the Schur decompositions, one might also opt

to recompute the Schur decomposition of transposes : in this case our scheme has an operation count of $108n^3$, compared to $416n^3$ for the approach computing $\tilde{P}$ and $\tilde{Q}$. So whatever options are taken, the scheme presented here finds P_c and Q_c using about one fourth of the number of flops needed for the vanilla approach.

With respect to computer memory needed, it is advantageous that the six Sylvester equations above may be solved consecutively, so that all one needs to store into memory at one time is three $n \times n$ matrices (four when solving the Sylvester equations).

Questions may arise as to whether the procedure given above is as numerically reliable as the normal approach of finding a Schur decomposition of $\tilde{A}$; after all, the similarity transformations using the matrix T constitute non-orthogonal operations as well. But this point is not really valid : the matrices A_i, B_i, C_i, $i = 1, 2$, are not computed as $T^{-1}\tilde{A}T$, etc., but immediately as $\begin{bmatrix} A + B_u K_r & B_u K_r \\ 0 & A + K_e C_y \end{bmatrix}$, etc. Therefore, it would be hard to determine a priori whether e.g. the equation $\tilde{A}_1 \tilde{P}_1 + \tilde{P}_1 \tilde{A}_1^T + \tilde{B}_1 \tilde{B}_1^T = 0$ has worse or better condition than the equation $\tilde{A}\tilde{P} + \tilde{P}\tilde{A}^T + \tilde{B}\tilde{B}^T = 0$.

Furthermore, the use of these Sylvester equations may easily be avoided, if one would like to do so. Indeed, the usual way of solving a Lyapunov equation $AP + PA^T + BB^T = 0$ is to bring A into (quasi-)triangular form by means of a Schur decomposition, and to solve the transformed Lyapunov equation by means of forward or backward substitution [116, 38]. The splitting of a $2n \times 2n$ Lyapunov equation into three Sylvester equations amounts to a block version of this substitution process. This artifice is not really necessary :
if $U_r^T(A + B_u K_r)U_r$ and $U_e^T(A + K_e C_y)U_e$ are in real Schur form,
then $\begin{bmatrix} U_e & 0 \\ 0 & U_r \end{bmatrix}^T \tilde{A}_1 \begin{bmatrix} U_e & 0 \\ 0 & U_r \end{bmatrix}$ and $\begin{bmatrix} U_r & 0 \\ 0 & U_e \end{bmatrix}^T \tilde{A}_2 \begin{bmatrix} U_r & 0 \\ 0 & U_e \end{bmatrix}$ will also have the quasi-triangular form necessary for starting the substitution process. The only advantage of this splitting into three equations is the memory argument mentioned before; the operation count is identical for both ways, and with respect to accuracy, no remarkable differences were recorded. It is possible that the $2n \times 2n$ quasi-triangular form constructed here needs reordering over the full diagonal for accuracy; this is certainly a deviation from the block version, but is still cheaper than the standard approach of computing a (reordered) Schur decomposition of the matrix $\tilde{A}$.

A final word of comfort about the accuracy business is that one often need not worry too much about it : after all, one is interested only in the dominant eigenspaces of P_cQ_c, and the accumulated errors will usually show in the discarded dynamics, except for systems of such large order that they are beyond the scope of dense methods.

Once P_{c1} and Q_{c2} are computed by consecutively solving equations (3.2), (3.3), (3.4) and (3.5), (3.6), (3.7), the reduced order controller may be found, as in Subsection 3.1.1, without actually balancing P_{c1} and Q_{c2}, by using Safonov's Schur method [234]. The resulting reduced order controller may be considered as the series connection of a reduced order observer $[L(A+BK_r+K_eC)R, -LK_e, R, 0]$ and the optimal state feedback gain K_r.

3.4.2 Discrete-time systems

Once more, in the discrete-time case, only the Lyapunov equations change; from $\tilde{P}_1 = \tilde{A}_1\tilde{P}_1\tilde{A}_1^T + \tilde{B}_1\tilde{B}_1^T$ and $\tilde{Q}_2 = \tilde{A}_2^T\tilde{Q}_2\tilde{A}_2 + \tilde{C}_2^T\tilde{C}_2$, it is found that equations (3.2), (3.3), (3.4) and (3.5), (3.6), (3.7) should be replaced by

$$\begin{aligned}
P_{p1} &= (A+K_eC_y)P_{p1}(A+K_eC_y)^T + (B_w+K_eD_{yw})(B_w+K_eD_{yw})^T \\
P_{pc1} &= (A+K_eC_y)P_{pc1}(A+B_uK_r)^T \\
&\quad - (A+K_eC_y)P_{p1}C_y^TK_e^T - (B_w+K_eD_{yw})(K_eD_{yw})^T \\
P_{c1} &= (A+B_uK_r)P_{c1}(A+B_uK_r)^T - K_eC_yP_{pc1}(A+B_uK_r)^T \\
&\quad - (A+B_uK_r)P_{pc1}^TC_y^TK_e^T + K_eC_yP_{p1}^TC_y^TK_e^T + K_eD_{yw}(K_eD_{yw})^T
\end{aligned}$$

and

$$\begin{aligned}
Q_{p2} &= (A+B_uK_r)^TQ_{p2}(A+B_uK_r) + (C_z+D_{zu}K_r)^T(C_z+D_{zu}K_r) \\
Q_{pc2} &= (A+B_uK_r)^TQ_{pc2}(A+K_eC_y) \\
&\quad + (A+B_uK_r)^TQ_{p2}B_uK_r + (C_z+D_{zu}K_r)^TD_{zu}K_r \\
Q_{c2} &= (A+K_eC_y)^TQ_{c2}(A+K_eC_y) + K_r^TB_u^TQ_{pc2}(A+K_eC_y) \\
&\quad + (A+K_eC_y)^TQ_{pc2}^TB_uK_r + K_r^TB_u^TQ_{pc2}B_uK_r + K_r^TD_{zu}^TD_{zu}K_r \ ,
\end{aligned}$$

which yields $P_{c1} = P_c$ and $Q_{c2} = Q_c$.

3.5 The optimal H_2 controller case

In this section, we shall prove that when used on an optimal H_2 controller, CLBT is equivalent to BCRAM, an algorithm proposed by Yousuff and Skelton [312] (BCRAM is an acronym for Balanced Controller Reduction Algorithm Modified). This algorithm, as the keen reader may have guessed from the nomenclature, is a modification of another one, called Balanced Controller Reduction

Algorithm (BCRA), discussed in the same paper, which amounts to merely applying Moore's Balanced Truncation to the controller $C(s)$ [195]. For an observer-based controller $C(s) = [A + B_uK_r + K_eC_y, K_r, -K_e, 0]$, the gramians to be used for balancing are computed in this case from

$$(A + B_uK_r + K_eC_y)P + P(A + B_uK_r + K_eC_y)^T + K_eD_{yw}D_{yw}^TK_e^T = 0$$

$$(A + B_uK_r + K_eC_y)^TQ + Q(A + B_uK_r + K_eC_y) + K_r^TD_{zu}^TD_{zu}K_r = 0$$

for continuous systems, and from

$$(A + B_uK_r + K_eC_y)P(A + B_uK_r + K_eC_y)^T - P + K_eD_{yw}D_{yw}^TK_e^T = 0$$

$$(A + B_uK_r + K_eC_y)^TQ(A + B_uK_r + K_eC_y) - Q + K_r^TD_{zu}^TD_{zu}K_r = 0$$

for discrete systems.
The only subtlety to be noted here is the scaling of controller inputs and outputs inherent in replacing $K_eK_e^T$ by $K_eD_{yw}D_{yw}^TK_e^T$ and $K_r^TK_r$ by $K_r^TD_{zu}^TD_{zu}K_r$; i.e. the measurements y are first scaled in such a way that the measurement noise has a 'white component' whose covariance matrix is a unit matrix, and the control inputs are scaled so that they are given equal importance in the performance criterion. The authors note that BCRA has the favorable property of reducing non-minimal controllers to their minimal realizations, a property not shared by some other algorithms (e.g. LQG balanced truncation [147, 279]), but has the disadvantage that the controller is required to be stable. They propose the following modification to BCRA (for the case that the controller is H_2-optimal, but evidently the method may be used for reducing any observer-based controller) : compute P_m and Q_m from

$$(A + B_uK_r)P_m + P_m(A + B_uK_r)^T + K_eD_{yw}D_{yw}^TK_e^T = 0 \tag{3.8}$$

$$(A + K_eC_y)^TQ_m + Q_m(A + K_eC_y) + K_r^TD_{zu}^TD_{zu}K_r = 0 \tag{3.9}$$

for continuous systems, or from

$$(A + B_uK_r)P_m(A + B_uK_r)^T - P_m + K_eD_{yw}D_{yw}^TK_e^T = 0 \tag{3.10}$$

$$(A + K_eC_y)^TQ_m(A + K_eC_y) - Q_m + K_r^TD_{zu}^TD_{zu}K_r = 0 \tag{3.11}$$

for discrete systems, and obtain the reduced order controller by balancing P_m and Q_m and truncating accordingly. This seemingly random change is motivated by the fact that the matrices $A+B_uK_r$ and $A+K_eC_y$ are stable by construction, and the following result from [298] :

- $(A + B_uK_r + K_eC_y, K_e)$ is controllable iff $(A + B_uK_r, K_e)$ is controllable.

- $(K_r, A + B_u K_r + K_e C_y)$ is observable iff $(K_r, A + K_e C_y)$ is observable.

Hence, this scheme may be used also for unstable controllers, and whenever the controller is non-minimal, it will certainly eliminate non-minimal modes first. However, no motivation with respect to performance considerations was given for this scheme, although good performance has often been recorded in comparisons on examples [68, 177]. In [177], a controller reduction method with emphasis on stability preservation proposed therein, labeled LCFW, was shown to be equivalent with BCRAM for LQG/LTR designed controllers. The structure of the closed-loop gramians for H_2-optimal control, noticed in [67], enables us to add the following result :

Theorem 3.2
Let $C(s)$ *be a CT* H_2*-optimal controller for a standard plant* $P(s)$*, where*

$$P(s) \triangleq \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$

and $D_{zu}^T D_{zu}$ *and* $D_{yw} D_{yw}^T$ *are full rank matrices.*
Let $C_r^{(c)}(s)$ *be the controller of order* r *obtained by CLBT reduction of* $C(s)$*, and let* $C_r^{(m)}(s)$ *be the controller of order* r *obtained by BCRAM reduction of* $C(s)$*,*

Then $$C_r^{(c)}(s) = C_r^{(m)}(s)\ .$$

Proof :
Define $R_u \triangleq (D_{zu}^T D_{zu})^{-1}$, $R_y \triangleq (D_{yw} D_{yw}^T)^{-1}$, $D_{zu}^\dagger \triangleq R_u D_{zu}^T$, and $D_{yw}^\dagger \triangleq D_{yw}^T R_y$. Let X_2 be the solution of the algebraic Riccati equation

$$\begin{aligned}(A - B_u D_{zu}^\dagger C_z)^T X &+ X(A - B_u D_{zu}^\dagger C_z) \\ &- X B_u R_u B_u^T X + C_z^T (I - D_{zu} R_u D_{zu}^T) C_z \ = \ 0\end{aligned}$$

such that $A - B_u D_{zu}^\dagger C_z - B_u R_u B_u^T X_2$ is stable, and let Y_2 be the solution of

$$\begin{aligned}(A - B_w D_{yw}^\dagger C_y) Y &+ Y (A - B_w D_{yw}^\dagger C_y)^T \\ &- Y C_y^T R_y C_y Y + B_w (I - D_{yw}^T R_y D_{yw}) B_w^T \ = \ 0\end{aligned}$$

such that $A - B_w D_{yw}^\dagger C_y - Y_2 C_y^T R_y C_y$ is stable.
Then it is a standard result [86] that $C(s)$ is obtained as an observer-based controller, with $K_r \triangleq -R_u B_u^T X_2 - D_{zu}^\dagger C_z$ and $K_e \triangleq -Y_2 C_y^T R_y - B_w D_{yw}^\dagger$. Plugging these expressions into equation (3.5), one obtains

$$0 = (A - B_u R_u B_u^T X_2 - B_u D_{zu}^\dagger C_z)^T Q_{pc2} + Q_{pc2} (A - B_u R_u B_u^T X_2 - B_u D_{zu}^\dagger C_z)$$

$$+ (C_z - D_{zu}R_uD_{zu}^TC_z - D_{zu}R_uB_u^TX_2)^T(C_z - D_{zu}R_uD_{zu}^TC_z - D_{zu}R_uB_u^TX_2)$$

Substituting X_2 for Q_{p2} yields the Riccati equation for X_2, so $Q_{p2} = X_2$. The expression $Q_{p2}B_uK_r + (C_z + D_{zu}K_r)^TD_{zu}K_r$ in equation (3.6) turns into

$$\begin{aligned} &-X_2B_uR_uB_u^TX_2 - X_2B_uD_{zu}^\dagger C_z \\ &\qquad + (C_z^T - C_z^TD_{zu}R_uD_{zu}^T)D_{zu}K_r + X_2B_uR_uD_{zu}^TD_{zu}(R_uB_u^TX_2 + D_{zu}^\dagger C_z) \\ &= \quad - X_2B_uR_uB_u^TX_2 - X_2B_uD_{zu}^\dagger C_z + X_2B_uR_uB_u^TX_2 + X_2B_uD_{zu}^\dagger C_z \;=\; 0\,, \end{aligned}$$

hence $(A + B_uK_r)^TQ_{pc2} + Q_{pc2}(A + K_eC_y) = 0$ and $Q_{pc2} = 0$. Thus equation (3.7) reads

$$(A + K_eC_y)^TQ_{c2} + Q_{c2}(A + K_eC_y) + K_r^TD_{zu}^TD_{zu}K_r = 0,$$

which is identical to equation (3.9), hence $Q_c = Q_{c2} = Q_m$.
Dual expression reveal that $P_{p1} = Y_2$, $P_{pc1} = 0$, and P_{c1} satisfies equation (3.8)

$$(A + B_uK_r)P_{p1} + P_{p1}(A + B_uK_r)^T + K_eD_{yw}D_{yw}^TK_e^T = 0$$

and hence $P_c = P_{c1} = P_m$. □

Corollary 3.1
The closed-loop gramians for an H_2-optimal design are given by

$$\tilde{P} = \begin{bmatrix} P_m + Y_2 & P_m \\ P_m & P_m \end{bmatrix} \quad \textit{and} \quad \tilde{Q} = \begin{bmatrix} Q_m + X_2 & -Q_m \\ -Q_m & Q_m \end{bmatrix}$$

Proof :
This follows from $\tilde{P}_1 = \mathrm{diag}(X_2, P_m)$ and $\tilde{Q}_2 = \mathrm{diag}(Y_2, Q_m)$, as shown in the previous proof, and $\tilde{P} = T\tilde{P}_1T^T$ and $\tilde{Q} = T^{-1}\tilde{Q}_2T^{-T}$ (see also [67]). □

The simplicity and brevity of the BCRAM procedure, compared with the six Sylvester equations to be solved for CLBT on observer-based controllers, together with the fact that the Riccatians X_2 and Y_2 are obtained during the H_2 control design stage, may perhaps lead to the belief that the former results in a much smaller operation count. This is in fact not true : in both cases the cost of the Schur decompositions of $A + B_uK_r$ and of $A + K_eC_y$ dominate the computation of P_c and Q_c, and only two of them are needed in both cases; the number of operations needed is only marginally smaller for BCRAM, at least if full advantage is taken of the observer-based structure. However, the BCRAM procedure is certainly less cumbersome. As no error can propagate from one Sylvester equation to the other as in the CLBT case, it seems reasonable to surmise that it will generally be subject to a smaller error. Such conjectures should be treated with much caution, however, and it is certainly possible to find examples where this is not the case.

Note that in [140], BCRAM was already found to be equivalent to a scheme coined AOPC_1, which amounts to taking the intermediate result after one step of an iterative H_2-optimal fixed-order controller reduction algorithm.

The equivalence between BCRAM and CLBT is lost for other types of controllers, e.g. those obtained by pole placement techniques; moreover the result cannot be extended to the discrete-time case.

3.6 Enns's scheme with stability-oriented weight

It is interesting to look at the frequency weighted gramians Enns's method produces when applying it with the stability-oriented weight of Subsection 2.4.1 :

$$W_s(s) = (I - P_{yu}(s)C(s))^{-1}P_{yu}(s) = \left[\begin{array}{cc|c} A + B_uD_cC_y & B_uC_c & B_u \\ B_cC_y & A_c & 0 \\ \hline C_y & 0 & 0 \end{array}\right].$$

Recall that $W_s(s)$ may be used both as an input weight and an output weight; when used as an output weight, one constructs the realization for the weighted controller :

$$W_s(s)C(s) = \left[\begin{array}{ccc|c} A + B_uD_cC_y & B_uC_c & B_uC_c & B_uD_c \\ B_cC_y & A_c & 0 & 0 \\ 0 & 0 & A_c & B_c \\ \hline C_y & 0 & 0 & 0 \end{array}\right].$$

Since this is a one-sided weighting, one of the two gramians to be balanced is P_c, where $A_cP_c + P_cA_c^T + B_cB_c^T = 0$. We are interested in the observability gramian $\tilde{Q}$ of this realization. Replacing the state vector x of this realization by $T^{-1}\bar{x}$, where $T \triangleq \begin{bmatrix} I & 0 & 0 \\ 0 & I & I \\ 0 & 0 & I \end{bmatrix}$, yields a new realization :

$$W_s(s)C(s) = \left[\begin{array}{ccc|c} A + B_uD_cC_y & B_uC_c & 0 & B_uD_c \\ B_cC_y & A_c & 0 & B_c \\ 0 & 0 & A_c & B_c \\ \hline C_y & 0 & 0 & 0 \end{array}\right].$$

It is clear this realization (and by the same token also the first one) is non-minimal, and its observability gramian $T^{-T}\tilde{Q}T^{-1}$ has the following form :

$$T^{-T}\tilde{Q}T^{-1} = \begin{bmatrix} Q_p & Q_{pc} & 0 \\ Q_{pc}^T & Q_c & 0 \\ 0 & 0 & 0 \end{bmatrix}, \text{ hence } \tilde{Q} = \begin{bmatrix} Q_p & Q_{pc} & Q_{pc} \\ Q_{pc}^T & Q_c & Q_c \\ Q_{pc}^T & Q_c & Q_c \end{bmatrix}.$$

This shows that it is not necessary to follow Enns's procedure completely, whereby one would have to compute an $(n + 2n_c) \times (n + 2n_c)$ gramian. It is sufficient to compute the $(n + n_c) \times (n + n_c)$ observability gramian $\begin{bmatrix} Q_p & Q_{pc} \\ Q_{pc}^T & Q_c \end{bmatrix}$ of the minimal realization

$$W_s(s)C(s) = \left[\begin{array}{cc|c} A + B_uD_cC_y & B_uC_c & B_uD_c \\ B_cC_y & A_c & B_c \\ \hline C_y & 0 & 0 \end{array}\right] \tag{3.12}$$

and take its lower right $n_c \times n_c$ block Q_c to be balanced with P_c.

This represents considerable savings in a situation where $n_c \geq n$, which frequently occurs. If e.g. $n_c = n$ as in LQG or H_∞ design, 7the cost of computing Q_c is lowered from about $700n^3$ operations to about $200n^3$. Computing a gramian of a smaller size is usually also numerically more reliable.

Note that using the same trick, a realization for $W_s(s)(C(s) - C_r(s))$ is found :

$$W_s(s)(C(s) - C_r(s)) = \left[\begin{array}{ccc|c} A + B_uD_cC_y & B_uC_c & -B_uC_r & B_u(D_c - D_r) \\ B_cC_y & A_c & 0 & B_c \\ 0 & 0 & A_r & B_r \\ \hline C_y & 0 & 0 & 0 \end{array}\right],$$

where $C_r(s) = [A_r, B_r, C_r, D_r]$. This realization has $n + n_c + r$ states, which is n_c states less than a 'conventional' realization of $W_s(s)(C(s) - C_r(s))$; this could be useful e.g. in Alg. 2.1.

A dual version of this reduction, where the weight $W_s(s)$ is applied on the right-hand side, is of course also possible; in this case Q_c is found from $A_c^TQ_c + Q_cA_c + C_c^TC_c = 0$, and a realization for $C(s)W_s(s)$ is found, i.e.

$$C(s)W_s(s) = \left[\begin{array}{cc|c} A + B_uD_cC_y & B_uC_c & B_u \\ B_cC_y & A_c & 0 \\ \hline D_cC_y & C_c & 0 \end{array}\right] \tag{3.13}$$

with controllability gramian $\begin{bmatrix} P_p & P_{pc} \\ P_{pc}^T & P_c \end{bmatrix}$.

Remark furthermore that this section applies to discrete-time systems as well, if discrete-time Lyapunov equations are used to compute the gramians.

3.7 Stability-oriented FWBT for observer-based controllers

Just as Section 3.4 presents a refinement of the performance-oriented FWBT of Section 3.3 (i.e. of CLBT) for controllers with observer-feedback structure, stability-oriented FWBT as in Section 3.6 may be adapted for observer-based controller as well.

3.7.1 Continuous-time systems

In this case, with $C(s) = \left[\begin{array}{c|c} A + B_u K_r + K_e C_y & -K_e \\ \hline K_r & 0 \end{array}\right]$, and once more assuming $D_{yu} = 0$, realization (3.12) of $W_s(s)C(s)$ becomes :

$$W_s(s)C(s) = \left[\begin{array}{c|c} \tilde{A} & \tilde{B}_s \\ \hline \tilde{C}_s & 0 \end{array}\right] \triangleq \left[\begin{array}{cc|c} A & B_u K_r & 0 \\ -K_e C_y & A + B_u K_r + K_e C_y & -K_e \\ \hline C_y & 0 & 0 \end{array}\right] \;;$$

(the lower right block of) the observability gramian $\tilde{Q}$ of this realization is needed.

With $T = \begin{bmatrix} I & I \\ 0 & I \end{bmatrix}$ and $\tilde{A}_2 = T^{-T} \tilde{A} T^T$ as in Section 3.4, one finds an equivalent realization :

$$W_s(s)C(s) = \left[\begin{array}{c|c} \tilde{A}_2 & T^{-T}\bar{B} \\ \hline C\bar{T}^T & 0 \end{array}\right] \triangleq \left[\begin{array}{cc|c} A + B_u K_r & B_u K_r & 0 \\ 0 & A + K_e C_y & -K_e \\ \hline C_y & 0 & 0 \end{array}\right]$$

with observability gramian $T\tilde{Q}T^T = \begin{bmatrix} Q_{p2} & Q_{pc2} \\ Q_{pc2} & Q_{c2} \end{bmatrix}$, i.e. with the same lower right block, or $Q_{c2} = Q_c$.

Hence stability-oriented FWBT may be performed by first solving for P_c

$$(A + B_u K_r + K_e C_y) P_c + P_c (A + B_u K_r + K_e C_y)^T + K_e K_e^T = 0 \ ,$$

and consecutively solving for Q_{p2}, Q_{pc2}, and Q_{c2}

$$\begin{aligned}
(A + B_u K_r)^T Q_{p2} + Q_{p2} (A + B_u K_r) + C_y^T C_y &= 0 \ , \\
(A + B_u K_r)^T Q_{pc2} + Q_{pc2} (A + K_e C_y) + Q_{p2} B_u K_r &= 0 \ , \\
(A + K_e C_y)^T Q_{c2} + Q_{c2} (A + K_e C_y) + K_r^T B_u^T Q_{pc2} + Q_{pc2}^T B_u K_r &= 0 \ .
\end{aligned}$$

The same remarks concerning numerical reliability apply, and similar savings in the flop count result, although the advantage is of course somewhat less conspicuous, since structure is only exploited for one of the gramians.

A dual version is found in an analogous way, yielding the equations

$$\begin{aligned}
(A + B_u K_r + K_e C_y)^T Q_c + Q_c (A + B_u K_r + K_e C_y) + K_r K_r^T &= 0 \ , \\
(A + K_e C_y) P_{p2} + P_{p2} (A + K_e C_y)^T + B_u B_u^T &= 0 \ , \\
(A + K_e C_y) P_{pc2} + P_{pc2} (A + B_u K_r)^T - P_{p2} C_y^T K_e^T &= 0 \ , \\
(A + B_u K_r) P_{c2} + P_{c2} (A + B_u K_r)^T - K_e C_y P_{pc2} - P_{pc2}^T C_y^T K_e^T &= 0 \ .
\end{aligned}$$

3.7.2 Discrete-time systems

For completeness, we include also the discrete-time equations :

$$\begin{aligned}
P_c &= (A + B_u K_r + K_e C_y) P_c (A + B_u K_r + K_e C_y)^T + K_e K_e^T \\
Q_{p2} &= (A + B_u K_r)^T Q_{p2} (A + B_u K_r) + C_y^T C_y \\
Q_{pc2} &= (A + B_u K_r)^T Q_{pc2} (A + K_e C_y) + (A + B_u K_r)^T Q_{p2} B_u K_r \\
Q_{c2} &= (A + K_e C_y)^T Q_{c2} (A + K_e C_y) + K_r^T B_u^T Q_{pc2} (A + K_e C_y) \\
&\quad + (A + K_e C_y)^T Q_{pc2}^T B_u K_r + K_r^T B_u^T Q_{p2} B_u K_r
\end{aligned}$$

and the dual :

$$\begin{aligned}
Q_c &= (A + B_u K_r + K_e C_y)^T Q_c (A + B_u K_r + K_e C_y) + K_r K_r^T \\
P_{p2} &= (A + K_e C_y) P_{p2} (A + K_e C_y)^T + B_u B_u^T \\
P_{pc2} &= (A + K_e C_y) P_{pc2} (A + B_u K_r)^T - (A + K_e C_y) P_{p2} C_y^T K_e^T \\
P_{c2} &= (A + B_u K_r) P_{c2} (A + B_u K_r)^T - K_e C_y P_{pc2} (A + B_u K_r)^T \\
&\quad - (A + B_u K_r) P_{pc2}^T C_y^T K_e^T + K_e C_y P_{p2} C_y^T K_e^T \ .
\end{aligned}$$

3.8 Decentralized controller reduction

In some applications it is required that a controller is *decentralized*, i.e. that the controller $C(s)$ has a block diagonal structure or decoupled structure $C(s) = \text{diag}(C_1(s), \ldots, C_m(s))$. The inputs u and the measurement outputs y are subdivided into groups u_i and y_i, $i = 1, \ldots, m$, with the restriction that controller $C_i(s)$ uses only measurements y_i and acts only on control inputs u_i. Such control is in general suboptimal, since one restricts the set of admissible controllers.

Decentralized control design may be necessary e.g. if the physical locations of sensor(s) i and actuator(s) j are too far apart, making one central controller too expensive with respect to to communication, e.g. in a large plant, a power distribution network, etc. It may also be preferred to keep some modularity in a control system; in fact in this light decentralized control systems are more common than 'centralized' ones.

Though in a certain sense the decentralized control makes the control system simpler, it obviously does not reduce the order of the overall controller; in general the order n of a decentralized controller $C(s)$ is the sum of the orders n_i of the 'sub-controllers' $C_i(s)$. Therefore controller reduction is quite likely to be appropriate here as well.

In the sequel we take $m = 2$ for convenience, i.e. $C(s) = \text{diag}(C_1(s), C_2(s))$, but the our treatment can easily be generalized to $m > 2$. The controller therefore has the following structure :

$$C(s) = \left[\begin{array}{c|c} A_c & B_c \\ \hline C_c & D_c \end{array}\right] = \left[\begin{array}{cc|cc} A_{c1} & 0 & B_{c1} & 0 \\ 0 & A_{c2} & 0 & B_{c2} \\ \hline C_{c1} & 0 & D_{c1} & 0 \\ 0 & C_{c2} & 0 & D_{c2} \end{array}\right].$$

The plant can be partitioned accordingly :

$$P(s) = \begin{bmatrix} P_{zw} & P_{zu_1} & P_{zu_2} \\ P_{y_1w} & P_{y_1u_1} & P_{y_1u_2} \\ P_{y_2w} & P_{y_2u_1} & P_{y_2u_2} \end{bmatrix}.$$

Before proceeding, we merely note that translation to the discrete-time case is trivial.

If one applies CLBT to controller $C(s)$, the reduced order controller $C_r(s)$ is no longer guaranteed to have the same block diagonal structure, and will in general be only approximately decentralized. To ensure that $C_r(s)$ is decentralized, a similar reduction procedure may be applied, where the similarity transformation T_c is restricted to have the same block diagonal structure as A_c, i.e. $T_c = \begin{bmatrix} T_{c1} & 0 \\ 0 & T_{c2} \end{bmatrix}$. With such no longer possible to balance P_c and Q_c. but partitioning P_c and Q_c according to the state dimensions of $C_1(s)$ and $C_2(s)$, i.e.

$$P_c = \begin{bmatrix} P_{c1} & P_{c12} \\ P_{c12}^T & P_{c2} \end{bmatrix} \text{ and } Q_c = \begin{bmatrix} Q_{c1} & Q_{c12} \\ Q_{c12}^T & Q_{c2} \end{bmatrix},$$

one may still balance P_{c1} and Q_{c1}, and likewise P_{c2} and Q_{c2}. It is easily seen that this amounts to closed-loop balancing $C_1(s)$ in the presence of $C_2(s)$, and vice versa. Therefore one can again construct weighting functions for both controllers so that Enns's FWBT with these weightings on $C_1(s)$ and $C_2(s)$ yields the same reduced order controllers : e.g., defining

$$\begin{aligned} G_1(s) &= \begin{bmatrix} G_{zw} & G_{zu_1} \\ G_{y_1 w} & G_{y_1 u_1} \end{bmatrix} \\ &\triangleq \begin{bmatrix} P_{zw} & P_{zu_1} \\ P_{y_1 w} & P_{y_1 u_1} \end{bmatrix} + \begin{bmatrix} P_{zu_2} \\ P_{y_1 u_2} \end{bmatrix} (I - C_2 P_{y_2 u_2})^{-1} C_2 \begin{bmatrix} P_{y_2 w} & P_{y_2 u_1} \end{bmatrix}, \end{aligned}$$

$W_{o1} \triangleq G_{zu_1}(I - C_1 G_{y_1 u_1})^{-1}$ and $W_{i1} \triangleq (I - G_{y_1 u_1} C_1)^{-1} G_{y_1 w}$ are equivalent output and input FWBT weights for reduction of $C_1(s)$; a formula for W_{o2} and W_{i2} is obtained by interchanging the indices 1 and 2 in the definition above. In this sense nothing really new has been explained here; but of course even more savings are made in the operation count here, since we need to compute only two $(n + n_{c1} + n_{c2}) \times (n + n_{c1} + n_{c2})$ gramians instead of two $(n + 2n_{c1} + n_{c2}) \times (n + 2n_{c1} + n_{c2})$ and two $(n + n_{c1} + 2n_{c2}) \times (n + n_{c1} + 2n_{c2})$ gramians. Moreover, some information regarding the relative importance of the states in these balanced realizations of $C_1(s)$ and $C_2(s)$ for the closed-loop behavior can be obtained by comparing the 'decentralized closed-loop singular values' $\sigma_k(C_i(s))$, $i = 1, 2$, i.e. the square roots of the eigenvalues of $P_{ci} Q_{ci}$.

By way of example, we apply this technique to the Boeing B-767 airplane model from the IFAC 1990 Control System Design benchmark problems [24, 100, 180, 181]. This is an unstable 55th order model with two control inputs u_e and u_a, and two measured outputs y_e and y_a. The elevator control u_e steers the pitch of the airplane, and the aileron control u_a its roll; y_e is the output of the pitch rate gyro and y_a is the wing-tip acceleration.

The knowledge of the noise characteristics and the performance specifications may be incorporated in a standard plant

$$P(s) = \begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix} = \begin{bmatrix} P_{zw} & P_{ze} & P_{za} \\ P_{ew} & P_{ee} & P_{ea} \\ P_{aw} & P_{ae} & P_{aa} \end{bmatrix}$$

(for the sake of simpler notation we adopt the convention $aa = y_a u_a$, etc.). The model is truly multivariable, i.e. P_{yu} is not decoupled or P_{ea} and P_{ae} are nonzero. An H_2-optimal design for this plant was investigated in [180, 181]. Controller reduction of this design has been considered in [177]. It is a natural question how much performance as measured by the quadratic criterion is lost if one is restricted to decentralized controllers consisting of a controller $C_e(s)$ with input y_e and output u_e, and a controller $C_a(s)$ with input y_a and output u_a.

The method employed here to design a decentralized controller is an iterative one : starting with any controller $C_e(s)$ that makes the aileron loop stabilizable (possibly $C_e(s) = 0$ if the aileron loop is stabilizable as is), first design an H_2-optimal aileron controller $C_a(s)$ for the plant with $C_e(s)$ in the elevator loop; then design a new $C_e(s)$ by optimal control of the plant with $C_a(s)$ controlling the aileron loop; with the new $C_e(s)$ controlling the elevator loop, design a new $C_a(s)$ etc. This iteration is continued until it converges. A less verbal version of this algorithm is given in Alg. 3.1, where we introduce the following notation :

$$H(P, C_e, C_a) \triangleq P_{zw} + \begin{bmatrix} P_{ze} & P_{za} \end{bmatrix} \left(I - \begin{bmatrix} C_e & 0 \\ 0 & C_a \end{bmatrix} \begin{bmatrix} P_{ee} & P_{ea} \\ P_{ae} & P_{aa} \end{bmatrix} \right)^{-1} \begin{bmatrix} C_e & 0 \\ 0 & C_a \end{bmatrix} \begin{bmatrix} P_{ew} \\ P_{aw} \end{bmatrix}.$$

This algorithm, like the D-K iteration, is a kind of cyclic coordinate optimization method [39], and thus is guaranteed to converge to a stationary point of the problem

$$\min_{C_e, C_a} \|H(P, C_e, C_a)\|_2 \ ;$$

however, this may be a local minimum and even a saddle point (though the latter case is unlikely to occur).

Remark that the algorithm, although its simplicity is appealing, suffers from a severe drawback if implemented as above : if the order of $P(s)$ is n and the order of $C_e^0(s)$ is n_0, then the order of $C_a^1(s)$ is $n + n_0$, the order of $C_e^1(s)$ is $2n + n_0$, etc. By induction, one easily finds that the order of $C_a^k(s)$ is $(2n-1)k + n_0$ and the order of $C_e^k(s)$ is $2nk + n_0$. Even if the fact that each new iteration is thus considerably slower than the previous one were acceptable, the algorithm

Algorithm 3.1 : H_2-optimal decentralized control design

Choose a tolerance ϵ
$k = 0$
Find $C_e^0(s)$ s.t.
$P_{aa}(s) + P_{ae}(s)(I - C_e^0(s)P_{ee}(s))^{-1}C_e^0(s)P_{ea}(s)$ stabilizable
do {

$$\begin{aligned}
k &\leftarrow k+1 \\
C_a^k(s) &= \arg\min_{C_a(s)} \|H(P(s), C_e^{k-1}(s), C_a(s))\|_2 \\
C_e^k(s) &= \arg\min_{C_e(s)} \|H(P(s), C_e(s), C_a^k(s))\|_2 \\
E_k &= |\|H(P, C_e, C_a^k)\|_2 - \|H(P, C_e, C_a^{k-1})\|_2|
\end{aligned}$$

} **while** ($E_k > \epsilon$)

will soon break down because of numerical problems caused by the high plant orders. Hence it is essential for the practical implementation that an intermediate model reduction be performed each iteration, such that the order of these controllers never exceeds a certain predetermined number n_i. This is a nice illustration of how plant or controller reduction techniques may be useful or even indispensable in the design process, even if controller order is no concern in the implementation phase.

The value of n_i is at the control designer's discretion; it should be chosen low enough to avoid unacceptably low speed and numerical problems related to size, yet high enough that the performance of the approximations differs negligibly from the full order versions; if n_i is chosen too small, this may cause the algorithm to limit cycle, just as the DK-iteration may exhibit limit cycling if the scaling systems are approximated by models of too low complexity. Therefore, it should be kept in mind that the purpose of this reduction is not to obtain a low order controller, suitable for implementation, but rather to keep the orders encountered during the iteration from growing beyond the workable range.
For our example here $n_i = 20$ was chosen, and it met these two constraints admirably.

The designer has many other options as well : he can choose plant or controller reduction, or possibly both; moreover he could pick any model reduction method for this purpose, ranging from simple balanced or modal truncation to direct fixed order controller optimization methods. Obviously an iterative optimization within the iteration loop is likely to be computationally expensive; on the other hand, it should be clear by now that closed-loop performance considerations should be kept in mind, and that there are better choices here than e.g. mere Balanced Truncation.

We draw attention to the fact that BCRAM is a very good candidate here. As we pointed out above, for H_2-optimal designs it is equivalent to closed-loop balanced truncation, which is very suitable for closed-loop performance preservation if the reduced order is not chosen too small, as is certainly the case here. On the other hand, it is not much more expensive with respect to the number of operations than Moore's balanced truncation.
Thus, a more practical algorithm is Alg. 3.2.

Algorithm 3.2 : H_2-optimal decentralized control design with reduction in the loop

Choose a tolerance ϵ
$k \leftarrow 0$
Find $C_e^0(s)$ s.t.
$P_{aa}(s) + P_{ae}(s)(I - C_e^0(s)P_{ee}(s))C_e^0(s)^{-1}P_{ea}(s)$ stabilizable
do {

$$\begin{aligned}
k &\leftarrow k+1 \\
C_a^k(s) &= \arg\min_{C_a(s)} \|H(P(s), C_e^{k-1}(s), C_a(s))\|_2 \\
C_a^k(s) &\leftarrow \text{order } n_i \text{ BCRAM-model to } C_a^k(s) \\
C_e^k(s) &= \arg\min_{C_e(s)} \|H(P(s), C_e(s), C_a^k(s))\|_2 \\
C_e^k(s) &\leftarrow \text{order } n_i \text{ BCRAM-model to } C_e^k(s) \\
E_k &= |\ \|H(P, C_e, C_a^k)\|_2 - \|H(P, C_e, C_a^{k-1})\|_2\ |
\end{aligned}$$

} **while** ($E_k > \epsilon$)

In contrast with the former ideal algorithm, this algorithm is no longer guaranteed to converge to a stationary point but may well 'limit cycle' or 'oscillate' around such a point; the magnitude of these oscillations of course depend on

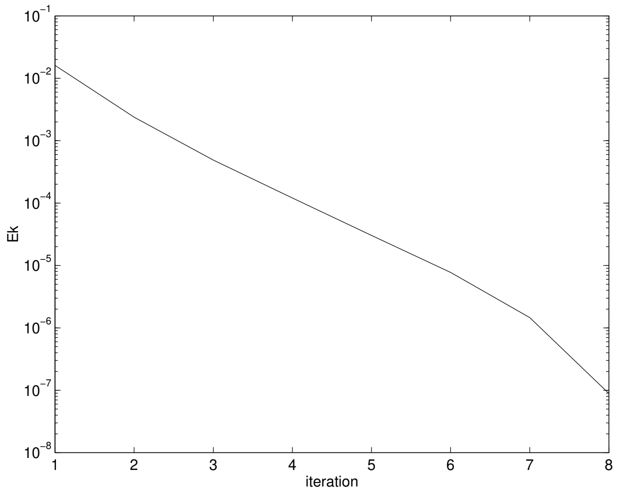


Figure 3.1: Convergence of E_k for Algorithm 3.2.

the value chosen for n_i. This has its parallel in the practical implementation of the DK-iteration for μ synthesis : there too, an approximation of the ideal frequency-dependent scaling matrices $D(s)$ has to be made, the quality of which may determine whether or not such convergence behavior is observed.

In this example, the algorithm converges in 9 iterations when the tolerance is taken $\epsilon = 10^{-6}$; the convergence, as seen from a plot of E_k in Fig. 3.1, appears to be linear here; there is no theoretical indication, however, that this will always be the case. Remark that the decentralized controller yields a closed-loop H_2 norm $\|H_{zw}(s)\|_2^{(dc)} = 0.983$, whereas the unconstrained optimal controller gives $\|H_{zw}(s)\|_2 = 0.967$, which is a gain of only 1.7% with respect to the decentralized one; again, no general conclusions should be drawn from this.

The largest 'decentralized closed-loop singular values' are shown in Table 3.1. Although error bounds in terms of these values cannot be given, they can be useful for determining the orders r_1 and r_2. If one has determined in advance the total controller order $r_1 + r_2$, where r_i is the order of the approximations of $C_i(s)$, it appears to be a good rule of thumb, for a first try at least, to keep the $r_1 + r_2$ largest singular values in both columns; i.e. since the column containing the singular values of $C_2(s)$ has more large values than the column for $C_1(s)$, it is natural to suppose that the former should be approximated with higher order

k	$\sigma_k(C_1(s))$	$\sigma_k(C_2(s))$
1	0.3133	0.2128
2	0.1666	0.1251
3	0.0859	0.1024
4	0.0503	0.0882
5	0.0159	0.0713
6	0.0102	0.0316
7	0.0057	0.0239
8	0.0038	0.0240
9	0.0035	0.0211

Table 3.1: 'Decentralized closed-loop singular values'

than the former, since its states in the new realization 'contribute more' to the closed-loop behavior than those of the latter. This is corroborated by the actual computed errors for some of the reduced order decentralized controllers listed in Table 3.2; the quantity that is actually tabulated there is the deterioration of the H_2 norm relative to the optimal decentralized controller, i.e.

$$\frac{\|H(P(s), C_{r_1}(s), C_{r_2}(s))\|_2}{\|H_{zw}\|_2^{(dc)}} - 1 \; .$$

3.9 Examples

In this section, two further examples are given to show how CLBT may be useful in conjunction with μ-synthesis using the DK-iteration.

3.9.1 Stability-oriented reduction

In a first academic example, Procedure 2.5 is illustrated, with CLBT substituted for the generic approximation scheme denoted there by **appr min**.

r_1 \ r_2	1	2	3	4	5	6	7	8	9
1	58.2	-	-	-	**6.61**	**6.26**	5.95	-	5.85
2	58.3	-	-	-	6.74	6.32	6.07	-	5.90
3	57.9	-	-	-	**5.46**	5.22	4.97	-	4.90
4	64.7	-	-	-	**0.899**	**0.624**	0.797	-	**0.483**
5	65.0	-	-	-	0.830	**0.562**	0.674	-	0.395
6	52.6	-	-	-	0.764	**0.504**	0.656	-	0.363
7	48.9	-	-	-	0.742	0.485	0.619	-	0.338
8	48.9	-	-	-	0.742	0.485	0.619	-	0.338
9	48.7	-	-	-	0.741	0.484	0.618	-	0.338

Table 3.2: Closed-loop H_2 performance deterioration of reduced order decentralized controllers in percents. An unstable closed loop is indicated by - . The best options for fixed $r_1 + r_2$ are printed in boldface.

Consider the standard plant $P(s) \triangleq \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$ with

$$A = \frac{1}{2}\begin{bmatrix} -5 & 0 & 2 & 2 & 2 \\ -2 & -1 & 4 & 2 & 0 \\ 2 & 0 & -1 & 0 & 2 \\ 0 & -2 & 8 & 3 & -2 \\ -2 & 0 & -2 & -2 & -1 \end{bmatrix} \quad B_w = \frac{1}{2}\begin{bmatrix} 6 & 2 \\ -4 & -4 \\ 4 & 0 \\ 9 & 6 \\ -1 & -5 \end{bmatrix} \quad B_u = \frac{1}{2}\begin{bmatrix} -2 & -2 \\ -2 & 8 \\ 2 & 3 \\ 0 & 0 \\ 6 & -3 \end{bmatrix}$$

$$C_z = \tfrac{1}{2}\begin{bmatrix} 1 & 4 & 0 & -6 & -2 \\ 2 & -2 & -4 & -1 & 2 \end{bmatrix} \quad D_{zw} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad D_{zu} = \begin{bmatrix} 0 & -2 \\ -1 & \frac{3}{2} \end{bmatrix}$$

$$C_y = \tfrac{1}{2}\begin{bmatrix} 5 & -2 & 2 & 0 & 4 \\ -4 & -1 & 6 & -4 & -4 \end{bmatrix} \quad D_{yw} = \begin{bmatrix} 0 & -2 \\ 2 & 4 \end{bmatrix} \quad D_{yu} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$C(s)$ is the H_2-optimal controller for $P(s)$; $C(s)$ is stable. A stabilizing controller C_r of order $r = 3$, approximating C, is desired. Stability-oriented FWBT, which aims at making $\|(I - P_{yu}C)^{-1}P_{yu}(C - C_r)\|_\infty$ small, yields a reduced-order controller C_r that is not stabilizing. Alg. 2.5 then proceeds to find a scaling minimizing the μ-norm of this quantity, etc.

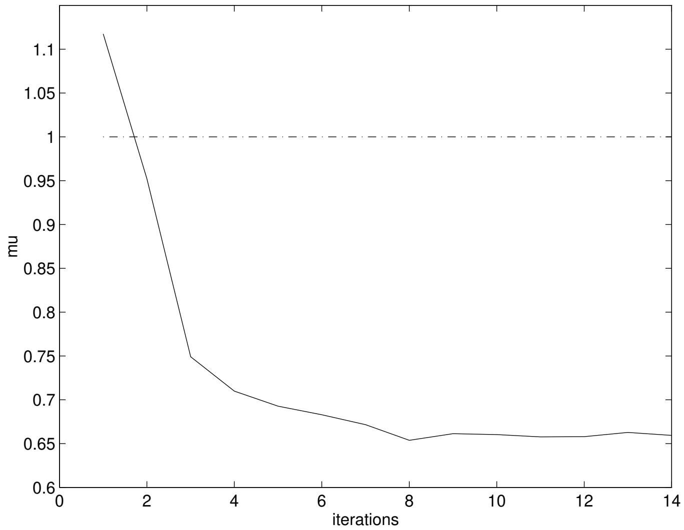


Figure 3.2: Convergence of μ_k for Algorithm 2.5.

Fig. 3.2 plots μ_k versus the iteration count k, where

$$\mu_k \stackrel{\triangle}{=} \|D_k(s)(I - P_{yu}(s)C(s))^{-1}P_{yu}(s)(C(s) - C_r(s))D_k^{-1}(s)\|_\infty \ .$$

After eight iterations or so, 'convergence' has been reached, i.e. μ_k does not decrease anymore, but starts to oscillate a bit; this is due to the fact that the model reduction is not actually minimizing the scaled H_∞ norm. Such oscillations are of course more likely to occur or grow in amplitude as the reduced order r is lower.

3.9.2 CVT with μ control

In [309], μ synthesis is applied to design a controller for a Continuously Varying Transmission (CVT). A standard plant P and corresponding uncertainty block structure Δ is constructed there modeling eight uncertainties of this system, caused by nonlinearities, (temperature-dependent) time variations, and tolerances on components; this is done in such a way that the controller C minimizing the corresponding closed-loop μ-norm of the plant yields the best possible robust closed-loop performance.
Using the D-K iteration, a controller C_{39} of order 39 is found yielding

$$\min_{\omega\in\mathbb{R}} \mu_\Delta(F_l(P(j\omega), C_{39}(j\omega))) \leq 0.966 < 1 \ ,$$

which means that both robust stability and performance can be guaranteed as specified. Using Moore's Balanced Truncation (BT), C_{39} is reduced to a sixth order controller C_{bt}, with

$$\min_{\omega \in \mathbb{R}} \mu_\Delta(F_l(P(j\omega), C_{bt}(j\omega))) \leq 0.979 < 1 \; ;$$

lower order approximations obtained by BT yield a μ-norm larger than one.

Let $D(s)$ be the optimal scaling system, i.e.

$$D(s) = \arg \min_{D, D^{-1} \in S_D} \|D(s) F_l(P(s), C_{39}(s)) D(s)^{-1}\|_\infty \; ,$$

then CLBT with the standard plant $\begin{bmatrix} D & 0 \\ 0 & I \end{bmatrix} P \begin{bmatrix} D^{-1} & 0 \\ 0 & I \end{bmatrix}$ and controller C_{39} gives a fourth-order controller C_{clbt} such that $\|D F_l(P, C_{clbt}) D^{-1}\|_\infty = 1.004$, at first sight failing to guarantee robust stability and performance. Recomputing the scalings however, one finds $\min_{D, D^{-1} \in S_D} \|D F_l(P, C_{clbt}) D^{-1}\|_\infty = 0.9976$, i.e. the control system is robust against all specified perturbations after all.

3.10 Conclusions

Closed-Loop Balanced Truncation [65, 302, 305], is shown to be equivalent to (but more efficient than) Enns's Frequency Weighted Balanced Truncation [93] with certain performance-oriented weightings.

Furthermore, it is indicated how the computational effort of CLBT may be reduced considerably if the method is used to reduce the order of an observer-based controller, as obtained e.g. using pole placement or LQG techniques. In this case, the structure of the system matrices enables one to find the relevant sub-blocks of the closed-loop gramians using about one fourth of the number of operations needed otherwise.

Similar improvements are found for FWBT with a stability-oriented weight; again, for the general case, the size of one of the Lyapunov equations to be solved may be reduced considerably, and even more savings are obtained for observer-based controllers.

The idea of CLBT is also shown to be useful for reduction of decentralized controllers.

Chapter 4

Relative error model reduction

In this chapter, two relative error model reduction methods are treated.

In Chapter 3, we have demonstrated how Enns's Frequency Weighted Balanced Truncation (FWBT) [93, 94, 24] may be simplified considerably for some specific frequency weights. As relative error approximation is but a special case of frequency weighted model reduction, it seems a good idea to investigate what gains may be obtained when using FWBT with as a weighting function the inverse of the outer factor of the system to be reduced. The computation of the observability gramian of the weighted system in Enns's method can be reduced in this case to the computation of the observability gramian of the weighting function only.
This yields a new method, Inverse Outer Factor Weighted Balanced Truncation (IOWBT), very similar to Balanced Stochastic Truncation (BST), in the sense that a spectral factorization of $G(s)G^T(-s)$ is performed, involving the solution of a Riccati equation, and a controllability gramian to be computed. Both methods perform very similarly as well; for minimum-phase systems, they are even equivalent.

Moreover, it is shown that for square causal invertible systems, both IOWBT and BST may be performed by solving a Riccati equation without constant term, which may be solved with less effort. The derivations are done both for the continuous-time and the discrete-time case.

Extensions for non-square systems are given as well; however, some of the advantages of the new schemes are lost in this case.

4.1 Introduction

A type of weighted error model reduction that has attracted a lot of attention and has been treated extensively in the literature is relative error model reduction. When approximating a square causal invertible system G by a reduced-order model G_r, the *relative error* is defined as $G^{-1}(G_r - G)$ and the *multiplicative error* as $G_r^{-1}(G_r - G)$, or dually as $(G_r - G)G^{-1}$ and $(G_r - G)G_r^{-1}$, respectively.

Relative error model reduction methods attempt to make some measure of the relative error small, e.g. $\|G^{-1}(G_r - G)\|_{L_\infty}$ or $\|G^{-1}(G_r - G)\|_{L_2}$.

Note that when G is not causal invertible, but $s^m G$ (or $z^m G$ for DT systems) is for some integer $m > 0$, then relative error may still be defined as $(s^m G)^{-1} s^m (G_r - G)$ (or $(z^m G)^{-1} z^m (G_r - G)$); the pole and zero at $s = 0$ $(z = 0)$ evidently cancel out.

Of course, such a model reduction method may be a sensible approximation technique in many fields besides control design. For instance, when approximating a notch filter, or more generally a filter with stop-bands and pass-bands, one is interested in a good fit on a logarithmically scaled plot rather than a small absolute error; indeed, an approximation obtained by Balanced Truncation would probably fit the passband well but the attenuation in the stopband would probably be much worse. Hence for the approximation of many filters, it is important that not the additive error is minimized, but the relative error.

Two related important relative error approximation methods have arisen in the context of stochastic realization theory, for the approximation of spectra [11] : Balanced Stochastic Truncation (BST) [78, 133, 120, 121, 123, 233, 290, 291] and phase matching [134, 149, 110, 119, 122, 209, 150]. Other contributions were made by Zhou [318] and Glover [109].

As a motivation for relative error model reduction of controllers, Safonov also gave the following not quite rigorous argument [233]. A common control design specification is good tracking performance. Consider a square plant P and a one-degree-of-freedom (1-DOF) controller C (Fig. 4.1) achieving good tracking, i.e. $H = (I + PC)^{-1}PC \approx I$. Suppose C is causal invertible, i.e. C^{-1} does exist. It was mentioned in Subsection 2.4.1 that if the stable controller C is replaced by another one C_r, the closed loop will still be stable if

$$\|W_s(C - C_r)\|_\infty < 1 , \qquad \text{where } W_s \stackrel{\triangle}{=} (I + PC)^{-1}P ;$$

this condition can be rewritten as

$$\|(I + PC)^{-1}PCC^{-1}(C - C_r)\|_\infty < 1 .$$

Now since $H = (I + PC)^{-1}PC \approx I$ for a tracking controller, one may approximate $(I + PC)^{-1}PCC^{-1}(C - C_r)$ by the relative error $C^{-1}(C - C_r)$; hence for preserving closed-loop stability it is good to keep $\|C^{-1}(C - C_r)\|_\infty$ small.

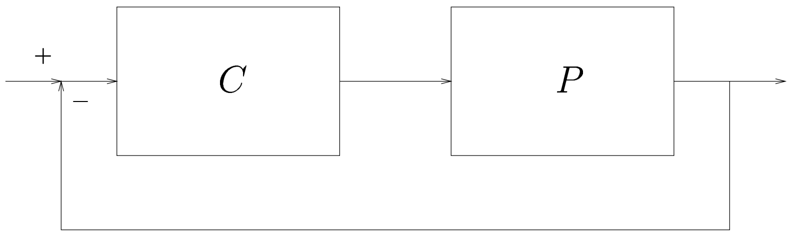


Figure 4.1: 1-DOF configuration

Of course some hand-waving has been going on here. In tracking systems, the relation $(I + P(j\omega)C(j\omega))^{-1}P(j\omega)C(j\omega) \approx I$ can only be true up to a certain bandwidth Ω_B; most often, $\bar{\sigma}\big((I + P(j\omega)C(j\omega))^{-1}P(j\omega)C(j\omega)\big) < 1$ for higher frequencies. Hence minimizing the relative error may be more conservative than minimizing the frequency weighted error $\|W_s(C - C_r)\|_\infty$; specifically, high frequencies will be emphasized too much. However, for many model reduction methods (e.g. those based on projection of dynamics), the modeling error is zero at infinite frequencies, and thus often the relative error $C^{-1}(C_r - C)$ is significant only for frequencies below the closed-loop bandwidth, and one has indeed $(I + PC)^{-1}P(C_r - C) \approx C^{-1}(C_r - C)$.

A more rigorous formulation would be

Theorem 4.1
Let P *be a square system and* C *a stable and causal invertible controller such that the 1-DOF control loop is stable. Let* $\|(I + PC)^{-1}PC\|_\infty = (1 + \epsilon)$.
Then *a stable controller* C_r *stabilizes* P *in the 1-DOF setup if*

$$\|C^{-1}(C_r - C)\|_{L_\infty} < (1 + \epsilon)^{-1} .$$

Proof :

$$\begin{aligned} \|W_s(C - C_r)\|_\infty &= \|(I + PC)^{-1}P(C - C_r)\|_\infty \\ &\leq \|(I + PC)^{-1}PC\|_\infty \|C^{-1}(C - C_r)\|_{L_\infty} < 1 \qquad \square \end{aligned}$$

Of course this theorem is valid for other controllers than those that achieve good tracking as well; but then the weighting will not be approximately equivalent to the stability-oriented weight, and relative error model reduction is more conservative.

Some conservatism is necessarily introduced by reducing the relative error rather than the proper stability-oriented weight W_s; so relative error will need some further justification if it will be applied for this purpose, e.g. that there is a simpler or faster algorithm, or one that guarantees better results.

This is indeed the case : one well-known method, Balanced Stochastic Truncation (BST), has a pre-computable H_∞ error bound on the relative error. For another one, proposed in the following section, the computational burden is smaller than for frequency weighted model reduction with weight W_s.

4.2 Inverse Outer Factor Weighted Balanced Truncation

Let G be a square, stable, causal invertible plant of McMillan degree n.
Let $[A, B, C, D]$ be a minimal realization of G. Recall that G^{-1} has a realization $\left[\begin{array}{c|c} A_i & B_i \\ \hline C_i & D_i \end{array}\right] \triangleq \left[\begin{array}{c|c} A - BD^{-1}C & BD^{-1} \\ \hline -D^{-1}C & D^{-1} \end{array}\right]$, which is also minimal.
The number of non-minimum phase zeroes of G is denoted by p.

One could regard the relative error model reduction problem as a special case of a frequency weighted model reduction problem, where the inverse system is chosen as a weighting system on the input or the output side. Therefore, every frequency weighted model reduction procedure is a candidate to be specialized for relative error approximation.

Frequency Weighted Balanced Truncation (FWBT) [93] is one such procedure. The reader is referred to p. 54 for a review of this method. For minimum-phase systems, i.e. $p = 0$, G^{-1} is stable, and this method can be directly applied to find a reduced order model with small relative error; this was treated by Zhou et al. [320, 321, 324]. Zhou found that in this case, there is structure in the weighted gramians computed in FWBT, such that the parts of these gramians that are to be balanced may be found by solving two $n \times n$ Lyapunov equations. Furthermore, he noted that if G is minimum phase, Enns's method with the inverse system of G as a weighting function is equivalent to Balanced Stochastic Truncation (BST) [78, 120] (in the sense that both methods yield the same reduced-order system); remark that this is a relevant special case, particularly when reducing a stochastic realization, where one is only interested in approximating a spectrum. An error bound was given that is equivalent to the error bound proposed by Green [121] for BST.

For non-minimum phase systems, the inverse is unstable, and as for Enns's scheme, the weighting systems need to be stable, so it seems to be less suited for non-minimum phase systems. Next, we discuss how Enns's scheme can still be useful for these systems, and find an elegant relative error model reduction method from this. In Subsection 4.2.2, we treat the discrete-time case. In Section 4.3, we show that the equations to be solved in Balanced Stochastic Truncation (BST) may be put in very similar form.

4.2.1 Continuous-time systems

As mentioned above, the procedure of Zhou cannot be used directly with non-minimum phase systems, because the weighting system is then no longer stable. We propose to generalize Zhou's approach by weighting with the inverse of the left outer spectral factor $W(s)$ of the system $G(s)$ at hand, i.e. $W(s)$ is stable and minimum phase with $W(s)W^T(-s) = G(s)G^T(-s)$. Note that Enns already advances this fix for unstable weighting systems; the question is, whether the gramians still have structure with this weight, like the one that Zhou found in the case of minimum-phase systems. Remark that in such a case $W^{-1}G$ is all-pass, W^{-1} is stable, and

$$\|G^{-1}(G - G_r)\|_{L_\infty} = \|W^{-1}GG^{-1}((G - G_r)\|_{L_\infty} = \|W^{-1}(G - G_r)\|_\infty \, .$$

The derivation of realizations for $W(s)$ and $W^{-1}(s)$ is quite standard, but as it is short, it is replicated here. Using these in Enns's procedure, it will be shown that P and Q of Enns's scheme can be found by solving two Lyapunov equations (4.1), (4.3) and a Riccati equation without a constant term (4.2). It is then shown how a Riccati equation without a constant term can be solved with a smaller number of operations than a general Riccati equation.

Derivation

$$\text{Let } G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right], \qquad \text{then} \qquad G^T(-s) = \left[\begin{array}{c|c} -A^T & C^T \\ \hline -B^T & D^T \end{array}\right] .$$

In order to find the stable spectral factors of $G(s)G^T(-s)$, we find its stable part, i.e. $Z_G(s)$ such that $Z_G(s) + Z_G^T(-s) = G(s)G^T(-s)$.
Let P be the controllability gramian of $[A, B, C, D]$, i.e.

$$AP + PA^T + BB^T = 0 \; ; \tag{4.1}$$

then $$Z_G(s) + Z_G^T(-s) = G(s)G^T(-s) = \left[\begin{array}{cc|c} A & -BB^T & BD^T \\ 0 & -A^T & C^T \\ \hline C & -DB^T & DD^T \end{array}\right]$$
$$= \left[\begin{array}{cc|c} A & 0 & BD^T + PC^T \\ 0 & -A^T & C^T \\ \hline C & -(DB^T + CP) & DD^T \end{array}\right]$$

hence $Z_G(z) = \left[\begin{array}{c|c} A & BD^T + PC^T \\ \hline C & \frac{1}{2}DD^T \end{array}\right]$.

$W(s)W^T(-s) = G(s)G^T(-s)$ implies $Z_W(s) = Z_G(s)$. To find $W(s)$, we propose a candidate realization $W(s) = \left[A, \bar{B}, C, D\right]$, where the matrix $\bar{B}$ is to be determined such that $\bar{B}D^T + \bar{P}C^T = BD^T + PC^T$ and $A\bar{P} + \bar{P}A^T + \bar{B}\bar{B}^T = 0$; note that $\bar{P}$ is the controllability gramian of $W(s) = \left[A, \bar{B}, C, D\right]$.

Defining $X \triangleq \bar{P} - P$, one finds

$$\begin{aligned} \bar{B} &= B - XC^TD^{-T} \\ \bar{B}\bar{B}^T &= (B - XC^TD^{-T})(B - XC^TD^{-T})^T = BB^T - AX - XA^T \ , \end{aligned}$$

which reduces to
$$A_iX + XA_i^T + XC_i^TC_iX = 0 \ . \tag{4.2}$$

It is also easily derived that $W_o(s) \triangleq (W(s))^{-1} = [A_{wi}, B_{wi}, C_i, D_i]$, where

$$\begin{aligned} A_{wi} &\triangleq A - \bar{B}D^{-1}C = A - BD^{-1}C + XC^TD^{-T}D^{-1}C = A_i + XC_i^TC_i, \\ B_{wi} &\triangleq \bar{B}D^{-1} = BD^{-1} - XC^TD^{-T}D^{-1} = B_i + XC_i^TD_i \ . \end{aligned}$$

If X is the solution to the Riccati equation (4.2) such that $A_{wi} = A_i + XC_i^TC_i$ has all its eigenvalues in the left half plane, then $W(s)$ is minimum phase and its inverse is stable.

With this weighting system, and defining $T \triangleq \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}$, Enns's procedure may be simplified considerably :

$$W^{-1}(s)G(s) = \left[\begin{array}{cc|c} A_i + XC_i^TC_i & B_iC - XC_i^TC_i & B + XC_i^T \\ 0 & A & B \\ \hline C_i & -C_i & I \end{array}\right]$$
$$= \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & I \end{array}\right] = \left[\begin{array}{c|c} T\tilde{A}T^{-1} & T\tilde{B} \\ \hline \tilde{C}T^{-1} & I \end{array}\right]$$

$$= \left[\begin{array}{cc|c} A_i + XC_i^T C_i & 0 & XC_i^T \\ 0 & A & B \\ \hline C_i & 0 & I \end{array}\right]$$

so that one finds

$$\tilde{Q} = T^T \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix} T = \begin{bmatrix} Q & -Q \\ -Q & Q \end{bmatrix}$$

with Q the solution of $\qquad A_{wi}^T Q + QA_{wi} + C_i^T C_i = 0 \; . \qquad (4.3)$

This derivation proves the following

Theorem 4.2
Let $G(s) \triangleq [A, B, C, D]$ *be a square CT stable causal invertible system,*
Let $W(s)$ *be a minimum-phase system such that* $W(s)W^T(-s) = G(s)G^T(-s)$*,*
Let $G_r(s)$ *be the reduced-order system of order* r *obtained by Frequency Weighted Balanced Truncation with* $W^{-1}(s)$ *as an output weight and* I *as an input weight,*
Define $A_i \triangleq A - BD^{-1}C$, $C_i \triangleq -D^{-1}C$, *and* $A_{wi} \triangleq A_i + XC_i^T C_i$*,*
Let X *be the stabilizing solution to* $A_iX + XA_i^T + XC_i^T C_i X = 0$ *,* $\qquad (4.2)$

Then *finding a realization of* $G(s)$ *such that* P *and* Q *are balanced, where* P *and* Q *satisfy*

$$\begin{array}{rcll} AP + PA^T + BB^T & = & 0 & (4.1) \\ A_{wi}^T Q + QA_{wi} + C_i^T C_i & = & 0 & (4.3) \end{array}$$

and truncating this realization accordingly to order r *results in the same reduced-order system* $G_r(s)$*.*

Trivial solutions

The Riccati equation (4.2) has two special solutions, which can be computed at little or no cost. Whether these solutions are stabilizing, depends on the number of non-minimum phase zeroes.

One is the trivial solution $X = 0$. This solution is the stabilizing one if $G(s)$ is minimum phase; indeed in this case $A_{wi} = A_i + XC_i^T C_i = A_i$ is stable.
One then finds $Q = Q_w = Q_i$ (see [320, 321, 324]), where Q_i is the solution of $A_i^T Q_i + Q_i A_i + C_i^T C_i = 0$.

Another special solution to the Riccati equation (4.2) is Q_i^{-1} (note that Q_i is invertible for a minimal system $G(s)$).

In this case, $A_{wi} = A_i + Q_i^{-1}C_i^TC_i = -Q_i^{-1}A_i^TQ_i$, which is a stable matrix if $G^{-1}(s)$ is anti-stable. Moreover, we have that $Q = Q_w = -Q_i$, since $(-Q_iA_iQ_i^{-1})(-Q_i) + (-Q_i)(-Q_i^{-1}A_i^TQ_i) + C_i^TC_i = Q_iA_i + A_i^TQ_i + C_i^TC_i = 0$.

Solving the Riccati equation without constant term

Even in the general case, when $G^{-1}(s)$ is neither stable nor anti-stable, the Riccati equation $A_iX + XA_i^T + XC_i^TC_iX = 0$ (4.2) can be solved faster than a general algebraic Riccati equation.

One should find a basis $\begin{bmatrix} X_1^T & X_2^T \end{bmatrix}^T$ for the anti-stable eigenspace of the Hamiltonian corresponding to (4.2), i.e.

$$\begin{bmatrix} A_i & 0 \\ -C_i^TC_i & -A_i^T \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \Lambda \, ;$$

for any eigenspace basis chosen such that X_2 is invertible, $X_1X_2^{-1}$ yields a solution to (4.2), and it is easy to verify that if Λ is anti-stable, $A_i + XC_i^TC_i$ is stable. The means to find such an eigenspace in a numerically reliable fashion is provided by the QR algorithm for Schur decomposition. Once a Schur decomposition is found, the Schur matrix can be reordered so that the anti-stable eigenvalues are all in the upper left block, and a basis for the anti-stable eigenspace may directly be read off. One may take advantage of the fact that one has the trivial solution $X = 0$ to write down a Schur decomposition of the Hamiltonian; indeed, $\begin{bmatrix} 0 & X_2^T \end{bmatrix}^T$ is then a basis for one eigenspace. Let $A_i = USU^T$ be a real Schur decomposition of A_i, with all right half plane eigenvalues in the upper left block, an let J be the permutation matrix of the same size as A_i with ones on the anti-diagonal and zeroes elsewhere; then the knowledge of the trivial solution $X = 0$ enables us to write

$$\begin{bmatrix} A_i & 0 \\ -C_i^TC_i & -A_i^T \end{bmatrix} \begin{bmatrix} 0 & U \\ UJ & 0 \end{bmatrix} = \begin{bmatrix} 0 & U \\ UJ & 0 \end{bmatrix} \begin{bmatrix} -JS^TJ & -JU^TC_i^TC_iU \\ 0 & S \end{bmatrix} \tag{4.4}$$

Using an orthogonal transformation $V \triangleq \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}$, one can reorder the $2n \times 2n$ Schur matrix such that all right half plane eigenvalues are in the upper left block. Therefore

$$\begin{bmatrix} A_i & 0 \\ -C_i^TC_i & -A_i^T \end{bmatrix} \begin{bmatrix} 0 & U \\ UJ & 0 \end{bmatrix} \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & U \\ UJ & 0 \end{bmatrix} \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} S_+ & S_{12} \\ 0 & S_- \end{bmatrix} \tag{4.5}$$

where S_+ (S_-) is upper triangular with only RHP (resp. LHP) eigenvalues. The stabilizing solution to the Riccati equation (4.2) is then $X = UV_{21}V_{11}^{-1}JU^T$.

Note that the preliminary reordering of the Schur matrix S, such that the anti-stable eigenvalues are in the upper left block, limits the number of subsequent reordering swaps : one only needs to swap the lower right $p \times p$ block of $-JS^TJ$, containing its stable eigenvalues, with the upper left block of S; this requires p^2 swaps.

Non-invertible systems

So far, it has been required that the system $G(s)$ be causal invertible, i.e. that its D-matrix be invertible. Apparently, this excludes application of this method to a class of very commonly occurring controllers or other systems, to wit those that are strictly proper. Various schemes can be devised to get around this, but one of the most simple-minded is very adequate : just perturb the D-matrix with a small invertible matrix, e.g. ϵI, where ϵ could be equal to the magnitude $|G(j\omega) - D|$ at some frequency ω well above the critical frequency of $G(s)$; it is rare that this number ϵ needs to be larger than $10^{-4}A$ (where A is the passband magnitude), and numerical problems due to a too small ϵ are to be expected only when $\frac{\epsilon}{A}$ approaches the machine precision.
This small perturbation ϵI is dropped when truncating, so that the reduced-order system has the same D-matrix as $G(s)$.

This is of course cheating, as one is changing the weighting function, but this particular change has very little impact. Indeed, the only effect this has is that the infinitely large weight on the high-frequency error is replaced by a very large but bounded weight; but the error is zero at infinite frequencies anyway, as the approximant has the same D-matrix as the original system.

4.2.2 Discrete-time systems

The purpose is here to achieve the discrete-time equivalent of the previous Subsection 4.2.1 : for a discrete-time system $G(z)$, to find a reduced-order approximant $G_r(z)$ such that the relative error $\|G^{-1}(z)(G(z) - G_r(z))\|_{L_\infty}$ is made small.

The derivation for the discrete-time case proceeds analogously to the continuous-time case : first a realization of the inverse of the outer factor $W(z)$ is found and used in Enns's procedure. The desired blocks P and Q of the gramians are then found by solving similar Riccati and Lyapunov equations :

Theorem 4.3
Let $G(z) \triangleq [A, B, C, D]$ *be a square DT stable causal invertible system,*
Let $W(z)$ *be a minimum-phase system such that* $W(z)W^T(z^{-1}) = G(z)G^T(z^{-1})$,
Let $G_r(z)$ *be the reduced-order system of order* r *obtained by Frequency Weighted Balanced Truncation with* $W^{-1}(z)$ *as an output weight and* I *as an input weight,*
Defining $A_i \triangleq A - BD^{-1}C$, $C_i \triangleq -D^{-1}C$, *let* X *be the stabilizing solution to*

$$A_i X C_i^T (I + C_i X C_i^T)^{-1} C_i X A_i^T - A_i X A_i^T + X = 0 \tag{4.6}$$

and let $\bar{D}$ *be a square matrix such that* $\bar{D}\bar{D}^T = DD^T + CXC^T$,
$\bar{A}_i \triangleq A_i(I + XC_i^T C_i)^{-1}$ *and* $\bar{C}_i \triangleq -\bar{D}_i^{-1}C$,
Let P *and* Q *satisfy*

$$\begin{aligned} APA^T - P + BB^T &= 0 \qquad (4.7) \\ \bar{A}_i^T Q \bar{A}_i - Q + \bar{C}_i^T \bar{C}_i &= 0\,, \qquad (4.8) \end{aligned}$$

Then *finding a realization of* $G(z)$ *such that the corresponding* P *and* Q *are balanced, and truncating this realization to order* r *results in the same reduced-order system* $G_r(s)$.

Proof :

We have
$$G^T(z^{-1}) = \left[\begin{array}{c|c} A^{-T} & A^{-T}C^T \\ \hline -B^T A^{-T} & D^T - B^T A^{-T} C^T \end{array}\right].$$

In order to find the stable spectral factors of $G(z)G^T(z^{-1})$, we find a stable $Z_G(z)$ such that

$$\begin{aligned} Z_G(z) + Z_G^T(z^{-1}) &= G(z)G^T(z^{-1}) \\ &= \left[\begin{array}{cc|c} A & -BB^T A^{-T} & BD^T - BB^T A^{-T} C^T \\ 0 & A^{-T} & A^{-T}C^T \\ \hline C & -DB^T A^{-T} & DD^T - DB^T A^{-T} C^T \end{array}\right] \\ &= \left[\begin{array}{cc|c} A & 0 & BD^T + APC^T \\ 0 & A^{-T} & A^{-T}C^T \\ \hline C & -(DB^T + CPA^T)A^{-T} & DD^T - DB^T A^{-T} C^T \end{array}\right] \end{aligned}$$

where $APA^T - P + BB^T = 0$; (4.7)

therefore $Z_G(z) = \left[\begin{array}{c|c} A & BD^T + APC^T \\ \hline C & D_z \end{array}\right]$.
D_z should satisfy $DD^T - DB^T A^{-T} C^T = D_z + D_z^T - DB^T A^{-T} C^T - CPC^T$,

hence we may choose $D_z = \frac{1}{2}(DD^T + CPC^T)$.

As $W(z)W^T(z^{-1}) = G(z)G^T(z^{-1})$, $Z_W(z)$ should equal $Z_G(z)$; hence we propose $W(z) = \left[A, \bar{B}, C, \bar{D}\right]$, and try to find $\bar{B}$ and $\bar{D}$ such that

$$\begin{aligned}
\bar{B}\bar{D}^T + A\bar{P}C^T &= BD^T + APC^T \ ,\\
\bar{D}\bar{D}^T + C\bar{P}C^T &= DD^T + CPC^T \ ,\\
\bar{B}\bar{B}^T + A\bar{P}A^T &= \bar{P} \ .
\end{aligned}$$

Defining $X = P - \bar{P}$, and $A_i \triangleq A - BD^{-1}C$, $B_i \triangleq BD^{-1}$, $C_i \triangleq -D^{-1}C$, $D_i \triangleq D^{-1}$ as before, one has :

$$\begin{aligned}
\bar{B}\bar{D}^T &= BD^T + AXC^T\\
\bar{B}\bar{B}^T &= BB^T + AXA^T - X\\
\bar{D}\bar{D}^T &= DD^T + CXC^T \ .
\end{aligned}$$

Now since $\bar{B}\bar{B}^T = \bar{B}\bar{D}^T(\bar{D}\bar{D}^T)^{-1}\bar{D}\bar{B}^T$, the following equalities hold :

$$\begin{aligned}
&BB^T + AXA^T - X\\
&\quad = \ (BD^T + AXC^T)(DD^T + CXC^T)^{-1}(BD^T + AXC^T)^T,\\
&BB^T + (A_i - BC_i)X(A_i - BC_i)^T - X\\
&\quad = \ (B - (A_i - BC_i)XC_i^T)(I + C_iXC_i^T)^{-1}(B - (A_i - BC_i)XC_i^T)^T,\\
&B(I + C_iXC_i^T)B^T - A_iXC_i^TB^T - BC_iXA_i^T + A_iXA_i^T - X\\
&\quad = \ (B(I + C_iXC_i^T) - A_iXC_i^T)(I + C_iXC_i^T)^{-1}(B(I + C_iXC_i^T) - A_iXC_i^T)^T,
\end{aligned}$$

which results in :

$$A_iXC_i^T(I + C_iXC_i^T)^{-1}C_iXA_i^T - A_iXA_i^T + X = 0 \ . \qquad (4.6)$$

One easily finds $W^{-1}(z) = \left[\bar{A}_i, \bar{B}_i, \bar{C}_i, \bar{D}_i\right]$, with

$$\begin{aligned}
\bar{A}_i &\triangleq A - \bar{B}\bar{D}^{-1}C = A + (B - AXC_i^T)(I + C_iXC_i^T)^{-1}C_i\\
&= A(I + XC_i^TC_i)^{-1} + BC_i(I + XC_i^TC_i)^{-1} = A_i(I + XC_i^TC_i)^{-1} \ ,\\
\bar{B}_i &\triangleq \bar{B}\bar{D}^{-1} = B_i - A_iXC_i^T(I + C_iXC_i^T)^{-1}D_i \ ,\\
\bar{C}_i &\triangleq -\bar{D}^{-1}C \ , \quad \text{and} \quad \bar{D}_i \triangleq \bar{D}^{-1} \ .
\end{aligned}$$

Again choosing the solution X of algebraic Riccati equation (4.6) such that $\bar{A}_i$ has all its eigenvalues inside the unit circle, Enns's procedure simplifies as in the continuous-time case :

$$\text{if} \qquad W^{-1}(z)G(z) = \left[\begin{array}{cc|c} \bar{A}_i & \bar{B}_iC & \bar{B}_iD \\ 0 & A & B \\ \hline \bar{C}_i & 0 & \bar{D}_iD \end{array}\right] = \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right]$$

$$\text{and } T \triangleq \left[\begin{array}{cc} I & -I \\ 0 & I \end{array}\right], \text{ then } \left[\begin{array}{c|c} T\tilde{A}T^{-1} & T\tilde{B} \\ \hline \tilde{C}T^{-1} & \tilde{D} \end{array}\right] = \left[\begin{array}{cc|c} \bar{A}_i & 0 & -\bar{A}_i X C_i^T \\ 0 & A & B \\ \hline \bar{C}_i & 0 & \bar{D}_i D \end{array}\right],$$

so that one finds

$$\tilde{Q} = T^T \left[\begin{array}{cc} Q & 0 \\ 0 & 0 \end{array}\right] T = \left[\begin{array}{cc} Q & -Q \\ -Q & Q \end{array}\right]$$

$$\text{where} \qquad Q = \bar{A}_i^T Q \bar{A}_i + \bar{C}_i^T \bar{C}_i \ . \qquad (4.8) \qquad \square$$

Trivial solutions

Exactly as in the continuous-time case, the Riccati equation (4.6) has two special solutions.

Evidently $X = 0$ is again a solution, corresponding to $\bar{A}_i = A_i(I + XC_i^T C_i)^{-1} = A_i$, so this is again the stabilizing solution if $G(s)$ is minimum phase; also $Q = Q_w = Q_i$ (see [320, 321, 324]), where Q_i is the solution of $Q_i = A_i^T Q_i A_i + C_i^T C_i$.

$X = -Q_i^{-1}$ is easily verified to be another special solution to the Riccati equation (4.6). In this case, $\bar{A}_i = A_i(I - Q_i^{-1} C_i^T C_i)^{-1} = Q_i^{-1} A_i^{-T} Q_i$, which is a stable matrix if $G^{-1}(s)$ is anti-stable.

Solving the Riccati equation

It's readily verified that $X = X_2 X_1^{-1}$ is a solution to (4.6), where X_1, X_2 and Λ are $n \times n$ matrices such that

$$\left[\begin{array}{cc} I & C_i^T C_i \\ 0 & A_i \end{array}\right] \left[\begin{array}{c} X_1 \\ X_2 \end{array}\right] = \left[\begin{array}{cc} A_i^T & 0 \\ 0 & I \end{array}\right] \left[\begin{array}{c} X_1 \\ X_2 \end{array}\right] \Lambda \ .$$

Moreover, $A_i(I + XC_i^T C_i)^{-1} = (X_1 \Lambda^{-1} X_1^{-1})^T$, i.e. $\bar{A}_i$ is stable iff Λ has all its eigenvalues outside the unit disk. Therefore if $A_i = USU^T$ is a Schur decomposition of A_i, then

$$\begin{aligned} &\left[\begin{array}{cc} I & C_i^T C_i \\ 0 & A_i \end{array}\right] \left[\begin{array}{cc} UJ & 0 \\ 0 & U \end{array}\right] \\ &\qquad = \left[\begin{array}{cc} A_i^T & 0 \\ 0 & I \end{array}\right] \left[\begin{array}{cc} UJ & 0 \\ 0 & U \end{array}\right] \left[\begin{array}{cc} JS^{-T}J & JS^{-T}U^T C_i^T C_i U \\ 0 & S \end{array}\right] . \end{aligned}$$

Reordering the diagonal elements of the rightmost matrix above, all elements outside the unit disk can be put in the leftmost upper block, using an orthogonal transformation V :

$$\begin{bmatrix} JS^{-T}J & JS^{-T}U^TC_i^TC_iU \\ 0 & S \end{bmatrix} \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} S_+ & S_{12} \\ 0 & S_- \end{bmatrix}$$

and one finds $X = UV_{21}V_{11}^{-1}JU^T$.

The operation count is the same as for the continuous-time case.

4.3 Balanced Stochastic Truncation

A classic relative error approximation algorithm is Balanced Stochastic Truncation (BST). In this section, it is shown that the algebraic Riccati equations without constant term encountered in the previous section are instrumental in obtaining a BST as well.

4.3.1 Continuous-time systems

A Balanced Stochastic Realization (BSR) is found by balancing P and Q_w computed from the following equations [233] :

$$\begin{aligned} AP + PA^T + BB^T &= 0 \\ B_w \triangleq PC^T + BD^T & \\ A^TQ_w + Q_wA + (C^T - Q_wB_w)(DD^T)^{-1}(C - B_w^TQ_w) &= 0 \qquad (4.9) \end{aligned}$$

where the Q_w is chosen that makes $W(s) \triangleq D_w + C_w(sI - A)^{-1}B_w$ (with $C_w \triangleq D^{-1}(C - B_w^TQ_w)$ and $D_w \triangleq D^T$) minimum phase.

The standard procedure to obtain the solution Q_w is to find a basis $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$ for the anti-stable eigenspace of a Hamiltonian [168], i.e.

$$H = \begin{bmatrix} (A - B_w(DD^T)^{-1}C)^T & C^T(DD^T)^{-1}C \\ -B_w(DD^T)^{-1}B_w & -(A - B_w(DD^T)^{-1}C) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \Lambda$$

where Λ is an anti-stable matrix. If X_2 is invertible, it is easy to check that $X_1X_2^{-1}$ solves (4.9) and makes $W(s)$ minimum phase. The solution of (4.9) that makes $W(s)$ minimum phase is the minimal one, i.e. $Q_w = Q_{min}$, where

$Q_{min} \leq Q$ for all solutions Q of (4.9) [121]. Thus a Balanced Stochastic Realization is a realization $\left[TAT^{-1}, TB, CT^{-1}, D\right]$ such that $TPT^T = T^{-T}QT_w^{-1} = \Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$, with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0$. The σ_i, $i = 1, \ldots, n$, are the *relative Hankel singular values*. It is shown [120] that $\sigma_1 \leq 1$, and that if $G(s)$ has p non-minimum phase zeroes, then $\sigma_1 = \sigma_2 = \ldots = \sigma_p = 1$.

Balanced Stochastic Truncation (BST) is the reduction method where a BSR is truncated, as in Balanced Truncation, according to the largest relative Hankel singular values. Again, it is not necessary to actually balance P and Q_w to obtain the reduced order system $G_r(s)$; one may use Safonov's Schur method [234].

An error bound, due to Green, is available for BST [121] :

$$\|G^{-1}(G - G_r)\|_{L_\infty} \leq -1 + \prod_{i=r+1}^{n} \frac{1+\sigma_i}{1-\sigma_i} \,.$$

An alternative procedure for BST is now presented, in which another ARE has to be solved, but this time an ARE without constant term; this is possible basically because one solution of the Riccati equation (4.9) is already known, i.e. $Q_w = P^{-1}$. This can be exploited to reduce the number of operations needed to solve such an equation. How to find all solutions of an algebraic Riccati equation, when one solution is already available, by solving an ARE without constant term, is well known, see e.g. [19]. Yet this is never taken advantage of for computing a BSR; it is remarkable that when doing so, one arrives at ARE (4.2) which also appears in IOWBT.

Theorem 4.4
Let $X \triangleq Q_w^{-1} - P$, *where* Q_w *solves* (4.9), *then*

$$A_i X + X A_i^T + X C_i^T C_i X = 0 \,. \tag{4.2}$$

Conversely any $Q_w = (P + X)^{-1}$ *solves* (4.9).

Proof : Note that the formulation of the theorem presupposes that Q_w is invertible, i.e. that $G(s)$ is observable.

$$\begin{aligned}
0 &= A^T Q_w + Q_w A + (C^T - Q_w B_w)(DD^T)^{-1}(C - B_w^T Q_w) \\
&= A^T Q_w + Q_w A + (Q_w P C_i^T - C_i^T - Q_w B)(C_i P Q_w - C_i - B^T Q_w) \\
&= A_i^T Q_w + Q_w A_i + (I - Q_w P) C_i^T C_i (I - P Q_w) \\
&\qquad - Q_w (B C_i P + P C_i^T B^T - B B^T) Q_w \\
&= A_i^T Q_w + Q_w A_i + (I - Q_w P) C_i^T C_i (I - P Q_w) - Q_w (A_i P + P A_i^T) Q_w \\
&= (I - Q_w P) A_i^T Q_w + Q_w A_i (I - P Q_w) + (I - Q_w P) C_i^T C_i (I - P Q_w) \\
&= Q_w (X A_i^T + A_i X + X C_i^T C_i X) Q_w \qquad \square
\end{aligned}$$

In the sequel, we will frequently use the fact that $I - PQ_w = XQ_w$ and $I - Q_wP = Q_wX$.

$Q_w = (P+X)^{-1}$ yields a minimum phase $W(s)$ if the anti-stabilizing solution X to equation (4.2) is chosen, i.e. if X is chosen such that $A_i + XC_i^TC_i$ has only RHP eigenvalues. Indeed,

Lemma 4.1 *The zeroes of $W(s)$ are the eigenvalues of $-(A_i + XC_i^TC_i)$.*

Proof : The eigenvalues of $A - B_wD_w^{-1}C_w$ are the zeroes of $W(s)$, and

$$\begin{aligned}
&A - B_wD_w^{-1}C_w \\
&\quad= A - (PC^T + BD^T)D^{-T}D^{-1}(C - CPQ_w - DB^TQ_w) \\
&\quad= A - (PC_i^T - B)(C_i(I - PQ_w) + B^TQ_w)) \\
&\quad= A - PC_i^TC_iXQ_w + BB^TQ_w - PC_i^TB^TQ_w + BC_iXQ_w \\
&\quad= A + BC_iXQ_w - PC_i^TC_iXQ_w - (AP + PA^T)Q_w - PC_i^TB^TQ_w \\
&\quad= A(I - PQ_w) + BC_iXQ_w - P(A^T + C_i^TB^T + C_i^TC_iX)Q_w \\
&\quad= A_iXQ_w + (X - Q_w^{-1})(A_i^T + C_i^TC_iX)Q_w \\
&\quad= (A_iX + XA_i^T + XC_i^TC_iX)Q_w - Q_w^{-1}(A_i^T + C_i^TC_iX)Q_w
\end{aligned}$$

the first term of which is zero, and hence the zeroes of $W(s)$ are the eigenvalues of $-(A_i + XC_i^TC_i)$. $\square$

Summing up, P and Q_w for BST may be found by solving $AP + PA^T + BB^T = 0$ for P, finding the anti-stabilizing solution of $A_iX + XA_i^T + XC_i^TC_iX = 0$, and computing $Q_w \triangleq (P+X)^{-1}$. Again, Safonov's procedure [233] may then be used to obtain the BST reduced order model.

Special cases

Evidently, as the same Riccati equation needs to be solved as in IOWBT, the same special solutions $X = 0$ and $X = Q_i^{-1}$ are available. But as now the *anti-stabilizing* solution is required, $X = 0$ will be found for maximum-phase systems, and $X = Q_i^{-1}$ corresponds to the minimum-phase case.

When $p = n$, $X = 0$ yields $Q_w = (P+X)^{-1} = P^{-1}$ and $PQ_w = I$, hence no dominant eigenspaces for PQ_w can be found, or in other words PQ_w contains no information on which states should be retained. BST is totally senseless in this case.

When $p = 0$, $X = Q_i^{-1}$ yields $Q_w = (P + Q_i^{-1})^{-1} = Q_i(I + PQ_i)^{-1}$ and $PQ_w = PQ_i(I + PQ_i)^{-1}$. Recall that for this case, IOWBT balances P and $Q =$

Q_i, hence projects on the dominant eigenspaces of $PQ_i = PQ_w(I - PQ_w)^{-1}$. Suppose $PQ_w = Y\Sigma^2 Y^{-1}$, then clearly $PQ_i = Y\Sigma^2(I-\Sigma^2)^{-1}Y^{-1} = YM^2Y^{-1}$, where $M \stackrel{\triangle}{=} \mathrm{diag}(\mu_1, \ldots, \mu_n)$; the numbers μ_i could be coined the *IOWBT Hankel singular values*. As $\sigma^2(1-\sigma^2)^{-1}$ increases monotonously as a function of σ, $\sigma > 0$, PQ_w and PQ_i have the same dominant eigenspaces, i.e. IOWBT and BST are equivalent in this case. This is precisely the result of [321, 324]. The relation between the IOWBT Hankel singular values μ_i and the relative Hankel singular values σ_i is clearly

$$\sigma_i = \frac{\mu_i}{\sqrt{1+\mu_i^2}} \; .$$

Hence the BST error bound in terms of the IOWBT Hankel singular values is

$$\begin{aligned} \|G^{-1}(G-G_r)\|_{L_\infty} &\leq -1 + \prod_{i=r+1}^{n} \frac{\sqrt{1+\mu_i^2+\mu_i}}{\sqrt{1+\mu_i^2-\mu_i}} \\ &= -1 + \prod_{i=r+1}^{n} (\sqrt{1+\mu_i^2}+\mu_i)^2 \; . \end{aligned}$$

Finding the anti-stabilizing solution

Here a basis for the stable eigenspace of the Hamiltonian is needed, i.e.

$$\begin{bmatrix} A_i & 0 \\ -C_i^T C_i & -A_i^T \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \Lambda \; ,$$

with Λ a stable matrix.

Thus, the anti-stabilizing solution X may be obtained as in Section 4.2.1, using the Schur decomposition (4.4) and which this time has to be reordered such that all diagonal elements with negative real parts are in the left upper block of the Schur matrix. Of course one should do the preliminary reordering of S, the Schur matrix of A_i, exactly the other way round here, i.e. such that its $n-p$ stable eigenvalues are all in the upper left block; in that way, the number of subsequent swaps needed to reorder the Schur matrix $\begin{bmatrix} -JS^TJ & -JU^TC_i^TC_iU \\ 0 & S \end{bmatrix}$ is reduced to $(n-p)^2$. This procedure has the advantage of being numerically very stable, since the reordering is done using Givens rotations only. On the other hand, where for IOWBT one needed to do but few swaps for systems with few non-minimum phase zeroes, here systems with a small p require more reordering. It so happens that most systems encountered in practice, especially those modeling controllers, have more LHP than RHP transmission zeroes.

One may benefit, however, from the knowledge of both special solutions, $X = 0$ and $X = Q_i^{-1}$; indeed this means that $\begin{bmatrix} 0 \\ X_2 \end{bmatrix}$ and $\begin{bmatrix} X_2 \\ Q_i X_2 \end{bmatrix}$, with X_2 any invertible matrix, form bases for eigenspaces of the Hamiltonian, with which the Hamiltonian may be block diagonalized. Assume a Schur decomposition $A_i = USU^T$, where this time S has all its LHP eigenvalues in the upper left block, and let J be the same permutation matrix as before, and $\bar{Q}_i$ solve $S^T\bar{Q}_i + \bar{Q}_i S + U^T C_i^T C_i U = 0$, then

$$\begin{bmatrix} A_i & 0 \\ -C_i^T C_i & -A_i^T \end{bmatrix} \begin{bmatrix} U & 0 \\ U\bar{Q}_i & UJ \end{bmatrix} = \begin{bmatrix} U & 0 \\ U\bar{Q}_i & UJ \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & -JS^TJ \end{bmatrix}$$

$$\begin{bmatrix} A_i & 0 \\ -C_i^T C_i & -A_i^T \end{bmatrix} \begin{bmatrix} U & 0 \\ U\bar{Q}_i & UJ \end{bmatrix} V = \begin{bmatrix} U & 0 \\ U\bar{Q}_i & UJ \end{bmatrix} V \begin{bmatrix} S_- & 0 \\ 0 & S_+ \end{bmatrix}$$

with $V \triangleq \begin{bmatrix} I_{n-p} & 0 & 0 & 0 \\ 0 & 0 & I_p & 0 \\ 0 & I_p & 0 & 0 \\ 0 & 0 & 0 & I_{n-p} \end{bmatrix}$, and where S_- has only LHP eigenvalues. Note that this is no longer a Schur decomposition of the Hamiltonian, but a block diagonalization using non-orthogonal transformations; hence the method may be numerically less reliable. On the other hand, it is cheaper, since $\bar{Q}_i$ may be computed in approximately n^3 flops.

Of course a basis for the stable eigenspace can be immediately read off, and the anti-stabilizing solution is $X = X_1 X_2^{-1}$, where

$$X_1 \triangleq \begin{bmatrix} U \begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & 0 \end{bmatrix} \text{ and } X_2 \triangleq \begin{bmatrix} U\bar{Q}_i \begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & UJ \begin{bmatrix} I_p \\ 0 \end{bmatrix} \end{bmatrix}.$$

It is evident from these expressions that X_1 and therefore also X has rank $n-p$. For non-minimal systems Q_w^{-1} and hence X may not exist; this corresponds to a singular X_2. In that case PQ_w may be computed directly as $PQ_w = PX_1(PX_1 + X_2)^{-1}$.

The block-diagonalizing approach is equivalent to exploiting the fact that $Q_w = Q_i(I + PQ_i)^{-1}$ and $Q_w = P^{-1}$ are solutions to the BSR Riccati equation (4.9), and hence that $\begin{bmatrix} Q_i U \\ U + Q_i PU \end{bmatrix}$ and $\begin{bmatrix} U \\ PU \end{bmatrix}$ form eigenspaces for the Hamiltonian H connected to (4.9) :

$$\begin{bmatrix} (A-B_w(DD^T)^{-1}C)^T & C^T(DD^T)^{-1}C \\ -B_w(DD^T)^{-1}B_w & -(A-B_w(DD^T)^{-1}C) \end{bmatrix} \begin{bmatrix} Q_iU & U \\ (I+Q_iP)U & PU \end{bmatrix}$$
$$= \begin{bmatrix} Q_iU & U \\ (I+Q_iP)U & PU \end{bmatrix} \begin{bmatrix} -S & 0 \\ 0 & S^* \end{bmatrix} .$$

Note that the block-diagonalizing approach may just as well be applied to IOWBT, only in general there is less need for it there, since most often but few (p^2) swaps are needed for the subsequent reordering.

4.3.2 Discrete-time systems

A discrete-time BSR results from balancing P and Q_w found from the following equations (see [78]) :

$$\begin{aligned} P &= APA^T + BB^T \\ B_w &\triangleq APC^T + BD^T \\ R &\triangleq DD^T + CPC^T \\ \tilde{A} &\triangleq A - B_wR^{-1}C \end{aligned}$$

$$Q_w - \tilde{A}^TQ_w\tilde{A} + \tilde{A}^TQ_wB_w(B_w^TQ_wB_w - R)^{-1}B_w^TQ_w\tilde{A} - C^TR^{-1}C = 0 \quad (4.10)$$

where Q_w is chosen such that $W(z) \triangleq D_w + C_w(zI-A)^{-1}B_w$ is minimum phase, with $D_w \triangleq (R - B_w^TQ_wB_w)^{\frac{1}{2}}$ and $C_w \triangleq D_w^{-T}(C - B_w^TQ_wA)$.

The main result of this subsection is the fact that again, the same Riccati equation (4.6) as in Subsection 4.2.2 is found when solving the BSR equations by means of an ARE without a constant term. This is expressed in the following

Theorem 4.5
Let $X \triangleq P - Q_w^{-1}$, *where* Q_w *solves* (4.10)*,*

Then $$A_iXC_i^T(I+C_iXC_i^T)^{-1}C_iXA_i^T - A_iXA_i^T + X = 0 . \quad (4.6)$$

Proof : Equation (4.10) is equivalent to

$$Q_w^{-1} - \tilde{A}Q_w^{-1}\tilde{A}^T + \tilde{A}Q_w^{-1}C^T(CQ_w^{-1}C^T - R)^{-1}CQ_w^{-1}\tilde{A}^T - B_wR^{-1}B_w^T = 0 ,$$

or, defining $B_{wi} \triangleq B - APC_i^T = B_wD^{-T}$ and $R_i \triangleq I + C_iPC_i^T$,

$$Q_w^{-1} - \tilde{A}Q_w^{-1}\tilde{A}^T + \tilde{A}Q_w^{-1}C_i^T(C_iQ_w^{-1}C_i^T - R_i)^{-1}C_iQ_w^{-1}\tilde{A}^T - B_{wi}R_i^{-1}B_{wi}^T = 0 . \quad (4.11)$$

This equation can also be written as

$$Q_w^{-1} - B_{wi}R_i^{-1}B_{wi}^T = \tilde{A}(Q_w - C_i^T R_i^{-1} C_i)^{-1}\tilde{A}^T \ . \tag{4.12}$$

Now $B_{wi}R_i^{-1}B_{wi}^T = (B - \tilde{A}PC_i^T)R_iR_i^{-1}R_i(B - \tilde{A}PC_i^T)^T$

$$\begin{aligned}
&= BB^T - \tilde{A}PC_i^T R_i B^T - BR_iC_iP\tilde{A}^T + \tilde{A}PC_i^T R_i C_i P\tilde{A}^T + BC_iPC_i^T B^T \\
&= P - APA^T - A_iPC_i^T B^T - BC_iPA_i^T + BC_iPC_i^T B^T + A_iPC_i^T R_i^{-1} C_i P A_i^T \\
&= P - A_iPA_i^T + A_iPC_i^T R_i^{-1} C_i P A_i^T \\
&= P - \tilde{A}(I + PC_i^T C_i)P(I - C_i^T R_i^{-1} C_i P)(I + C_i^T C_i P)\tilde{A}^T \\
&= P - \tilde{A}(P + PC_i^T C_i P)\tilde{A}^T \\
&= P - \tilde{A}(P^{-1} - C_i^T R_i^{-1} C_i)^{-1}\tilde{A}^T \ ,
\end{aligned}$$

such that $Q_w^{-1} = P$ is seen to be a solution of (4.11). This time define

$$\begin{aligned}
X &\triangleq P - Q_w^{-1} \\
&= \tilde{A}(P^{-1} - C_i^T R_i^{-1} C_i)^{-1}\tilde{A}^T - \tilde{A}(Q_w - C_i^T R_i^{-1} C_i)^{-1}\tilde{A}^T \\
&= \tilde{A}(Q_w - C_i^T R_i^{-1} C_i)^{-1}(P^{-1} - Q_w)(P^{-1} - C_i^T R_i^{-1} C_i)^{-1}\tilde{A}^T \\
&= \tilde{A}(I - (P - X)C_i^T R_i^{-1} C_i)^{-1} X (I - C_i^T R_i^{-1} C_i P)^{-1}\tilde{A}^T \\
&= \tilde{A}(I - PC_i^T R_i^{-1} C_i)^{-1} . \\
&\quad (I + XC_i^T R_i^{-1} C_i (I - PC_i^T R_i^{-1} C_i)^{-1})^{-1} X (I - C_i^T R_i^{-1} C_i P)^{-1}\tilde{A}^T \ .
\end{aligned}$$

Now $I - PC_i^T R_i^{-1} C_i = I - PC_i^T (I + C_iPC_i^T)^{-1} C_i = (I + PC_i^T C_i)^{-1}$, and

$$\tilde{A}(I - PC_i^T R_i^{-1} C_i)^{-1} = \tilde{A}(I + PC_i^T C_i) = (A + B_{wi}R_i^{-1}C_i)(I + PC_i^T C_i)$$

$$= (A + (B - APC_i^T)C_i(I + PC_i^T C_i)^{-1})(I + PC_i^T C_i) = A + BC_i = A_i \ . \tag{4.13}$$

Moreover,

$$C_i^T R_i^{-1} C_i (I - PC_i^T R_i^{-1} C_i)^{-1} = C_i^T C_i (I + PC_i^T C_i)^{-1}(I + PC_i^T C_i) = C_i^T C_i \ , \tag{4.14}$$

so that the Riccati equation is finally rewritten as :

$$\begin{aligned}
X &= A_i(I + XC_i^T C_i)^{-1} X A_i^T \\
&= A_i(I - XC_i^T (I + C_iXC_i^T)^{-1} C_i) X A_i^T \\
&= A_iXA_i^T - A_iXC_i^T (I + C_iXC_i^T)^{-1} C_i X A_i^T \ . \qquad \square
\end{aligned}$$

It remains to be shown which choice of X ensures that $W(z)$ is minimum phase. For this, we have the following

Lemma 4.2
The zeroes of $W(z)$ are the inverses of the eigenvalues of $(I + XC_i^T C_i)^{-1} A_i$.

Proof :

$$
\begin{aligned}
&A - B_w D_w^{-1} C_w \\
&= A - B_w (D_w D_w^T)^{-1} (C - B_w^T Q_w A) \\
&= A - B_w (R - B_w^T Q_w B_w)^{-1} (C - B_w^T Q_w A) \\
&= A - (I - B_w R^{-1} B_w^T Q_w)^{-1} B_w R^{-1} (C - B_w^T Q_w A) \\
&= (I - B_w R^{-1} B_w^T Q_w)^{-1} (A - B_w R^{-1} C) \\
&= Q_w^{-1} (Q_w^{-1} - B_w R^{-1} B_w^T)^{-1} \tilde{A} \\
&= Q_w^{-1} \tilde{A}^{-T} (I - C_i^T R_i^{-1} C_i Q_w^{-1}) Q_w \qquad \text{(see (4.12))} \\
&= Q_w^{-1} \tilde{A}^{-T} (I - C_i^T R_i^{-1} C_i P + C_i^T R_i^{-1} C_i X) Q_w \\
&= Q_w^{-1} \tilde{A}^{-T} (I - C_i^T R_i^{-1} C_i P)(I + (I - C_i^T R_i^{-1} C_i P)^{-1} C_i^T R_i^{-1} C_i X) Q_w \\
&= Q_w^{-1} A_i^{-T} (I + C_i^T C_i X) Q_w \qquad \text{(see (4.13), (4.14))} \\
&= Q_w^{-1} ((I + X C_i^T C_i)^{-1} A_i)^{-T} Q_w \qquad \square
\end{aligned}
$$

Hence the zeroes of $W(z)$ are all inside the unit disk, if $(I + XC_i^T C_i)^{-1} A_i$ is anti-stable, and a BSR is found from balancing P and $Q_w = (P - X)^{-1}$.

Finding the anti-stabilizing solution

The same special solutions are found as in the continuous-time case, and the same remarks apply; the equivalence of IOWBT and BST for minimum-phase systems [320] becomes evident in the same way.

For the general case, again assume a Schur decomposition $A_i = USU^T$, where this time the Schur matrix S has all its eigenvalues lying inside the unit disk in the upper left block, and let $\bar{Q}_i$ solve $S^T \bar{Q}_i S - \bar{Q}_i + U^T C_i^T C_i U = 0$, then

$$
\begin{bmatrix} I & C_i^T C_i \\ 0 & A_i \end{bmatrix}
\begin{bmatrix} -U\bar{Q}_i & UJ \\ U & 0 \end{bmatrix}
=
\begin{bmatrix} A_i^T & 0 \\ 0 & I \end{bmatrix}
\begin{bmatrix} -U\bar{Q}_i & UJ \\ U & 0 \end{bmatrix}
\begin{bmatrix} S & 0 \\ 0 & JS^{-*}J \end{bmatrix}
$$

$$
\begin{aligned}
&\begin{bmatrix} I & -C_i^T C_i \\ 0 & A_i \end{bmatrix}
\begin{bmatrix} U\bar{Q}_i & UJ \\ U & 0 \end{bmatrix}
\begin{bmatrix} I_{n-p} & 0 \\ 0 & 0 \\ 0 & I_p \\ 0 & 0 \end{bmatrix} \\
&\qquad =
\begin{bmatrix} A_i^T & 0 \\ 0 & I \end{bmatrix}
\begin{bmatrix} U\bar{Q}_i & UJ \\ U & 0 \end{bmatrix}
\begin{bmatrix} I_{n-p} & 0 \\ 0 & 0 \\ 0 & I_p \\ 0 & 0 \end{bmatrix} S_<
\end{aligned}
$$

where $|\text{eig}(S_<)| < 1$. Then, defining

$$X_1 \triangleq \left[\begin{array}{cc} -U\bar{Q}_i \left[\begin{array}{c} I_{n-p} \\ 0 \end{array} \right] & UJ \left[\begin{array}{c} I_p \\ 0 \end{array} \right] \end{array} \right], \text{ and } X_2 \triangleq \left[\begin{array}{cc} U \left[\begin{array}{c} I_{n-p} \\ 0 \end{array} \right] & 0 \end{array} \right],$$

$X \triangleq X_2 X_1^{-1}$ is the anti-stabilizing solution of (4.6), i.e. the solution for which $(I + XC_i^T C_i)^{-1} A_i$ is anti-stable.

4.4 Comparison

In this section, IOWBT is compared with its main competitor BST, with respect to computational cost, relative error, location of non-minimum phase zeroes after reduction, and reduction of systems with too many non-minimum phase zeroes.

4.4.1 Operation count

The operation counts below are based on the assumption that the computation of a Schur decomposition costs $25n^3$ flops; this value is of course only approximate [116, p. 380]. A reordering swap of two adjacent eigenvalues in an $n \times n$ Schur matrix requires $12n$ flops.
$2n^3$ flops are needed to solve a Sylvester equation once the coefficient matrices have been put in Schur form (n^3 flops for a Lyapunov equation).

Note that the operation counts are the same for the continuous-time and the discrete-time case.

The two Lyapunov equations in the IOWBT procedure require a Schur decomposition to be performed, as well as the eigenspace computation of PQ, and a Schur decomposition of A_i is needed, i.e. about $25n^3 \times 4 = 100n^3$ flops are needed there. The preliminary reordering of the Schur decomposition of A_i costs at most $12np(n-p)$ flops, and the p^2 subsequent swaps in the $2n \times 2n$ Schur matrix of the Hamiltonian $24np^2$ flops; hence the total reordering cost is maximally $12np(n+p)$ flops, i.e. between 0 and $24n^3$ flops. The other operations add up to about $10n^3$ flops, so that the total for IOWBT is between $110n^3$ (minimum-phase systems) and $135n^3$ flops (maximum-phase systems).

If the block-diagonalizing scheme of Subsection 4.3.1 were employed for IOWBT, this would suppress the $24np^2$ flops needed for the final reordering of the Hamiltonian Schur matrix, while n^3 flops need to be added for block diagonalization; the operation count is then independent of p and approximately $110n^3$ flops. Thus, for 'nearly minimum-phase' systems (p small), the difference would hardly be felt.

The standard algorithm for Balanced Stochastic Truncation [233] requires a Schur decomposition of a $2n \times 2n$ matrix, i.e. $25(2n)^3 = 200n^3$ flops, a Schur decomposition of the $n \times n$ matrix PQ_w, i.e. $25n^3$ flops, and a Lyapunov equation to be solved, i.e. $25n^3$ flops. Therefore this method needs roughly $250n^3$ flops, i.e. about the double of the method above.

For the BST procedures, the Schur decompositions of A_i and those needed for the computation of P, the eigenspace computation of PQ, are the most costly operations, adding up to about $75n^3$ flops. The other operations, including the 'preliminary' reordering of S, add up to approximately $15n^3$, giving a total of about $90n^3$ flops for the block-diagonalizing scheme. This is about three times less than the conventional procedure.

In case the Schur matrix of the Hamiltonian is reordered using orthogonal transformations, another $24n(n-p)^2$ flops should be thrown in; thus for this method the flop count ranges from about $110n^3$ flops for minimum phase systems to $90n^3$ flops for maximum phase systems.

4.4.2 Error bound

As was mentioned above, BST has a useful H_∞ error bound, due to Green [121].

A bound on the error has not been found for IOWBT, except for the case where $p = 0$, for which the method is equivalent to BST, and therefore a bound is available equivalent to Green's bound for BST.

Naturally, one might try and see if the result of [320, 321] extends to the general case and formulate the following conjecture :
If μ_i^2 are the eigenvalues of PQ, with $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_n \geq 0$, then

$$\|G(s)^{-1}(G(s) - G_r(s))\|_\infty \leq -1 + \prod_{i=r+1}^{n} (\sqrt{1+\mu_i^2} + \mu_i)^2 \ .$$

A test using randomly generated non-minimum phase systems however, shows that the would-be bound is occasionally violated (see figure 4.2, where the ratio of the actual error to the conjectured bound is plotted for 50 systems).

The system for which the bound is violated is

$$G(s) \triangleq \frac{\sum_{i=0}^{8} b_i s^i}{\sum_{i=0}^{8} a_i s^i}$$

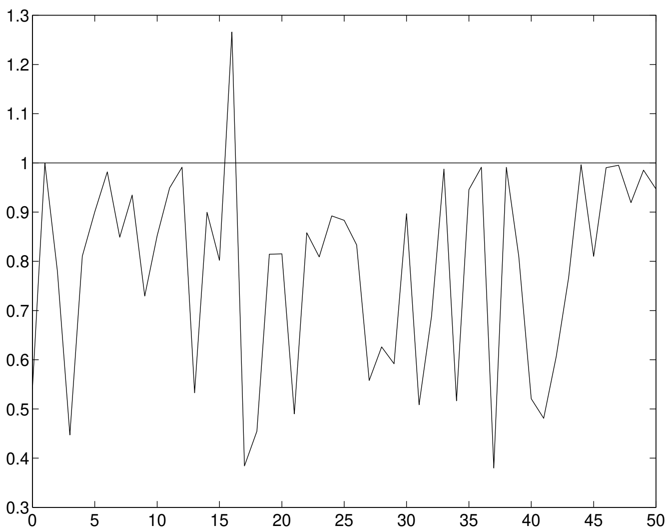


Figure 4.2: The ratio of the actual error of IOWBT to the conjectured bound shows that this bound is not valid.

$$\text{with} \quad \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \end{bmatrix} = \begin{bmatrix} -0.002 \\ 0.043 \\ 0.223 \\ 0.566 \\ 1.426 \\ 2.948 \\ 3.596 \\ 2.370 \\ 0.708 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix} = \begin{bmatrix} 0.0007 \\ 0.056 \\ 0.405 \\ 1.341 \\ 3.093 \\ 5.203 \\ 5.620 \\ 3.500 \\ 1.000 \end{bmatrix}$$

where the reduced order is 3; the error $\|G(s)^{-1}(G(s) - G_r(s))\|_\infty = 0.906$ exceeds the conjectured bound, which is equal to 0.706. As the would-be bound is so rarely violated, one might suspect Matlab's H_∞ norm routine to be in error, which in fact it is prone to be for nearly non-minimal systems; but this is certainly not the case here, as a Bode plot of the violating system shows (see figure 4.3).

It has been mentioned that BST and IOWBT are equivalent for minimum-phase systems, and it is argued below that they are very different when $r < p$, e.g.

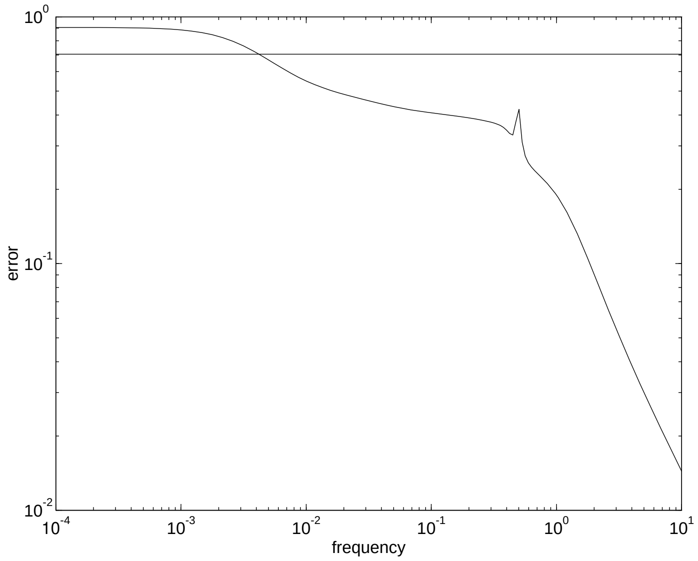


Figure 4.3: Conjectured bound and approximation error Bode plot for the exception.

for maximum-phase systems. In general, at least for systems with relatively small p, BST and IOWBT tend to yield systems with very similar relative errors $\|G^{-1}(G-G_r)\|_\infty$. Remark that the relative error plots coincide for $r=4$; for very low orders the errors differ more (see Subsection 4.4.4).

4.4.3 Transmission zeroes

It has been shown [120] that BST preserves the non-minimum phase zeroes, i.e. if $z_+ \in \mathbb{C}_+$ is a transmission zero of $G(s)$, and $G_b(s)$ is the reduced order model of order r, with $r \geq p$, obtained by BST on $G(s)$, then z_+ is a transmission zero of $G_b(s)$. This may be an interesting feature, but on the other hand for many applications this property will be irrelevant, and may be regarded as an unnecessarily stringent condition.

IOWBT does not have this feature; though, not surprisingly, the non-minimum phase zero locations often tend to be rather close to the original zeros, they do not coincide.

4.4.4 Too many non-minimum phase zeroes

IOWBT is superior to BST whenever $r < p$, i.e. when the desired order reduction is larger than the number of non-minimum phase transmission zeroes. In this case it may still yield an acceptable fit, whereas BST fails. This will be illustrated in an example below.

It should be noted that the case $r < p$ is one where no sensible H_∞ error can be obtained. Indeed, p of the relative Hankel singular values σ_i equal one [120], and a well-known lower bound [121] for the relative error when approximating with a system G_r of order at most r is

$$\min_{G_r} \|G^{-1}(G-G_r)\|_\infty \geq \sigma_{r+1} \ .$$

Therefore, for $r < p$, $\min_{G_r} \|G^{-1}(G-G_r)\|_\infty \geq 1$. That there is such a large lower bound is not surprising : e.g., for SISO systems, the phase of G_r varies over at most $r\pi$, while the phase of G, which has p non-minimum phase zeroes, will vary over $p\pi$; hence the relative error must be large in some frequency range. The minimum is obtained e.g. by $G_r = 0$; it does not follow, however, that this is the best possible fit; the L_∞ norm is but one criterion with which one can measure the relative error, and apparently for these particular problems, not a very useful one.

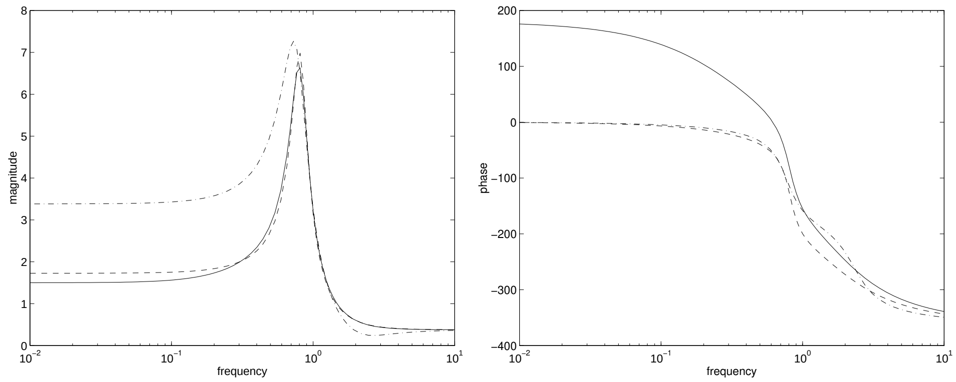


Figure 4.4: 'Maximum-phase' system
Full line : original system, dashed line : IOWBT, dash-dotted line : BST .
left : magnitude Bode plots right : phase Bode plot

Consider the system

$$G(s) \triangleq \frac{3s^3 - 9s^2 + 13s - 3}{8s^3 + 5s^s + 6s + 2}$$

$G(s)$ is stable and 'maximum phase' : it has poles at $s = -0.1222 \pm 0.8013j$ and $s = -0.3805$, and transmission zeroes at $s = 1.3600 \pm 1.3123j$ and $s = 0.28$. With $r = 2$, the reduced order model G_r obtained by IOWBT gives a relative error $\|G^{-1}(G - G_r)\|_\infty = 2.151$, whereas the model G_b found with BST yields $\|G^{-1}(G - G_b)\|_\infty = 3.255$; both models seem useless, since $\|G^{-1}(G - 0)\|_\infty = 1$.

It is instructive, however, to look at the Bode plots of G, G_r and G_b, which are shown in Fig. 4.4; as expected, the phases are far off for some frequencies, but for the magnitude Bode plot, IOWBT gives a much better fit than BST.

That the BST method does not do well when $r < p$ is hardly surprising : in this case one tries to choose the r independent eigenvectors of PQ_w corresponding to the largest eigenvalues. As in BST, PQ_w has its p largest eigenvalues equal to one [120], one should not expect that BST can determine sensibly which r eigenvectors of these p dominant ones it should choose.

IOWBT does not suffer from this, since there is no such simple connection between the weighted Hankel singular values μ_i and the number of non-minimum phase zeroes here.

4.5 Procedures

For easy reference, we recapitulate the model reduction procedures described above, as well as their duals.

4.5.1 Continuous-time

IOWBT

1. Solve $AP + PA^T + BB^T = 0$ for P
2. Compute $A_i \triangleq A - BD^{-1}C$, $B_i \triangleq BD^{-1}$, $C_i \triangleq -D^{-1}C$.
3. Compute a Schur decomposition $A_i = USU^T$.
4. Reorder the Schur matrix $\begin{bmatrix} -JS^TJ & -JU^TC_i^TC_iU \\ 0 & S \end{bmatrix}$ of the Hamiltonian using an orthogonal transformation $V = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}$:
$$\begin{bmatrix} -JS^TJ & -JU^TC_i^TC_iU \\ 0 & S \end{bmatrix} \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} S_+ & S_{12} \\ 0 & S_- \end{bmatrix}$$
where S_+ has only RHP eigenvalues.
5. Compute $X = UV_{21}V_{11}^{-1}JU^T$.
6. Compute $A_{wi} \triangleq A_i + XC_i^TC_i$
7. Solve $A_{wi}^TQ + QA_{wi} + C_i^TC_i = 0$ for Q.

   We use Safonov's Schur method [233] to do the balanced truncation to a system of order r :

8. Compute matrices V_r and V_l, whose r columns form bases for the right and left eigenspaces of PQ associated with the r largest eigenvalues.
9. Compute a singular value decomposition $U_e\Sigma_eV_e^T = V_l^TV_r$
10. Defining $L \triangleq \Sigma_e^{-\frac{1}{2}}U_e^TV_l^T$ and $T \triangleq V_rV_e\Sigma_e^{-\frac{1}{2}}$, $A_r \triangleq LAT$, $B_r \triangleq LB$, $C_r \triangleq CT$, the reduced order model is $G_r(s) \triangleq [A_r, B_r, C_r, D]$.

Dual :
This version makes $\|(G(s) - G_r(s))G(s)^{-1}\|_\infty$ small, by using the inverse right outer factor $V(s)$ (which satisfies $V^T(-s)V(s) = G^T(-s)G(s)$) as a weighting function on the input side.

1. Solve $A^TQ + QA + C^TC = 0$ for Q.
2. Compute $A_i \triangleq A - BD^{-1}C$, $B_i \triangleq BD^{-1}$, $C_i \triangleq -D^{-1}C$.
3. Compute a Schur decomposition $A_i^T = USU^T$.
4. Reorder the Schur matrix of the Hamiltonian using an orthogonal transformation V
$$\begin{bmatrix} -JS^TJ & -JU^TB_iB_i^TU \\ 0 & S \end{bmatrix} \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} S_+ & S_{12} \\ 0 & S_- \end{bmatrix}$$
where S_+ has only RHP eigenvalues.
5. Compute $X = UV_{21}V_{11}^{-1}JU^T$.
6. Compute $A_{wi} \triangleq A_i + B_iB_i^TX$
7. Solve $A_{wi}P + PA_{wi}^T + B_iB_i^T = 0$ for P.
8. Execute steps 8, 9 and 10 as above.

BST

1. Solve $AP + PA^T + BB^T = 0$ for P
2. Compute $A_i \triangleq A - BD^{-1}C$, $C_i \triangleq -D^{-1}C$.
3. Compute a Schur decomposition $A_i = USU^T$, ordered such that the stable eigenvalues are in the upper left block of S.
Let p be the number of unstable eigenvalues of A_i.
4. Compute $\bar{Q}_i$ from $S^T\bar{Q}_i + \bar{Q}_iS + U^TC_i^TC_iU = 0$
5. $X_1 \triangleq \begin{bmatrix} U\bar{Q}_i \begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & UJ \begin{bmatrix} I_p \\ 0 \end{bmatrix} \end{bmatrix}$, $X_2 \triangleq \begin{bmatrix} U \begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & 0 \end{bmatrix}$,
$X \triangleq X_2X_1^{-1}$ and $Q \triangleq (P + X)^{-1}$
6. Execute steps 8, 9 and 10 of the procedure for IOWBT.

Dual :

1. Solve $A^TQ + QA + C^TC = 0$ for Q
2. Compute $A_i \triangleq A - BD^{-1}C$, $B_i \triangleq BD^{-1}$.
3. Compute a Schur decomposition $A_i^T = USU^T$, ordered such that the stable eigenvalues are in the upper left block of S.
   Let p be the number of unstable eigenvalues of A_i.
4. Compute $\bar{P}_i$ from $S^T\bar{P}_i + \bar{P}_iS + U^TB_iB_i^TU = 0$
5. $X_1 \triangleq \begin{bmatrix} U\bar{P}_i \begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & UJ \begin{bmatrix} I_p \\ 0 \end{bmatrix} \end{bmatrix}$, $X_2 \triangleq \begin{bmatrix} U \begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & 0 \end{bmatrix}$,
   $X \triangleq X_2X_1^{-1}$ and $P \triangleq (Q+X)^{-1}$
6. Execute steps 8, 9 and 10 of the procedure for IOWBT.

4.5.2 Discrete-time

IOWBT

1. Solve $APA^T - P + BB^T = 0$ for P
2. Compute $A_i \triangleq A - BD^{-1}C$, $B_i \triangleq BD^{-1}$, $C_i \triangleq -D^{-1}C$.
3. Compute a Schur decomposition $A_i = USU^T$.
4. Reorder the Schur matrix of the Hamiltonian using an orthogonal transformation $V = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}$:
   $$\begin{bmatrix} JS^{-T}J & JS^{-T}U^TC_i^TC_iU \\ 0 & S \end{bmatrix} \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} S_+ & S_{12} \\ 0 & S_- \end{bmatrix}$$
   where S_+ has only eigenvalues outside the unit disk.
5. Compute $X = UV_{21}V_{11}^{-1}JU^T$.
6. Compute $A_{wi} \triangleq A_i(I + XC_i^TC_i)^{-1}$.
7. Solve $A_{wi}^TQA_{wi} - Q + C_i^TC_i = 0$ for Q.
8. Execute steps 8, 9 and 10 as above.

Dual :
This version makes $\|(G(z)-G_r(z))G(z)^{-1}\|_\infty$ small, by using the inverse right outer factor $V(z)$ $(V^T(z^{-1})V(s) = G^T(z^{-1})G(z))$ as a weighting function on the input side.

1. Solve $A^TQA - Q + C^TC = 0$ for Q.
2. Compute $A_i \triangleq A - BD^{-1}C$, $B_i \triangleq BD^{-1}$, $C_i \triangleq -D^{-1}C$.
3. Compute a Schur decomposition $A_i^T = USU^T$.
4. Reorder the Schur matrix of the Hamiltonian using an orthogonal transformation V :
$$\begin{bmatrix} JS^{-T}J & JS^{-T}U^TC_i^TC_iU \\ 0 & S \end{bmatrix}\begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}\begin{bmatrix} S_+ & S_{12} \\ 0 & S_- \end{bmatrix}$$
where S_+ has only eigenvalues outside the unit disk.
5. Compute $X = UV_{21}V_{11}^{-1}JU^T$.
6. Compute $A_{wi} \triangleq (I + B_iB_i^TX)^{-1}A_i$
7. Solve $A_{wi}PA_{wi}^T - P + B_iB_i^T = 0$ for P.
8. Execute steps 8, 9 and 10 as above.

BST

1. Solve $APA^T - P + BB^T = 0$ for P
2. Compute $A_i \triangleq A - BD^{-1}C$, $C_i \triangleq -D^{-1}C$.
3. Compute a Schur decomposition $A_i = USU^T$, ordered such that the stable eigenvalues are in the upper left block of S.
Let p be the number of unstable eigenvalues of A_i.
4. Compute $\bar{Q}_i$ from $S^T\bar{Q}_iS - \bar{Q}_i + U^TC_i^TC_iU = 0$
5. $X_1 \triangleq \begin{bmatrix} -U\bar{Q}_i\begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & UJ\begin{bmatrix} I_p \\ 0 \end{bmatrix} \end{bmatrix}$, $X_2 \triangleq \begin{bmatrix} U\begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & 0 \end{bmatrix}$,
$X \triangleq X_2X_1^{-1}$ and $Q \triangleq (P - X)^{-1}$
6. Execute steps 8, 9 and 10 as above.

Dual :

1. Solve $A^TQA - Q + C^TC = 0$ for Q

2. Compute $A_i \triangleq A - BD^{-1}C$, $B_i \triangleq BD^{-1}$.

3. Compute a Schur decomposition $A_i^T = USU^T$, ordered such that the stable eigenvalues are in the upper left block of S.
   Let p be the number of unstable eigenvalues of A_i.

4. Compute $\bar{P}_i$ from $S^T\bar{P}_iS - \bar{P}_i + U^TB_iB_i^TU = 0$

5. $X_1 \triangleq \begin{bmatrix} -U\bar{P}_i \begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & UJ\begin{bmatrix} I_p \\ 0 \end{bmatrix} \end{bmatrix}$, $X_2 \triangleq \begin{bmatrix} U\begin{bmatrix} I_{n-p} \\ 0 \end{bmatrix} & 0 \end{bmatrix}$,
   $X \triangleq X_2X_1^{-1}$ and $P \triangleq (Q - X)^{-1}$

6. Execute steps 8, 9 and 10 as above.

4.6 Non-square systems

For non-square systems, the story is a little different. Inverses do not exist, and only right or left inverses can be found, if any; we only consider right or left (causal) invertible systems. These right or left inverses are not unique, hence a relative error cannot be defined as before in a unique way.

Let $G(s)$ be a fat system, i.e. with more inputs than outputs, and assume that $G^\dagger(s)$ is a right inverse, i.e. such that $G(s)G^\dagger(s) = I$. At first sight, $(G(s) - G_r(s))G^\dagger(s)$ may seem a natural definition for the relative error between $G(s)$ and its approximant $G_r(s)$; but this definition does not reflect how close $G(s)$ and $G_r(s)$ are, since $(G^\dagger(s))^T$ has a nonzero null space. A more sensible definition is

$$E_r \triangleq G^\dagger(G - G_r)\ . \tag{4.15}$$

Note that as the right inverse $G^\dagger$ is not unique, neither is E_r. An alternative is to define a relative error Δ_r, which is such that

$$G\Delta_r = G - G_r\ . \tag{4.16}$$

Plugging (4.16) into (4.15) yields $E_r = G^\dagger G\Delta_r$, hence in general $E_r \neq \Delta_r$ in the non-square case; on the other hand, when Δ_r is replaced by E_r, it does satisfy (4.16). Analogous definitions hold for the dual (tall system) case.

4.6.1 Motivation

Even the motivation given for the square case does not quite hold anymore. If $P(s)$ is tall, $C(s)$ is fat, and generically has (many) right inverses but no left inverse. It is possible that $(I+CP)^{-1}CP \approx I$, but this cannot be exploited to approximately reformulate the stability criterion as a relative error constraint, because of the lack of a left inverse of C.
On the other hand, $(I+PC)^{-1}PC \approx I$ cannot hold any longer, as $(I+PC)^{-1}PC$ is not even of full rank. Hence, with $C^\dagger$ a right inverse of C, i.e. $C(s)C^\dagger(s) = I$, one may still write $(I+PC)^{-1}P(C-C_r) = (I+PC)^{-1}PCC^\dagger(C-C_r)$, but this is in general not approximately equal to $C^\dagger(C-C_r)$. However, Theorem 4.1 does still hold, if it is adapted as follows :

Theorem 4.6
Let $P(s)$ *be a* $p \times m$ *plant,* $p > m$*, and* $C(s)$ *an* $m \times p$ *stable controller with right inverse* $C^\dagger(s)$*, such that the 1-DOF control loop is stable.*
Let $\|(I+PC)^{-1}PC\|_\infty = (1+\epsilon)$.
Then *a stable controller* $C_r(s)$ *stabilizes* $P(s)$ *in the 1-DOF setup if*

$$\|C^\dagger(s)(C_r(s) - C(s))\|_{L_\infty} < (1+\epsilon)^{-1} \; .$$

Proof :

$$\begin{aligned}\|W_s(s)(C(s)-C_r(s))\|_\infty &= \|(I+P(s)C(s))^{-1}P(s)(C(s)-C_r(s))\|_\infty \\ \leq \;\; & \|(I+P(s)C(s))^{-1}P(s)C(s)\|_\infty \|C^\dagger(s)(C(s)-C_r(s))\|_{L_\infty} < 1 \qquad \square\end{aligned}$$

It is easily seen, however, that there will be much more conservatism.

Non-square relative error model reduction is also useful for stability-oriented coprime factor model reduction (see Section 2.3).
Indeed, for some standard plant $P(s)$, let $P_{yu}(s)$ have a left coprime factorization $P_{yu} = N(s)(M(s))^{-1}$, and let $C(s)$ be a (possibly unstable) controller stabilizing P; then $C(s)$ has a left coprime factorization $C(s) = (\tilde{V}(s))^{-1}\tilde{U}(s)$ such that $\tilde{V}M - \tilde{U}N = I$ [286, 204]. Hence one possible coprime factor model reduction is to reduce $\tilde{C}(s) \triangleq \begin{bmatrix} \tilde{U} & \tilde{V} \end{bmatrix}$.
Note that with the building blocks $\tilde{V} - I$ and $\tilde{U}$, $C(s) = \tilde{V}^{-1}\tilde{U}$ may be implemented as in Fig. 4.5; thus, as is seen in Fig. 4.6, the feedback loop with P and C is equivalent with the feedback configuration of $\bar{P}$ and $\bar{C}$, with $\bar{P} \triangleq \begin{bmatrix} P \\ -I \end{bmatrix}$ and $\bar{C} \triangleq \begin{bmatrix} \tilde{U} & \tilde{V} - I \end{bmatrix}$. When reducing $\bar{C}$ with a stability-oriented model reduction, the frequency weight of Subsection 2.4.1 is

$$W_s(s) \;\; = \;\; \bar{P}(I - \bar{C}\bar{P})^{-1}$$

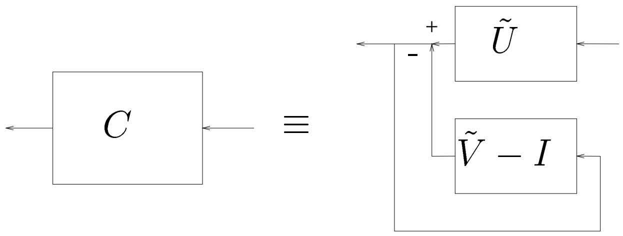


Figure 4.5: Coprime factor implementation of $C(s)$

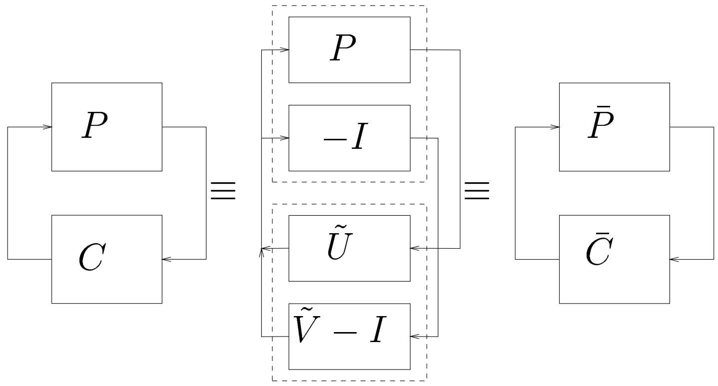


Figure 4.6: Equivalent feedback loop

$$
\begin{aligned}
&= \begin{bmatrix} N \\ -M \end{bmatrix} M^{-1}(I-\tilde{U}NM^{-1}+\tilde{V}-I)^{-1} \\
&= \begin{bmatrix} N \\ -M \end{bmatrix} (\tilde{V}M-\tilde{U}N)^{-1} = \begin{bmatrix} N \\ -M \end{bmatrix}
\end{aligned}
$$

Thus, $-\bar{P}(I-\bar{C}\bar{P})^{-1}$ is a right inverse of $\tilde{C}(s) = \begin{bmatrix} \tilde{U} & \tilde{V} \end{bmatrix}$; denoting it by $\tilde{C}^{\dagger}$, stability-oriented coprime factor model reduction may be formulated as reduction of $\bar{C} = \begin{bmatrix} \tilde{U} & \tilde{V}-I \end{bmatrix}$ making $\tilde{C}^{\dagger}(\bar{C}-\bar{C}_r)$ small, or equivalently reduction of $\tilde{C} = \begin{bmatrix} \tilde{U} & \tilde{V} \end{bmatrix}$ making $\tilde{C}^{\dagger}(\tilde{C}-\tilde{C}_r)$ small, i.e. relative error model reduction of $\tilde{C}$.

It goes without saying that the particular choice of coprime factorization has a bigger impact on the reduced order controller than the relative error approximation method employed to reduce the coprime factors.

Note that the approach above is very related to the one of [177].

4.6.2 IOWBT

In this subsection, a generalization of the output weighted IOWBT algorithm of Subsection 4.2.1 and the BST algorithm of Subsection 4.3.1 for fat systems is given. For square systems, there is no ambiguity with respect to the IOWBT weighting function : this is the inverse of the outer factor of $G(s)$. This changes for non-square systems.

Again, $W(s)W^T(-s) = G(s)G^T(-s)$ if $W(s) = \left[\begin{array}{c|c} A & \bar{B} \\ \hline C & D \end{array}\right]$ with

$$\bar{B}D^T + \bar{P}C^T = BD^T + PC^T \tag{4.17}$$

$$\text{and} \qquad A\bar{P} + \bar{P}A^T + \bar{B}\bar{B}^T = 0\,. \tag{4.18}$$

Defining $X \triangleq \bar{P} - P$, $D^\dagger \triangleq D^T(DD^T)^{-1}$, $C_i \triangleq -D^\dagger C$, $A_i \triangleq A - BD^\dagger C$, and $D^\perp$ such that $D^{\perp T}D^\perp = I - D^\dagger D = I - D^T {D^\dagger}^T = I - D^T(DD^T)^{-1}D$, it is readily verified that $\bar{B} \triangleq B(I - D^{\perp T}D^\perp) - XC^T(DD^T)^{-1}D$ satisfies (4.17).

For this definition of $\bar{B}$ to be compatible with (4.18), X should obey

$$\begin{aligned} 0 \;=\; & AX + XA^T - BB^T \\ & + (B(I - D^{\perp T}D^\perp) - XC^T(DD^T)^{-1}D)(B(I - D^{\perp T}D^\perp) - XC^T(DD^T)^{-1}D)^T\,, \end{aligned}$$

$$\text{i.e.} \qquad A_iX + XA_i^T + XC_i^TC_iX - BD^{\perp T}D^\perp B^T = 0\,. \tag{4.19}$$

Any X satisfying this equation yields a $W(s)$; this system has many right inverses, a particular one is

$$\begin{aligned} W^\dagger(s) \;&\triangleq\; \left[\begin{array}{c|c} A - \bar{B}D^\dagger C & \bar{B}D^\dagger \\ \hline -D^\dagger C & D^\dagger \end{array}\right] \\ &= \left[\begin{array}{c|c} A - BD^\dagger C + XC^T{D^\dagger}^T D^\dagger C & BD^\dagger - XC^T{D^\dagger}^T D^\dagger \\ \hline C_i & D^\dagger \end{array}\right] \\ &= \left[\begin{array}{c|c} A_i + XC_i^TC_i & BD^\dagger - XC^T{D^\dagger}^T D^\dagger \\ \hline C_i & D^\dagger \end{array}\right]. \end{aligned}$$

Hence one should again choose the solution X to (4.19) making $A_i + XC_i^TC_i$ stable, so that $W^\dagger(s)$ can be used as an output weight for Enns's FWBT.

From here, using the same procedure as in Subsection 4.2.1, Q in FWBT is found to obey the $n \times n$ Lyapunov equation

$$A_{wi}^TQ + QA_{wi} + C_iC_i^T = 0\,,$$

where $A_{wi} \triangleq A_i + XC_iC_i^T$ as before.

Thus, IOWBT generalizes rather straightforwardly to non-square systems, but as the Riccati equation (4.19) *does* have a constant term here, it looses some of its appeal, i.e. the computational advantage : no a priori Schur decomposition can be written down for the Hamiltonian.

4.6.3 BST

For non-square systems, the BST equations (4.9) generalize only for right (causal) invertible systems, i.e. DD^T should be invertible. This implies that the systems should be *fat*, i.e. $n \times m$ systems where $n \leq m$. Evidently, for the dual BST equations a tall system is required.

Recall that $D^\dagger \triangleq D^T(DD^T)^{-1}$, $(DD^T)^{-1} = {D^\dagger}^T D^\dagger$,
$D^{\perp T}D^\perp \triangleq I - D^\dagger D = I - D^T {D^\dagger}^T = I - D^T(DD^T)^{-1}D$, and $C_i \triangleq -D^\dagger C$.
Then a new version of Theorem 4.4 may be presented :

Theorem 4.7
Let $X \triangleq Q_w^{-1} - P$, *where* Q_w *solves* (4.9),

Then
$$A_iX + XA_i^T + XC_i^TC_iX - BD^{\perp T}D^\perp B^T = 0 \; .$$

Conversely, any $Q_w = (P+X)^{-1}$ *solves* (4.9).

Proof :

$$\begin{aligned}
0 &= A^TQ_w + Q_wA + (C^T - Q_wB_w){D^\dagger}^T D^\dagger (C - B_w^TQ_w) \\
&= A^TQ_w + Q_wA \\
&\qquad + (C - CPQ_w - DB^TQ_w)^T {D^\dagger}^T D^\dagger (C - CPQ_w - DB^TQ_w) \\
&= A^TQ_w + Q_wA \\
&\quad + (Q_wPC_i^T - C_i^T - Q_wB(I - D^{\perp T}D^\perp))(C_iPQ_w - C_i - (I - D^{\perp T}D^\perp)B^TQ_w) \\
&= A^TQ_w + Q_wA + (I - Q_wP)C_i^TC_i(I - PQ_w) + Q_wBC_i(I - PQ_w) \\
&\qquad + (I - Q_wP)C_i^TB^TQ_w + Q_wB(I - D^{\perp T}D^\perp)B^TQ_w \\
&= A_i^TQ_w + Q_wA_i + (I - Q_wP)C_i^TC_i(I - PQ_w) \\
&\qquad - Q_w(BC_iP + PC_i^TB^T - BB^T)Q_w - Q_wBD^{\perp T}D^\perp B^TQ_w \\
&= A_i^TQ_w + Q_wA_i + (I - Q_wP)C_i^TC_i(I - PQ_w) \\
&\qquad - Q_w(A_iP + PA_i^T)Q_w - Q_wBD^{\perp T}D^\perp B^TQ_w \\
&= (I - Q_wP)A_i^TQ_w + Q_wA_i(I - PQ_w) + (I - Q_wP)C_i^TC_i(I - PQ_w) \\
&\qquad - Q_wBD^{\perp T}D^\perp B^TQ_w
\end{aligned}$$

$$\begin{aligned} = \quad & (Q_w^{-1} - P)A_i^T + A_i(I - PQ_w) \\ & + (Q_w^{-1} - P)C_i^T C_i(Q_w^{-1} - P) - BD^{\perp T} D^{\perp} B^T \qquad \square \end{aligned}$$

Clearly the computational advantage is lost, as the ARE has a constant term. A result very similar to Theorem 4.7 has been obtained by Gu [124], however the ARE obtained there has X^{-1} as solution, and Gu's approach is no longer equivalent to the one presented here for square systems.

4.7 Examples

In a first academic example,

$$\text{the system} \qquad G(s) \triangleq \frac{\prod_{i=1}^{5}(s - z_i)}{\prod_{i=1}^{5}(s - p_i)}, \qquad \text{where}$$

$z_{1,2} = -0.5 \pm 5j$, $z_{3,4} = 1 \pm 4j$, $z_5 = 100$, and $p_{1,2} = -4 \pm 12j$, $p_{3,4} = -1 \pm 4j$, $p_5 = -1$, is reduced to order $r = 3$. $G(s)$ is non-minimum phase with $p = 3$. The IOWBT and BST reduced models are $G_{io}(s)$ and $G_{bst}(s)$, where

$$G_{io}(s) = \frac{(z - 100.1)(z^2 - 1.277z + 18.39)}{(z + 0.495)(z^2 + 8.86z + 173.0)},$$

$$G_{bst}(s) = \frac{(z - 100)(z^2 - 2z + 17)}{(z + 0.499)(z^2 + 9.00z + 157.6)}.$$

The IOWBT relative error is $\|G(s)^{-1}(G(s) - G_{io}(s))\|_\infty = 0.98$, whereas the error for BST is $\|G(s)^{-1}(G(s) - G_{bst}(s))\|_\infty = 1.86$. Note that the zeroes of $G_{bst}(s)$ are the non-minimum phase zeroes of $G(s)$, which is apparently too restrictive. The magnitude and phase Bode plots for the original system (full line), BST (dash-dotted line) and IOWBT (dashed line) are given in Fig. 4.7(a) and (b). The relative error magnitude vs. frequency for BST (dash-dotted line) and IOWBT (dashed line) are given in Fig. 4.8.

A second example, taken from the demo `remdemo` (by Chiang and Safonov) in Matlab's Robust Control Toolbox 2.0b, illustrates that IOWBT performs better than BST whenever $r < p$, not just for maximum-phase systems. This is a 16-th order, continuous-time MIMO system with 7 non-minimum phase zeroes, which is to be reduced to order 6.

The IOWBT error is $\|G(s)^{-1}(G(s) - G_r(s))\|_\infty = 1.949$, whereas the error for BST is $\|G(s)^{-1}(G(s) - G_{BST}(s))\|_\infty = 232.5$.
The singular values for the original system (full line), BST (dash-dotted line) and IOWBT (dashed line) are plotted vs. frequency in Fig. 4.9(a). The relative

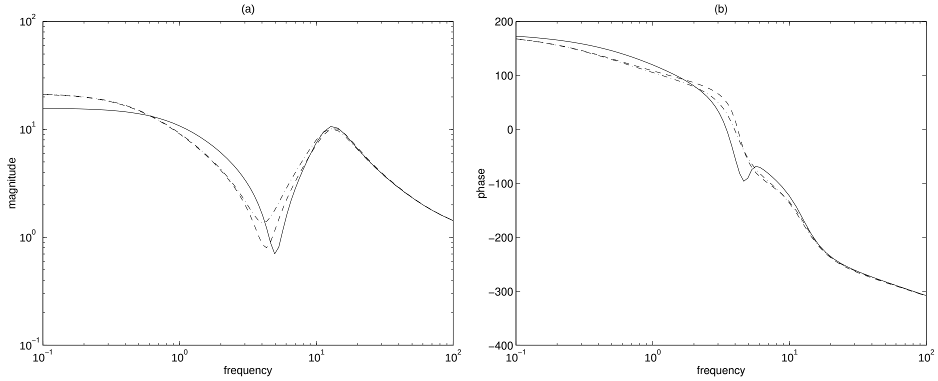


Figure 4.7: First example : $r = 3$;
full line : original system, dashed line : IOWBT, dash-dotted line : BST .
(a) magnitude Bode plots (b) phase Bode plots

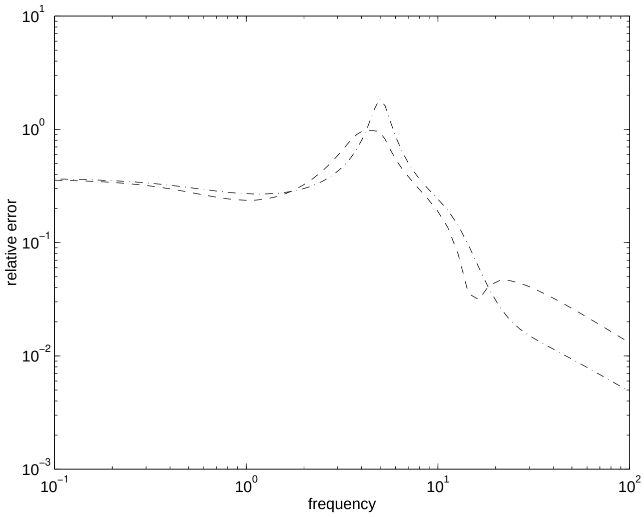


Figure 4.8: First example : relative error magnitude Bode plots.
Full line : original system, dashed line : IOWBT, dash-dotted line : BST .

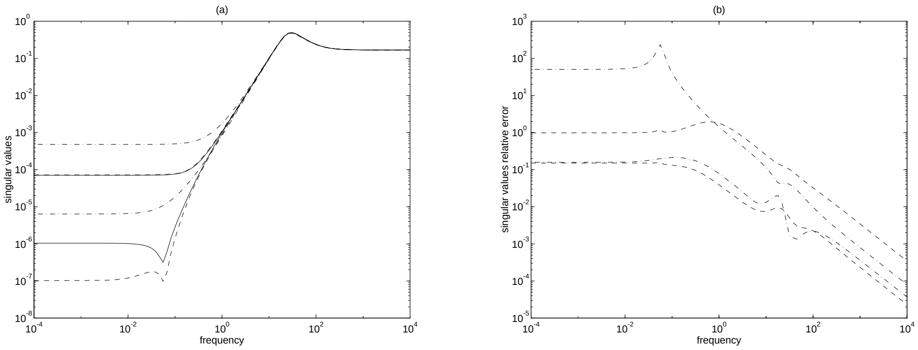


Figure 4.9: Second example :
Full line : original system, dashed line : IOWBT, dash-dotted line : BST.
(a) singular values σ_i (b) relative error singular values $\sigma_{BST,i}$

error singular values vs. frequency is for BST (dash-dotted line) and IOWBT (dashed line) are given in Fig. 4.9(b).

The last example illustrates the possible effectiveness of relative error controller reduction, even though it is but a simple-minded method, for which even no knowledge of the controlled plant is needed. The famous CD-player mechanism, presented in [89] and the references found therein, is a tracking control system, aimed at letting the focus point of a laser beam follow the tracks on a CD. The controller considered in this example has one input, the radial tracking error signal, and one output, the actuation of the radial arm on which the laser-sensor is mounted; lens focusing is not considered, as in [302]. The data for this system, as well as the full-order controller, are taken from [303]. The plant is the model G_{32} of order 32 adopted there, and the controller K_{37} a 37th order controller, H_∞-optimal for the standard plant $P = \begin{bmatrix} W_2 & 0 \\ 0 & W_3 \end{bmatrix} \begin{bmatrix} 0 & I \\ I & G_{32} \end{bmatrix}$, where W_2 and W_3 are weighting systems (of order 2 and 3, resp.) chosen in such a way that minimizing $\|L_f(P,K)\|_\infty$ yields a satisfactory controller; not surprisingly, this controller achieves good tracking.

K_{37} is strictly proper, hence the trick of Subsection 4.2.1 is applied to make it causal invertible; a D-matrix $D = 0.001$ is added here, afterwards the reduced-order controllers are replaced by their strictly proper parts. Both IOWBT and BST and ordinary Balanced Truncation (BT) are applied to K_{37}, with $r = 4$, yielding K_{io}, K_{bst}, and K_{bt}; the controller $\hat{K}_4$ proposed in [302], the result of

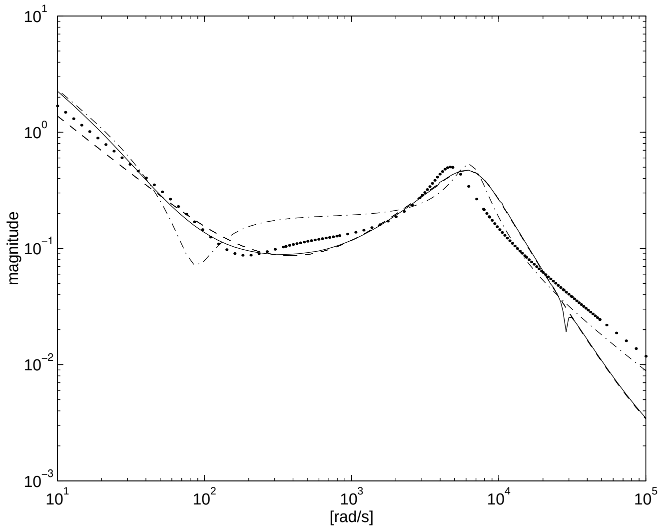


Figure 4.10: CD player : controller magnitude Bode plot $|K(j\omega)|$.
Full line : full order controller K_{37}, dashed line : coinciding plots of K_{io} and K_{bst}, dash-dotted line : K_{bt}, dotted line : $\hat{K}_4$.

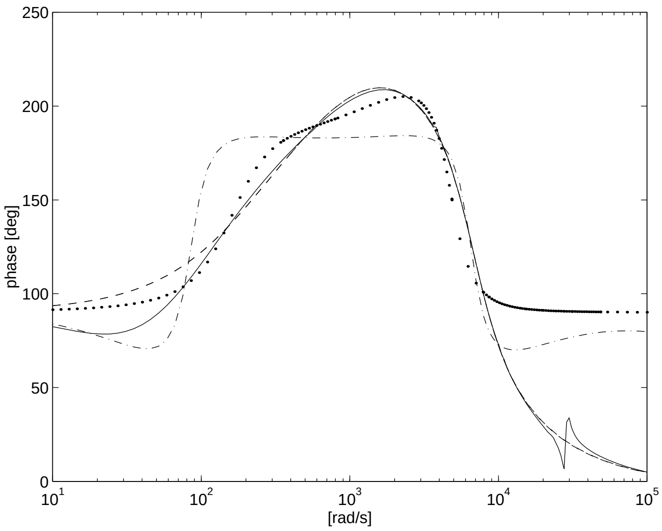


Figure 4.11: CD player :controller phase Bode plot $\angle K(j\omega)$.
Full line : full order controller K_{37}, dashed line : coinciding plots of K_{io} and K_{bst}, dash-dotted line : K_{bt}, dotted line : $\hat{K}_4$.

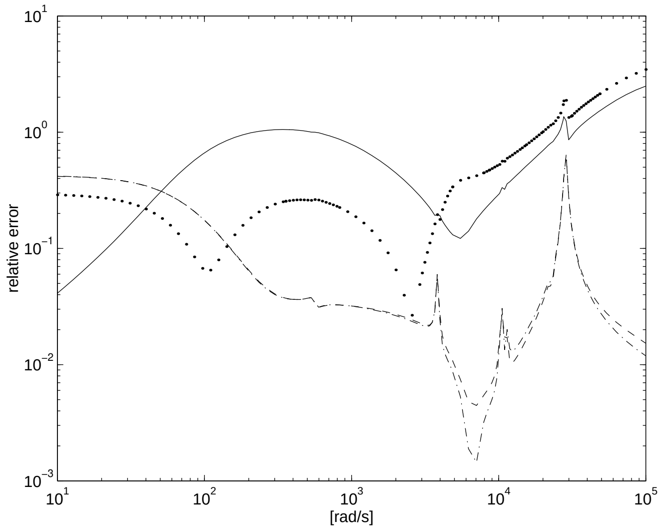


Figure 4.12: CD player : relative error $\bar{\sigma}(K^{-1}(K - K_r))$.
Full line : BT system, dashed line : IOWBT, dash-dotted line : BST, dotted line : improved CLBT.

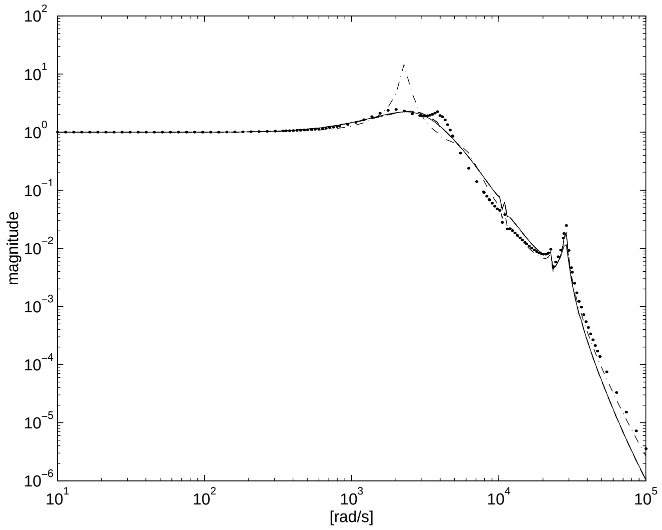


Figure 4.13: CD player : complementary sensitivity $\bar{\sigma}(I + G_{32}K^{-1}G_{32}K)$.
Full line : with full order controller K_{37}, dashed line : coinciding plots of IOWBT and BST, dash-dotted line : BT, dotted line : improved CLBT.

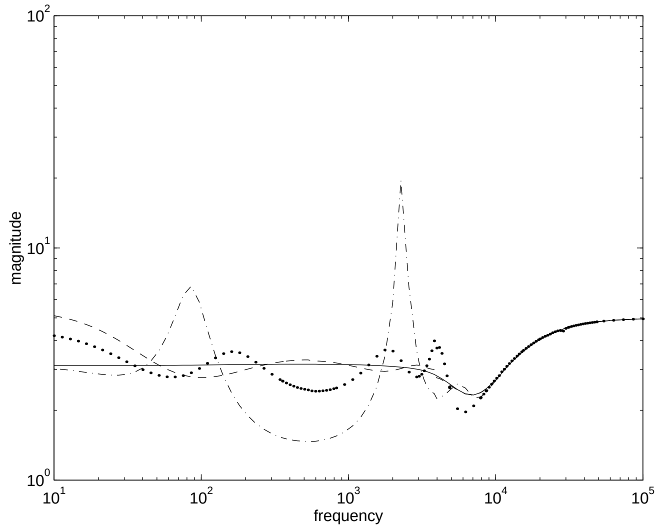


Figure 4.14: CD player : standard plant closed-performance $\bar{\sigma}(L_f(P, K))$.
Full line : with full order controller K_{37}, dashed line : coinciding plots of IOWBT and BST, dash-dotted line : BT, dotted line : improved CLBT.

CLBT using P improved by an interactive iterative procedure, is also included in the comparison, to provide an 'optimal reduced-order' reference. In Fig. 4.10 and Fig. 4.11, the Bode plots of the different controllers are given. Remark that even though K_{37} is non-minimum phase ($p = 1$), K_{io} and K_{bst} coincide, at least visually.

The relative error Bode plots $\bar{\sigma}(K^{-1}(K - K_r))$ (Fig. 4.12) exhibit only marginal difference; there is of course no reason why the relative errors of $\hat{K}_4$ and K_{bt} should be small, they are included in Fig. 4.12 only for the amusement of the reader. The Bode plot of the tracking transfer function or complementary sensitivity $\bar{\sigma}((I + G_{32}K)^{-1}G_{32}K)$ is displayed in Fig. 4.13; the full order transfer function almost coincides with the plots for $K = K_{io}$ and $K = K_{bst}$. The largest singular value plot of the closed-loop standard plant reflecting performance is given in Fig. 4.14.

Relative error model reduction is not being put forward as being superior to the more sophisticated method yielding $\hat{K}_4$, yet note that it is only slightly suboptimal; but the point made here is that the hurried control designer, looking for a quick, generic and simple-minded reduction method, is well advised to use a relative error model reduction such as IOWBT or BST instead of unweighted

Balanced Truncation.

4.8 Conclusions

Enns's Frequency Weighted Balanced Truncation is readily applicable to relative error model reduction, but it has been shown here that the structure of the gramians leads to a simplification in this case, making the method more attractive.

The method generally results in a modeling error of approximately the same magnitude as Balanced Stochastic Truncation; unfortunately, an error bound as for BST could not be derived. However, the method may be useful even if the reduced order is smaller than the number of non-minimum phase zeroes of the system to be reduced, which is not true for BST.

The operation count for IOWBT compares favorably with the one for the classical BST algorithm; however, it was shown how a BST reduced model may be found solving only Lyapunov equations and a Riccati equation without a constant term, which leads to a similar operation count.

A generalization to non-square systems was touched upon; although motivations for relative error model reduction can be found, this type of approximation is more ambiguous in this case, and some of the advantages of the algorithms proposed in this chapter are lost.

Chapter 5

H_2-optimal model reduction

This chapter deals with unweighted H_2-optimal model reduction. By way of introduction, the first section sums up some relevant properties (Subsections 5.1.2 and 5.1.3), which are used by some of the approaches to this problem found in the literature (Subsection 5.1.4). Section 5.2 introduces the STLS algorithm and shows how it can be used for H_2 approximation of SISO systems. The problems connected with a MIMO generalization is briefly touched upon in Section 5.3, where it is also shown that such a generalization *is* possible for SIMO or MISO systems. In Section 5.4, the iterative STLS algorithm for H_2 approximation is translated to a form in which the iteration may be performed at a much lower cost. The algorithm is illustrated by an example in Section 5.5.

5.1 Introduction

5.1.1 Motivation

Optimal model reduction in the H_2 norm was one of the first model reduction techniques considered and has continued to raise attention in the last thirty years. The fact that this interest peeked in the period that LQG was the reigning control paradigm is perhaps not surprising. Although there is of course no guarantee that H_2-optimal model reduction of a controller leads to an optimal closed-loop H_2 norm over all fixed-order controllers, frequency weighted H_2-optimal controller reduction with performance-oriented weights as in Section 2.4 is likely to yield a nearly optimal closed-loop H_2 norm $\|H_{zw}\|_2$, provided the reduced order is not chosen too low. Note that frequency weighted H_2-optimal model reduction is not treated in this chapter, but is developed separately in Chapter 6.

Following the recognition that H_2-optimal control designs can be arbitrarily unrobust [82] and the pioneer work of Zames [316], a shift of focus occurred towards the H_∞ norm. Using the small gain theorem, one can guarantee robust stability under H_∞ norm-bounded perturbations [83, 87]. This type of results was one of the main motivations for the enormous research effort in H_∞ control that we have known during the last decade, and also the starting point for μ-synthesis. Correspondingly, in the light of these robust stability results, more emphasis was put on methods which can provide H_∞ error bounds for the modeling error; balanced truncation (BT) [195] and optimal Hankel norm approximation [107] are examples of such methods.

Nobody will doubt that it is more pleasant to have an H_∞ error bound available than not to have one, yet a couple of remarks on the usefulness of such bounds can be made. In the first place, it is very doubtful that these bounds are often used in practice by control designers. Indeed, most often closed-loop stability will be checked for each candidate reduced-order controller by computing the closed-loop poles, as well as many other properties, making an H_∞ error bound which guarantees stability quite useless. Furthermore, for weighted model reduction methods, which are much more relevant with respect to closed-loop stability and performance than the unweighted ones given above, such error bounds are often lacking. However, the advantage by using such weighted methods is most often so considerable that they are to be preferred in spite of this shortcoming.

As pointed out before, a lot of stress is laid on the pre-computability of error bounds : if it is not considerably cheaper to compute such a bound than the actual error, the bound is evidently quite useless. But then another remark is in order : the computing power in terms of operations per second and megabytes of memory has grown exponentially for decades, and is likely to continue to do so for a while. Because of this increased computing power, the exact H_∞ error may be computed much faster now than a decade ago; moreover, efficient algorithms for computing the H_∞ norm were not available until late in the eighties [55]. Hence the range of system orders for which the error bounds are useful is continually shifting upwards. On the other hand, the order of the systems to which dense model reduction methods such as balanced truncation may be applied, is bounded above, not only by computing time considerations, but especially because numerical reliability becomes a problem for systems of very large size; the only machine parameter this problem depends on is the machine precision ϵ, which of course has not seen a similar exponential decrease over time. For system orders in the range of hundred or more one may expect such problems, depending on the actual data; an H_∞ norm of a 100th order system computes in less than a minute on a workstation today. Thus these bounds are gradually losing relevance.

Moreover, in defense of H_2 optimization, it should be recognized that as a performance measure, the H_2 norm is more natural than the H_∞ norm. The fact that the latter is a worst-case energy gain (see Subsection 2.2.7) causes performance criteria constructed using this norm to be overly conservative. Of course, a worst-case performance measure may have its practical relevance too; but then the H_1 norm, using which the instantaneous output amplitude may be bounded, might be a better candidate. Of course, we do not mean to imply that the H_∞ norm is quite useless; its importance for robust control design, in particular when it is combined with μ-synthesis, is firmly established.

5.1.2 Facts about H_2 model reduction

The H_2-optimal reduced order model G_r^* minimizes the H_2 norm of the modeling error, i.e.

$$G_r^* = \arg \min_{G_r \in \mathcal{S}_r} \|G - G_r\|_2 \ .$$

This minimizer cannot be non-minimal, i.e.

Theorem 5.1

$$\text{Let } G_r^* = \arg \min_{G_r \in \mathcal{S}_r} \qquad \|G - G_r\|_2 \ ,$$

Then *the McMillan degree of G_r^* is r.*

Proof : see [223, p.182] □

In other words, $\min_{G_r(s)\in\mathcal{S}_r} \|G(s) - G_r(s)\|_2$ is strictly decreasing as r increases. This theorem is valid even for local minimizers.

One important property was mentioned already in Chapter 2 : the H_2-optimal reduced-order approximant can be found by projection of dynamics. This follows easily from the necessary conditions for H_2-optimality derived in Subsection 5.1.3.

The following necessary conditions have been used as the base of some SISO algorithms :

Theorem 5.2

Let $G(s) \in \mathcal{S}$ *be a CT SISO system,*

$$\text{Let } G_r^*(s) = \arg \min_{G_r(s) \in \mathcal{S}_r} \|G(s) - G_r(s)\|_2 \ ,$$

$$\text{Let } p \text{ be a pole of } G_r(s),$$

$$\text{Then } \frac{d^m}{ds^m}(G(s)-G_r(s)) \Big|_{s=-p} = 0 \ , \qquad m=0,1 \ .$$

Proof : may be found in [190].

Similarly, for DT systems one has

Theorem 5.3

$$\text{Let } G(z) \in \mathcal{S}^d \text{ be a DT SISO system,}$$

$$\text{Let } G_r^*(z) = \arg \min_{G_r(z)\in\mathcal{S}_r^d} \|G(z)-G_r(z)\|_2 \ ,$$

$$\text{Let } p \text{ be a pole of } G_r(z),$$

$$\text{Then } \frac{d^m}{dz^m}(G(z)-G_r(z)) \Big|_{z=p^{-1}} = 0 \ , \qquad m=0,1 \ .$$

(See also Subsection 5.4.4.)

More results on H_2-optimal reduction can be found in [223, 37, 33, 32, 31, 35, 36].

5.1.3 Necessary conditions

The H_2 norm squared is differentiable everywhere; hence necessary conditions for optimality may easily be found by deriving the gradient of the square of the H_2 norm with respect to the state-space parameters. This is quite standard; this subsection is only included to set it off against Subsection 7.3.1.

Let $G(s) \triangleq [A,B,C,D]$ be a stable continuous-time system, P and Q be its controllability and observability gramians. Recall that only if $D=0$ the H_2-norm of $G(s)$ exists, and that $J \triangleq \|G(s)\|_2^2$ may be computed from $J = \text{Tr } PC^TC = \text{Tr } QBB^T$; using this, the gradients with respect to B and C are directly found to be

$$\frac{\partial}{\partial B_{ij}} J = 2[QB]_{(i,j)}, \qquad \frac{\partial}{\partial C_{ij}} J = 2[CP]_{(i,j)}.$$

The gradient with respect to A is derived as follows :

$$\begin{aligned} \frac{\partial}{\partial A_{ij}} J &= \text{Tr } \frac{\partial P}{\partial A_{ij}} C^T C \\ &= -\text{Tr } \frac{\partial P}{\partial A_{ij}} (A^T Q + QA) \\ &= -\text{Tr } Q(A\frac{\partial P}{\partial A_{ij}} + \frac{\partial P}{\partial A_{ij}} A^T) \ . \end{aligned}$$

Now from $AP + PA^T + BB^T = 0$ one finds

$$A\frac{\partial P}{\partial A_{ij}} + \frac{\partial P}{\partial A_{ij}}A^T + e_i e_j^T P + P e_j e_i^T = 0 \ ,$$

where e_i is the vector with 1 as the i-th element and zeros elsewhere,

$$\begin{aligned} \text{hence} \qquad \frac{\partial}{\partial A_{ij}} J &= \mathrm{Tr}\ Q(e_i e_j^T P + P e_j e_i^T) \\ &= 2[QP]_{(i,j)} \ . \end{aligned}$$

These expressions are directly useful when the H_2-norm of $\tilde{G}(s) \triangleq G(s) - G_r(s)$ is being minimized over $G_r(s)$; indeed if

$$G_r(s) \triangleq \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array}\right], \ \tilde{G}(s) = \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right] \triangleq \left[\begin{array}{cc|c} A & 0 & B \\ 0 & \hat{A} & \hat{B} \\ \hline C & -\hat{C} & D - \hat{D} \end{array}\right],$$

and $\tilde{J} \triangleq \|\tilde{G}(s)\|_2^2$, then the gradient of $\tilde{J}$ with respect to $G_r(s)$ is found by substituting A, B, and C by $\tilde{A}$, $\tilde{B}$, and $\tilde{C}$ in the expressions above, i.e.

$$\tilde{A}\tilde{P} + \tilde{P}\tilde{A}^T + \tilde{B}\tilde{B}^T = 0 \qquad \tilde{A}^T\tilde{Q} + \tilde{Q}\tilde{A} + \tilde{C}^T\tilde{C} = 0$$

$$\begin{aligned} \frac{\partial}{\partial \hat{A}_{ij}}\tilde{J} &= 2\left[\begin{bmatrix} 0 & I \end{bmatrix}\tilde{Q}\tilde{P}\begin{bmatrix} 0 \\ I \end{bmatrix}\right]_{(i,j)} \\ \frac{\partial}{\partial \hat{B}_{ij}}\tilde{J} &= 2\left[\begin{bmatrix} 0 & I \end{bmatrix}\tilde{Q}\tilde{B}\right]_{(i,j)} \\ \frac{\partial}{\partial \hat{C}_{ij}}\tilde{J} &= 2\left[\tilde{C}\tilde{P}\begin{bmatrix} 0 \\ I \end{bmatrix}\right]_{(i,j)} \end{aligned}$$

Necessary conditions can be derived from these gradients for (possibly frequency weighted) optimal H_2 model reduction, H_2 optimal static output feedback design, or H_2 optimal fixed order control.

Naturally, the gradients may also be used for gradient-based optimization such as quasi-Newton methods etc., see e.g. [296, 18, 186, 264]. It should be stressed that none of these algorithms give any guarantee that a global optimum is reached; in essence, they are all descent methods, where the correspondence between the initial point and the local optimum eventually reached is very clear.

Partitioning the gramians in the same way as $\tilde{A}$, i.e. $\tilde{P} = \begin{bmatrix} P & \bar{P} \\ \bar{P}^T & \hat{P} \end{bmatrix}$, and $\tilde{Q} = \begin{bmatrix} Q & \bar{Q} \\ \bar{Q}^T & \hat{Q} \end{bmatrix}$, one finds the following necessary conditions for optimality :

$$\begin{aligned} \bar{Q}^T \bar{P} &= -\hat{Q}\hat{P} \\ \bar{Q}^T B &= -\hat{Q}\hat{B} \\ C\bar{P} &= \hat{C}\hat{P} \, . \end{aligned} \tag{5.1}$$

Keeping in mind that $G_r(s)$ has McMillan degree r for each local minimum, and that therefore $\hat{P}$ and $\hat{Q}$ are invertible, one may define $L \triangleq -\hat{Q}^{-1}\bar{Q}^T$ and $R \triangleq \bar{P}\hat{P}^{-1}$. Then it is seen that $\hat{C} = CR$ and $\hat{B} = LB$; moreover $LR = I$, hence if it can be shown that $\hat{A} = LAR$, it can be proved from these relations that the H_2 optimal approximant is obtained by projection of dynamics.
Indeed, this latter equality follows from multiplying $A\bar{P} + \bar{P}\hat{A}^T + B\hat{B}^T = 0$ by L from the left : together with $\bar{P} = R\hat{P}$, this yields $LAR\hat{P} + LR\hat{P}\hat{A}^T + \hat{B}\hat{B}^T = 0$; as $\hat{A}\hat{P} + \hat{P}\hat{A}^T + \hat{B}\hat{B}^T = 0$ and $LR = I$ this means that $\hat{A}\hat{P} = LAR\hat{P}$, or $\hat{A} = LAR$, since $\hat{P}$ is invertible.

For a stable discrete-time system $G(z) \triangleq [A, B, C, D]$ with controllability and observability gramians P and Q, only the gradient with respect to A_{ij} is different from the continuous-time gradient :

$$\begin{aligned} \frac{\partial}{\partial A_{ij}} J &= \text{Tr } \left(\frac{\partial P}{\partial A_{ij}} (Q - A^T Q A) \right) \\ &= \text{Tr } \left((\frac{\partial P}{\partial A_{ij}} - A \frac{\partial P}{\partial A_{ij}} A^T) Q \right) \\ &= \text{Tr } \left((e_i e_j^T P A^T + A P e_j e_i^T) Q \right) \\ &= 2[QAP]_{(i,j)} \; . \end{aligned}$$

Hence for discrete-time H_2-optimal model reduction, condition (5.1) should be replaced by

$$\bar{Q}^T A \bar{P} = -\hat{Q}\hat{A}\hat{P} \; .$$

With the same definitions of L and R, this yields $\hat{A} = LAR$; however, $LR = I$ needs some further proof here. This follows similarly from multiplication of $A\bar{P}\hat{A}^T - \bar{P} + B\hat{B}^T = 0$ by L from the left, which yields $LAR\hat{P}\hat{A}^T - LR\hat{P} + \hat{B}\hat{B}^T = 0$; this together with $\hat{A}\hat{P}\hat{A}^T - \hat{P} + \hat{B}\hat{B}^T = 0$ means that $LR = I$, and again the projection of dynamics structure is made apparent.

5.1.4 Existing algorithms

In view of the fact that the H_2-optimal model reduction problem can be considered as a special case of the H_2-optimal static output feedback problem or the H_2-optimal fixed order controller problem, algorithms for the latter two problems are included in this concise and certainly not exhaustive survey.

Meier [190] is one of the first authors to propose a method based on the solution of the necessary conditions formulated as interpolation conditions as in Theorem 5.2.

Wilson [294] derives the necessary conditions as presented in Subsection 5.1.3, and puts forward algorithms based on this [294, 295, 296, 297]. In [295], $\hat{B}$ and $\hat{C}$ are first solved for fixed $\hat{A}$ in an inner loop, and $\hat{A}$ is only updated using gradient information after convergence of this iteration; this idea is further developed in [193]. Aplevich [27, 28], Galiana [99], and other authors [135, 182, 225, 252, 34] also consider steepest descent or other general optimization methods using the gradients computed as above.

Noldus gives a frequency weighted gradient-type algorithm, with an additional steady-state constraint [207].

Hyland and Bernstein focus on the projection of dynamics structure, already noted by Wilson, and formulate the optimal projection equations. These may be solved in an iterative way [139, 141, 140, 142, 302, 304] or using *homotopy* or *continuation* methods [118, 224, 191, 196, 104]. Halevi gives frequency weighted extensions [125, 126].

In a set of papers [184, 185, 186, 263, 264, 265], Mäkilä and Toivonen study a number of iterative methods for optimal static output feedback, notably a method of Anderson and Moore [18], the Levine-Athans method [172], Newton-Raphson and modifications thereof. A similar method is found in [194].

Hanzon and Maciejowski [131] investigate the possibility of using Groebner bases to solve the necessary conditions formulated as a set of polynomial equations.

Lepschy et al. propose a non-gradient algorithm [170, 171] based on a formulation of the necessary conditions as a polynomial identity corresponding to a set of nonlinear equations; these are solved iteratively, where each iteration a linear equation is solved. In this, the approach bears some affinity to the algorithm presented in the next sections; but reportedly, Lepschy's algorithm needs a close initial guess whenever $r < n - 1$ [170].

5.2 Structured Total Least Squares

In the remaining sections of this chapter, H_2-optimal model reduction is tackled in a completely different framework, i.e. Structured Total Least Squares (STLS). This section deals with the SISO case; the MIMO case, and the problems connected with it, are briefly discussed in Section 5.3. Section 5.4 introduces a translation of the necessary conditions and algorithms to be presented in this section to the z-domain; what is meant by this is explained there. This translation enables us to execute these algorithms much more efficiently.

The STLS problem, studied in [74], is formulated as follows :

$$\min_{b\in\mathbb{R}^n,\ y\in\mathbb{R}^q} f(a,b) \tag{5.2}$$

$$\text{s.t. } B(b)y = 0 \quad y^Ty = 1$$

where $a \in \mathbb{R}^n$ is a data vector, $f(a,b)$ is a quadratic function of a and b, and $B(b)$ is a $p \times q$ matrix depending affinely on b, i.e.

$$B(b) = B_0 + \sum_{i=1}^{n} b_i B_i \ , \qquad B_i \in \mathbb{R}^{p\times q} \ .$$

Problems of this type can be translated into a 'solution-dependent' generalized eigenvalue problem of the form

$$\begin{aligned} Ay &= D_y l \\ A^T l &= D_l y \\ b_k &= a_k - l^T B_k y \end{aligned}$$

where $A = \sum_{i=1}^{n} a_i B_i$ and D_y and D_l are symmetric nonnegative definite matrices whose elements are quadratic expressions of y and l, respectively.

In Subsection 5.2.1, it is explained how the H_2 approximation problem is recast as an STLS problem; some algorithms based on the new formulation are presented in Subsection 5.2.2. Subsection 5.2.3 shows how these algorithms may be sped up.

5.2.1 STLS for H_2 approximation

For SISO discrete-time systems, the H_2 approximation problem may also be formulated in the structured total least squares framework [73, 76]. In this approach, a parameterization of the approximant is chosen that does not inherently restrict its order, and the fact that the optimization is done over the set of approximants of order at most r is formulated as a rank constraint on the infinite Hankel matrix containing the Markov parameters. In this subsection, necessary conditions for a stationary point of this constrained optimization problem, derived using Lagrange multipliers, are reformulated in such a way that its structure as a solution-dependent generalized eigenvalue decomposition is made clear. This leads to an algorithm which is quite different from the gradient-based approaches mentioned above.

Consider a discrete-time, SISO, strictly proper, stable system with transfer function $h(z) = \sum_{i=1}^{\infty} h_i z^{-i}$. Recall that the H_2-optimal model reduction problem is stated as

$$\begin{aligned} \hat{h}(z) \triangleq \sum_{i=1}^{\infty} \hat{h}_i z^{-i} &= \arg\min_{\hat{h}_i} \frac{1}{2} \sum_{i=1}^{\infty} (h_i - \hat{h}_i)^2 \\ \text{s.t.} \quad &\text{McMillan } \deg \hat{h}(z) \leq r \ . \end{aligned} \tag{5.3}$$

Remark that this formulation addresses a broader problem than just model reduction : if inexact data are used for the Markov parameters h_i, this framework can be used to find the reduced order system that best matches the noisy impulse response data in a least squares sense, i.e. the *noisy realization* problem. This is essentially an identification problem from an impulse response measurement. Note that the STLS framework can also be used for more general identification, as well as for many other problems; see [73, 77].

If $\hat{h}(z)$ has McMillan degree at most r, then it is well known that the infinite-sized Hankel matrix

$$\hat{\mathcal{H}} \triangleq \begin{bmatrix} \hat{h}_1 & \hat{h}_2 & \hat{h}_3 & \cdots \\ \hat{h}_2 & \hat{h}_3 & \hat{h}_4 & \cdots \\ \hat{h}_3 & \hat{h}_4 & \hat{h}_5 & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix}$$

has rank $\leq r$; indeed, if $\hat{h}(z) \triangleq \left[\hat{A}, \hat{B}, \hat{C}, \hat{D}\right]$ is a realization of order r, then $\hat{\mathcal{H}} = \mathcal{O}\mathcal{C}$ where $\mathcal{O}$ is the (infinite-sized) observability matrix and $\mathcal{C}$ is the controllability matrix, i.e.

$$\mathcal{O} = \begin{bmatrix} \hat{C} \\ \hat{C}\hat{A} \\ \hat{C}\hat{A}^2 \\ \vdots \end{bmatrix} \qquad \mathcal{C} = \begin{bmatrix} \hat{B} & \hat{A}\hat{B} & \hat{A}^2\hat{B} & \cdots \end{bmatrix};$$

as $\mathcal{O}$ and $\mathcal{C}$ both have rank at most r, so has $\hat{\mathcal{H}}$. For the infinite Hankel matrix $\hat{\mathcal{H}}$ to have rank $\leq r$, it is sufficient that the matrix $\hat{H}$ made up of the first $r+1$

columns of $\hat{\mathcal{H}}$ is rank-deficient.
This can be expressed by stating that $\hat{H}$ should have a nonzero null vector $y = \begin{bmatrix} y_0 & y_1 & \cdots & y_r \end{bmatrix}^T$; without loss of generality, it may be required that $\|y\| = 1$. The optimization problem (5.3) may therefore be rephrased as

$$\hat{h}(z) = \arg\min_{\hat{h}_i} \frac{1}{2} \sum_{i=1}^{\infty} (h_i - \hat{h}_i)^2 \tag{5.4}$$

$$\text{subject to } \ \exists y \in \mathbb{R}^{(r+1)\times 1} \mid \hat{H}y = 0 \ \text{ and } \ y^T y = 1.$$

This is clearly an STLS problem, where $\hat{h}$ takes the role of b in (5.2), $\hat{H}$ stands for $B(b)$, and h for a. Next, we show how this problem may be converted into a 'solution-dependent' generalized singular value problem, as claimed above.

Using a Lagrange multiplier λ for the constraint $y^T y = 1$, and a vector of Lagrange multipliers l for the constraints $\hat{H}y = 0$, this is transformed into an unconstrained problem

$$\hat{h}(z) = \arg\min_{\hat{h}_i, y, l, \lambda} \frac{1}{2} \sum_{i=1}^{\infty} (h_i - \hat{h}_i)^2 + l^T \hat{H} y + \lambda (y^T y - 1) \ . \tag{5.5}$$

The 'vector' of Lagrange multipliers l has been treated rather informally here, however : the matrix $\hat{H}$ has an infinite number of rows, hence l is also an infinitely long vector. To be able to treat the problem more rigorously, we shall approximate the minimization problem by another one, where only a finite number $p + r$ of Markov parameters is taken into account :

$$\hat{h} = \arg\min_{\hat{h}_i} \frac{1}{2} \sum_{i=1}^{p+r} (h_i - \hat{h}_i)^2 \tag{5.6}$$

$$\text{s.t. } \ \exists y \in \mathbb{R}^{(r+1)\times 1} \text{ s.t. } \hat{H}_p y = 0 \text{ and } y^T y = 1,$$

$$\text{where } \hat{H}_p \triangleq \begin{bmatrix} \hat{h}_1 & \hat{h}_2 & \hat{h}_3 & \cdots & \hat{h}_{r+1} \\ \hat{h}_2 & \hat{h}_3 & \hat{h}_4 & \cdots & \hat{h}_{r+2} \\ \hat{h}_3 & \hat{h}_4 & \hat{h}_5 & & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ \hat{h}_p & \hat{h}_{p+1} & \hat{h}_{p+2} & \cdots & \hat{h}_{p+r} \end{bmatrix} .$$

To avoid overburdening the notation, we use the same symbol for the Markov parameters $\hat{h}_i$ here. Remark that this is no longer a model reduction problem, but for any stable $h(z)$ one may choose p large enough s.t. $\hat{h}_p(z) \triangleq \sum_{i=1}^{p+r} \hat{h}_i z^{-i}$

approximates $\hat{h}(z)$ to any desired degree of accuracy.

Define $\hat{h} \triangleq \left[\begin{array}{cccc} \hat{h}_1 & \hat{h}_2 & \cdots & \hat{h}_{p+r} \end{array} \right]^T$ and $h \triangleq \left[\begin{array}{cccc} h_1 & h_2 & \cdots & h_{p+r} \end{array} \right]^T$. The unconstrained formulation found by introducing Lagrange multipliers $l \in \mathbb{R}^{p\times 1}$ runs :

$$\hat{h} = \arg \min_{\hat{h}_i, y, l, \lambda} \frac{1}{2}\|h - \hat{h}\|^2 + l^T \hat{H}_p y + \lambda(y^T y - 1) \ . \tag{5.7}$$

Differentiation of $\frac{1}{2}\|h-\hat{h}\|^2 = \frac{1}{2}\sum_{i=1}^{p+r}(h_i - \hat{h}_i)^2$ with respect to $\hat{h}_i$ yields

$$\begin{aligned} \frac{\partial}{\partial \hat{h}_i}\frac{1}{2}\|h-\hat{h}\|^2 &= \hat{h}_i - h_i \ ; \\ \text{also} \ \ \frac{\partial}{\partial \hat{h}_i} l^T \hat{H}_p y &= \frac{\partial}{\partial \hat{h}_i}\sum_{j=1}^{p} l_j \sum_{i=j}^{j+r} \hat{h}_i y_{i-j} \\ &= \frac{\partial}{\partial \hat{h}_i} \sum_{j=\max(1,i-r)}^{\min(p,i)} l_j y_{i-j} \triangleq (l \star y)_i \ ; \end{aligned}$$

Differentiation with respect to y, l, λ yields $\hat{H}_p^T l + \lambda y = 0$, $\hat{H}_p y = 0$, and $y^T y = 1$. One immediately finds that $0 = y^T(\hat{H}_p^T l + \lambda y) = (\hat{H}_p y)^T l + \lambda y^T y = \lambda$.
Hence a stationary point of the optimization problem (5.6) must satisfy the necessary conditions :

$$\begin{aligned} h_i - \hat{h}_i &= (l \star y)_i & (5.8) \\ \hat{H}_p y &= 0 & (5.9) \\ \hat{H}_p^T l &= 0 & (5.10) \\ y^T y &= 1 & (5.11) \end{aligned}$$

We now introduce some new notation. For $x = \left[\begin{array}{cccc} x_1 & x_2 & \cdots & x_m \end{array} \right]^T \in \mathbb{R}^{m\times 1}$, define the $p \times (p+m-1)$ Toeplitz matrix

$$T(x,p) \triangleq \left[\begin{array}{ccccccc} x_1 & x_2 & \cdots & x_m & 0 & \cdots & 0 \\ 0 & x_1 & x_2 & \cdots & x_m & \ddots & \vdots \\ \vdots & \ddots & x_1 & x_2 & \cdots & x_m & 0 \\ 0 & \cdots & 0 & x_1 & x_2 & \cdots & x_m \end{array} \right]$$

and for a vector $z = \left[\begin{array}{cccc} z_1 & z_2 & \cdots & z_q \end{array} \right]^T \in \mathbb{R}^{q\times 1}$, $q \geq p$, define the $p\times(q-p+1)$

Hankel matrix

$$H(z,p) \triangleq \begin{bmatrix} z_1 & z_2 & \cdots & z_{q-p+1} \\ z_2 & z_3 & \cdots & z_{q-p+2} \\ \vdots & \vdots & \vdots & \vdots \\ z_p & z_{p+1} & \cdots & z_q \end{bmatrix} .$$

The following lemma shows how a product of a Hankel matrix and a vector may be rewritten in terms of such a Toeplitz matrix :

Lemma 5.1

Let $x \in \mathbb{R}^{m\times 1}$ *and* $z \in \mathbb{R}^{(p+m-1)\times 1}$,

then $H(z,p)x = T(x,p)z$.

Proof: it is easily verified that

$$[H(z,p)x]_s = \sum_{i=1}^{m} z_{i+s}x_i \text{ and } [T(x,p)z]_s = \sum_{i=1}^{m} x_i z_{i+s} . \qquad \square$$

The next lemma states that a convolution may be expressed in terms of these Toeplitz matrices :

Lemma 5.2

Let $x \in \mathbb{R}^{m\times 1}$ *and* $z \in \mathbb{R}^{p\times 1}$,

then $(T(x,p))^T z = (T(z,m))^T x = x \star z = z \star x$.

Proof: Assume without loss of generality that $m < p$.

$$\begin{aligned} \text{Then } [(T(x,p))^T z]_s &= \sum_{i=\max(1,s-m+1)}^{\min(p,s)} x_{s-i+1} z_i \\ \text{and } [(T(z,p))^T x]_s &= \sum_{j=\max(1,s-p+1)}^{\min(m,s)} z_{s-j+1} x_j = \sum_{i=\max(1,s-m+1)}^{\min(p,s)} x_{s-i+1} z_i . \end{aligned}$$

$$\text{Recall that } (x\star z)_s = (z \star x)_s \triangleq \sum_{j=\max(1,s-m+1)}^{\min(p,s)} z_j x_{s-j+1} . \qquad \square$$

Now define $H_p \triangleq H(h,p)$, $T_y \triangleq T(y,p)$ and $T_l \triangleq T(l, r+1)$; note furthermore that $\hat{H}_p = H(\hat{h},p)$, $\hat{H}_p^T = H(\hat{h}, r+1)$, $H_p^T = H(h, r+1)$.

Using the two lemmas above, one may eliminate the unknowns $\hat{h}_i$ from the necessary conditions (5.8)-(5.11) : indeed by Lemma 5.1, $h-\hat{h} = l\star y = T_l^T y = T_y^T l$, and by Lemma 5.2, $H_p y = H_p y - \hat{H}_p y = T_y(h-\hat{h})$ and $H_p^T l = H_p^T l - \hat{H}_p^T l = T_l(h-\hat{h})$, hence $H_p y = T_y T_y^T l$ and $H_p^T l = T_l T_l^T y$.
Conversely, if $H_p y = T_y T_y^T l$, $H_p^T l = T_l T_l^T y$, $\hat{h} = h - l\star y$, and $\hat{H}_p = H(\hat{h}, p)$, then $\hat{H}_p y = 0$ and $\hat{H}_p^T l = 0$.

Therefore, defining $D_y \triangleq T_y T_y^T$ and $D_l \triangleq T_l T_l^T$, equivalent necessary conditions may be formulated as the following

Theorem 5.4 *Conditions (5.8)-(5.11) are satisfied iff*

$$
\begin{aligned}
\hat{h} &= h - l\star y && (5.12)\\
H_p y &= D_y l && (5.13)\\
H_p^T l &= D_l y && (5.14)\\
\|y\| &= 1 && (5.15)
\end{aligned}
$$

In the sequel, wherever $x \in \mathbb{R}^{m\times 1}$, $D_x \triangleq T(x,m)T(x,m)^T$, unless stated otherwise. Note that each element of the matrices D_x is a quadratic function of x. As a consequence, $D_{ax} = a^2 D_x$ for each real a. It is also easily seen that D_x is invertible for every nonzero x, as $T(x,m)$ is a full rank matrix by construction.

The object function of the optimization equals

$$\frac{1}{2}\|h-\hat{h}\|^2 = \frac{1}{2}\|l\star y\| = \frac{1}{2}\|T_l^T y\|^2 = \frac{1}{2}y^T D_l y = \frac{1}{2}\|T_y^T l\|^2 = \frac{1}{2}l^T D_y l = \frac{1}{2}l^T H y \ .$$

If a solution can be found for equations (5.13),(5.14),(5.15), the corresponding $\hat{h}$ is readily found using (5.12). Remark that there is some asymmetry in these equations in the sense that $\|y\| = 1$ but $\|l\|$ is free. Defining

$$x \triangleq \frac{l}{\|l\|} \quad \text{and} \quad \sigma \triangleq \|l\| \ ,$$

equations (5.13),(5.14),(5.15) translate into

$$
\begin{aligned}
H_p y &= D_y x\sigma && (5.16)\\
H_p^T x &= D_x y\sigma && (5.17)\\
\|y\| &= 1 && (5.18)\\
\|x\| &= 1 \qquad ; && (5.19)
\end{aligned}
$$

i.e. we have an extra condition but also an extra variable σ.

The object function transforms into

$$\frac{1}{2}\|h-\hat{h}\|^2 = \frac{1}{2}y^T D_x y \sigma^2 = \frac{1}{2}x^T D_y x \sigma^2 = \frac{1}{2}x^T H y \sigma \ .$$

Since $\|x\| = \|y\| = 1$, it is to be expected that solutions with small σ will also yield a small value for the object function and vice versa; however, there is no rigorous relation, since it is impossible to predict what precise value $x^T D_y x = x^T D_y x$ will take.

In general, rescaling $x = \alpha u$ and $y = \beta v$ yields

$$\begin{aligned} H_p v &= D_v u \sigma \alpha \beta \\ H_p^T u &= D_u v \sigma \alpha \beta \qquad ; \end{aligned}$$

in particular, one may choose α and β such that $\alpha\beta = \sqrt{x^T D_y x}$. In this case $u^T D_v u = v^T D_u v = 1$, and defining $\tau \triangleq \sigma\alpha\beta$, one finds $\|h - \hat{h}\| = \tau$. This is significant: in this formulation the global minimum certainly corresponds to the smallest 'singular value' τ_{min}, in contrast to the one of (5.16)-(5.19), where this is only a heuristic. There are of course infinitely many such choices of α and β, and it is possible to impose an extra constraint, e.g. $\|u\| = \|v\|$; this yields

$$H_p v = D_v u \tau \tag{5.20}$$
$$H_p^T u = D_u v \tau \tag{5.21}$$
$$u^T D_v u = 1 = v^T D_y v \tag{5.22}$$
$$\|u\| = \|v\| \quad . \tag{5.23}$$

The structure of these equations is very reminiscent of the singular value decomposition (SVD); recall that σ is a singular value and u, v the corresponding singular vectors of a real matrix A if and only if $Av = u\sigma$ and $A^T u = v\sigma$. The only difference is the presence of the solution-dependent matrices D_u and D_v; not surprisingly however, this presence introduces all the difficulties in the problem. For an ordinary SVD e.g., there are (generically) exactly $r+1$ solutions with different σ, if $r+1$ is the smallest dimension of A; here, as all stationary points of the H_2 approximation problem must also satisfy the equations above, the latter may have many more solutions (with different σ or τ) than $r+1$.

5.2.2 Algorithms

An iterative method for solving the equations (5.13)-(5.15), (5.17)-(5.19) or (5.20)-(5.23) seems unavoidable.

Looking at the set of equations (5.17)-(5.19), one possibility is to fix the matrices D_y and D_x at each iteration and to compute the smallest nonzero generalized eigenvalue σ and corresponding eigenvector $\begin{bmatrix} x^T & y^T \end{bmatrix}^T$ of

$$\begin{bmatrix} 0 & H_p \\ H_p^T & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} D_{y_{k-1}} & 0 \\ 0 & D_{x_{k-1}} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \sigma \ .$$

Algorithm 5.1

$k = 0$
Initialize x_0, y_0
do {

$$k \leftarrow k+1$$

Solve $(H^T D_{y_{k-1}}^{-1} H)^{-1} D_{x_{k-1}} \bar{y} = \bar{y}\lambda$ *for the largest eigenvalue*

$$\begin{aligned} \sigma_k &= \lambda^{-\frac{1}{2}} \\ \bar{x} &\leftarrow D_{y_{k-1}}^{-1} H \bar{y} \sigma_k^{-1} \\ y_k &= \|\bar{y}\|^{-1} \bar{y} \\ x_k &= D_{y_{k-1}}^{-1} H y_k \\ \sigma_k &= \|x_k\| \\ x_k &\leftarrow \sigma_k^{-1} x_k \end{aligned}$$

} **while** $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - T_{x_k}^T y_k \sigma_k$

These can then be rescaled in order to satisfy the normalization constraints $x^T x = y^T y = 1$, and be used to construct D_x and D_y for the next iteration. This iteration is repeated until x and y (hopefully) converge. The generalized eigenvalue problem is equivalent to finding the smallest eigenvalue σ^2 and corresponding eigenvector y of $D_{x_{k-1}}^{-1} H^T D_{y_{k-1}}^{-1} H y = y\sigma^2$, or also the largest eigenvalue σ^{-2} and corresponding eigenvector y of $(H^T D_{y_{k-1}}^{-1} H)^{-1} D_{x_{k-1}} y = y\sigma^{-2}$, and taking $x = \sigma^{-1} D_{y_{k-1}}^{-1} H y$. This gives us Alg. 5.1.

The same can be done for eqs. (5.20)-(5.23), where D_u and D_v are kept fixed each iteration, and the smallest nonzero generalized eigenvalue τ and corresponding eigenvector $\begin{bmatrix} u^T & v^T \end{bmatrix}^T$ of

$$\begin{bmatrix} 0 & H_p \\ H_p^T & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} D_{v_{k-1}} & 0 \\ 0 & D_{u_{k-1}} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \tau$$

are computed; u and v are then rescaled in such a way that

$$u^T D_v u = v^T D_y v = 1 \text{ and } \|u\| = \|v\| \, .$$

The generalized eigenvalue problem is again equivalent to finding the largest eigenvalue τ^{-2} and corresponding eigenvector v of $(H^T D_{v_{k-1}}^{-1} H)^{-1} D_{u_{k-1}} v = v\tau^{-2}$, and taking $u = \tau^{-1} D_{v_{k-1}}^{-1} Hv$. This results in Alg. 5.2.

Algorithm 5.2

$k = 0$
Initialize u_0, v_0
do {

$$k \leftarrow k + 1$$

Solve $(H^T D_{v_{k-1}}^{-1} H)^{-1} D_{u_{k-1}} \bar{v} = \bar{v}\lambda$ *for the largest eigenvalue*

$$\begin{aligned} \tau_k &= \lambda^{-\frac{1}{2}} \\ \bar{u} &\leftarrow D_{v_{k-1}}^{-1} H \bar{v} \tau_k^{-1} \\ \alpha &\leftarrow \frac{\sqrt{\|\bar{v}\|}}{\sqrt{\|\bar{u}\|} \sqrt[4]{\bar{u}^T D_{\bar{v}} \bar{u}}} \\ \beta &\leftarrow \frac{\sqrt{\|\bar{u}\|}}{\sqrt{\|\bar{u}\|} \sqrt[4]{\bar{u}^T D_{\bar{v}} \bar{v}}} \\ v_k &\leftarrow \beta \bar{v} \\ u_k &\leftarrow \alpha \bar{u} \end{aligned}$$

} **while** $\|v_k - v_{k-1}\| > \epsilon$
$\hat{h} = h - T_{u_k}^T v_k \tau_k$

This type of iterative method might be a rather expensive option however, as each iteration requires an eigenvalue decomposition.

Seeking inspiration in existing methods for eigenvalue problems, we consider the power method [116] , an iteration converging (under mild conditions) to the largest eigenvalue λ and corresponding eigenvector x of a square matrix M .

power method (eigenvalue version)

$k = 0$
Initialize x_0
do {

$$
\begin{aligned}
k &\leftarrow k+1 \\
x_k &= M x_{k-1} \\
x_k &\leftarrow \frac{x_k}{\|x_k\|} \\
\lambda_k &= x_k^T M x_k
\end{aligned}
$$

} **while** $\|x_k - x_{k-1}\| > \epsilon$

Under certain (generically fulfilled) conditions, λ_k converges to λ and x_k to a corresponding eigenvector x of unit length. If A is invertible, the procedure may of course also be performed on A^{-1}, yielding the inverse of the smallest eigenvalue of A and a corresponding eigenvector.

To find the smallest singular value σ and corresponding singular vectors u and v of a non-square matrix A, it is not possible to apply the iteration

$$
\begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} = \begin{bmatrix} u_{k-1} \\ v_{k-1} \end{bmatrix}
$$

followed by normalization, since $\begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}$ is not invertible; instead, one can compute u_k and v_k from $Av_k = u_k$ and $A^T u_k = v_{k-1}$, resulting in
This power iteration is also applicable to the solution-dependent eigenvalue problems dealt with here. As there are several, but equivalent forms in which the equations can be written, e.g. formulas (5.13)-(5.15), (5.17)-(5.19), and (5.21)-(5.23), this will render also several power iteration algorithms; however, these are no longer mathematically equivalent.

power method (singular value version)

$k = 0$
Initialize u_0, v_0
do {

$$\begin{array}{rcl} k & \leftarrow & k+1 \\ v_k & = & A^T A v_{k-1} \\ u_k & = & A v_k \\ \sigma_k & = & \|v_k\| \\ \begin{bmatrix} u_k \\ v_k \end{bmatrix} & \leftarrow & \dfrac{1}{\sigma_k} \begin{bmatrix} u_k \\ v_k \end{bmatrix} \end{array}$$

} **while** $\|v_k - v_{k-1}\| > \epsilon$

Application of the power iteration idea to (5.13)-(5.15), keeping D_l and D_y fixed each iteration, yields Alg. 5.3.

Note that the equations

$$\begin{array}{rcl} y_k & = & (H^T D_{y_{k-1}}^{-1} H)^{-1} D_{l_{k-1}} y_{k-1} \\ l_k & = & D_{y_{k-1}}^{-1} H y_k \end{array}$$

are equivalent to

$$\begin{bmatrix} H & -D_{y_{k-1}} \\ 0 & H^T \end{bmatrix} \begin{bmatrix} y_k \\ l_k \end{bmatrix} = \begin{bmatrix} 0 \\ D_{l_{k-1}} y_{k-1} \end{bmatrix}. \tag{5.24}$$

A power iteration algorithm tailored for equations (5.20)-(5.23), looks as follows : As solutions for equations (5.13)-(5.15) can be scaled into solutions for equations (5.20)-(5.23) and vice versa, one might be led to believe that Algorithms 5.3 and 5.4 are also equivalent. This is not so, however : only the solutions of one set can be transformed into the solutions of the other set, the same cannot be said of the iterates (y_k,l_k) and (v_k,u_k). In fact Alg. 5.4 seems to be the most suitable one, since as the inverse power method is designed to converge towards the smallest singular value, one may expect Alg. 5.3 to be a good algorithm for minimizing σ, and Alg. 5.4 for minimizing τ. As the quantity to be minimized equals $\|h-\hat{h}\| = \sigma\|x\star y\| = \tau$, Alg. 5.4 minimizes more directly the approximation error.

Algorithm 5.3

$k = 0$
Initialize y_0, l_0
do {

$$
\begin{aligned}
k &\leftarrow k+1 \\
y_k &= (H^T D_{y_{k-1}}^{-1} H)^{-1} D_{l_{k-1}} y_{k-1} \\
l_k &= D_{y_{k-1}}^{-1} H y_k \\
\begin{bmatrix} y_k \\ l_k \end{bmatrix} &\leftarrow \frac{1}{\|y_k\|} \begin{bmatrix} y_k \\ l_k \end{bmatrix}
\end{aligned}
$$

} **while** $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - T_{l_k}^T y_k$

Because both algorithms do not necessarily render the same stationary point, this seems to indicate that Alg. 5.4 has a better chance of ending up in the global minimum. In general both algorithms yield the same solution, presumably generally the global optimum; but indeed most often, Alg. 5.4 needs fewer iterations.

Convergence has not been proven, not even local convergence in a small open set around a minimum, but no examples are known where there is no local convergence in a local minimum. The fact that this is not a gradient-based descent-type algorithm is of course a factor making such a proof difficult. Local convergence in every local minimum is not really desirable; of course global convergence to the global optimum is what one would like, or failing that, at least local convergence in the global optimum; but as no properties that distinguish the global minimum from other local minima have been characterized (except, obviously, the definition of the global minimum), this has been beyond our grasp.

Numerous examples however suggest that the zone of attraction for local minima different from the global minimum is often smaller for these STLS-based algorithms than for gradient-based descent algorithms; see the example at the end of this chapter.

Algorithm 5.4

$k = 0$
Initialize u_0, v_0
do {

$$\begin{array}{rcl} k & \leftarrow & k+1 \\ v_k & = & (H^T D_{v_{k-1}}^{-1} H)^{-1} D_{u_{k-1}} v_{k-1} \tau_{k-1}^2 \\ u_k & = & D_{v_{k-1}}^{-1} H v_k \tau_{k-1}^{-1} \\ \gamma & \leftarrow & \sqrt{\|T_{u_k}^T v_k\|} \\ \alpha & \leftarrow & \dfrac{\|v_k\|}{\|u_k\|} \\ u_k & \leftarrow & \alpha \gamma^{-1} u_k \\ v_k & \leftarrow & \alpha^{-1} \gamma^{-1} v_k \\ \tau_k & = & u_k^T H v_k \end{array}$$

} **while** $\|v_k - v_{k-1}\| > \epsilon$
$\hat{h} = h - T_{u_k}^T v_k \tau_k$

Stopping criteria

In the algorithms above, the iteration was always terminated when $\|v_k - v_{k-1}\|$ or $\|y_k - y_{k-1}\|$ was smaller than a certain tolerance ϵ. This criterion is only adopted because it is the simplest one; there is no reason why this should be superior, however, and other stopping criteria may be thought of as well.

One might monitor the change of the object function, i.e. take as stopping criterion $|\, y_k^T D_{l_k} l_k - y_{k-1}^T D_{l_{k-1}} l_{k-1}| < \epsilon^2$ or $|\, \|T_{l_k} y_k\| - \|T_{l_{k-1}} y_{k-1}\| \,| < \epsilon$. Note that the object function does not always need to decrease, which is why the absolute value of the change in object function is taken, but as one converges towards a minimum it is of course to be expected that the object criterion will start decreasing at some point. The latter stopping criterion is more useful for comparison between different algorithms; indeed, $\|v_k - v_{k-1}\| < \epsilon$ in Alg. 5.4 is not equivalent to $\|y_k - y_{k-1}\| < \epsilon$ in Alg. 5.3, but $|\, \|T_{l_k} y_k\| - \|T_{l_{k-1}} y_{k-1}\| \,| < \epsilon$ and $|\, \|T_{u_k} v_k\| - \|T_{u_{k-1}} v_{k-1}\| \,| < \epsilon$ is much more indicative that both algorithms have converged to within the same distance of the minimum.

Yet another, fairly secure, but expensive stopping criterion is

$$\frac{\sigma_{r+1}(\hat{H}_p)}{\bar{\sigma}(\hat{H}_p)} < 1 \; .$$

No stopping criteria have been derived that guarantee being close to an optimum.

5.2.3 Speeding up the time domain algorithm

It should be noted that the computational burden of one iteration, even though it is much better than in Alg. 5.1, is still quite heavy. Recall that the more closely one wishes to approximate the original model reduction problem, the larger one should choose the number of Markov parameters p. The most costly operation is the computation of the inverse $D_{y_{k-1}}^{-1}$, or more precisely, the computation of $D_{y_{k-1}}^{-1} H$, which can done by solving the linear system $D_{y_{k-1}} X = H$. Indeed, as $D_{y_{k-1}}$ is a $p \times p$ matrix, this requires $\frac{2}{3}p^3 + 2p^2(r+1)$ flops.

This subsection presents a couple of modest efficiency improvements; a much more substantial speed-up for our problem is found in Section 5.4. Still, the tricks of this subsection have their use, as they may be employed more generally, e.g. for the noisy realization problem.

Standard STLS algorithm

A small improvement may be obtained by using an algorithm proposed in [76], suitable not only for this type of problem but for any STLS problem, whereby the largest inverse to be computed can be reduced to size $(p-r-1) \times (p-r-1)$. This is explained in the next paragraphs.

Let $Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}$ be an orthogonal matrix and R an upper triangular $(r+1) \times (r+1)$ matrix such that $H = Q_1 R$, i.e. a QR decomposition of H.
In [76], two new variables $v_k \triangleq Q_1^T l_k$ and $w_k \triangleq Q_2^T l_k$ are defined;
$H^T l_k = D_{l_{k-1}} y_{k-1}$ then translates to $R^T v_k = D_{l_{k-1}} y_{k-1}$, and $H y_k = D_{y_{k-1}} l_k$ to $R y_k = Q_1^T D_{y_{k-1}}(Q_1 v_k + Q_2 w_k)$ and $0 = Q_2^T D_{y_{k-1}} Q_1 v_k + Q_2^T D_{y_{k-1}} Q_2 w_k$. This yields Alg. 5.5.

Remark that the largest inverse appearing here is $(Q_2^T D_{y_{k-1}} Q_2)^{-1}$, which does indeed have size $(p-r-1) \times (p-r-1)$. The flop count for one iteration is about $\frac{2}{3}p^3 + (9-2r)p^2$, compared to $\frac{2}{3}p^3 + (4r+4)p^2$ for Alg. 5.3. Admittedly the improvement is not overwhelming. In fact Alg. 5.5 was proposed before Alg. 5.3, which is only included here because it is the algorithm that directly inspires the

Algorithm 5.5

$k = 0$
Initialize y_0, l_0
do {

$$\begin{aligned}
k &\leftarrow k+1 \\
v_k &= R^{-T} D_{l_{k-1}} y_{k-1} && (5.25) \\
w_k &= -(Q_2^T D_{y_{k-1}} Q_2)^{-1} (Q_2^T D_{y_{k-1}} Q_1) v_k && (5.26) \\
l_k &= Q_1 v_k + Q_2 w_k \\
y_k &= R^{-1} Q_1^T D_{y_{k-1}} l_k && (5.27) \\
\begin{bmatrix} y_k \\ l_k \end{bmatrix} &\leftarrow \frac{1}{\|y_k\|} \begin{bmatrix} y_k \\ l_k \end{bmatrix} \\
\sigma_k &= \|l_k\|
\end{aligned}$$

} **while** $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - T_{l_k}^T y_k$

algorithms in Section 5.4.

This algorithm is easily seen to be mathematically equivalent to Alg. 5.3 : indeed, equations (5.25)-(5.27) may be rewritten

$$\begin{bmatrix} R^T & 0 & 0 \\ Q_2^T D_{y_{k-1}} Q_1 & Q_2^T D_{y_{k-1}} Q_2 & 0 \\ Q_1^T D_{y_{k-1}} Q_1 & Q_1^T D_{y_{k-1}} Q_2 & -R \end{bmatrix} \begin{bmatrix} v_k \\ w_k \\ y_k \end{bmatrix} = \begin{bmatrix} D_{l_{k-1}} y_{k-1} \\ 0 \\ 0 \end{bmatrix} \qquad (5.28)$$

Multiplying equation (5.28) from the left by $\begin{bmatrix} 0 & -Q_2 & -Q_1 \\ I & 0 & 0 \end{bmatrix}$ and taking into account that $\begin{bmatrix} v_k \\ w_k \end{bmatrix} = Q^T l_k$, we find this in turn to be equivalent to:

$$\begin{bmatrix} 0 & -Q_2 & -Q_1 \\ I & 0 & 0 \end{bmatrix} \begin{bmatrix} R^T & 0 & 0 \\ Q_2^T D_{y_{k-1}} Q_1 & Q_2^T D_{y_{k-1}} Q_2 & 0 \\ Q_1^T D_{y_{k-1}} Q_1 & Q_1^T D_{y_{k-1}} Q_2 & -R \end{bmatrix} \begin{bmatrix} 0 & Q_1^T \\ 0 & Q_2^T \\ I & 0 \end{bmatrix} \cdot$$

$$\begin{bmatrix} 0 & 0 & I \\ Q_1 & Q_2 & 0 \end{bmatrix} \begin{bmatrix} v_k \\ w_k \\ y_k \end{bmatrix} = \begin{bmatrix} 0 & -Q_2 & -Q_1 \\ I & 0 & 0 \end{bmatrix} \begin{bmatrix} D_{l_{k-1}} y_{k-1} \\ 0 \\ 0 \end{bmatrix}$$

$$\text{i.e.} \qquad \begin{bmatrix} H & D_{y_{k-1}} \\ 0 & H^T \end{bmatrix} \begin{bmatrix} y_k \\ l_k \end{bmatrix} = \begin{bmatrix} 0 \\ D_{l_{k-1}} y_{k-1} \end{bmatrix};$$

this is precisely equation (5.24). As the normalization is also done in the same way in both algorithms, the same iterates y_k and l_k will be produced, at least if the calculations are done with infinite precision.

Toeplitz structure

Alg. 5.5 does not turn to advantage any knowledge about the structure of $D_{y_{k-1}}$ and $D_{l_{k-1}}$; it can be used therefore for any STLS problem. This is why it presented in [74] as the standard algorithm for iteratively solving STLS problems.

In the case studied here however, D_y and D_l are symmetric Toeplitz matrices, a fact which can be exploited to reduce the number of computations per iteration. Moreover, D_y is banded, meaning that all entries are zero except in a band of width $2r+1$ around the diagonal.

The Toeplitz structure can be used in two ways : first, it is evidently only necessary to compute the first row of a Toeplitz matrix. This way the amount of computation needed to compute D_y is reduced from $p^2(p+r+1)$ to $\frac{1}{2}r(r+1)$ multiplications, and for D_l from $(r+1)^2(p+r+1)$ to $(r+1)(p+r+1)$.

Second, there are fast algorithms for computing inverses of Toeplitz matrices and solving Toeplitz systems. The inverse of a Toeplitz matrix of size p can be computed in $O(3.25p^2)$ flops using Trench's algorithm [266, 267]. Therefore $D_{y_{k-1}}^{-1} H$ can be computed in $O((5.25+2r)p^2)$ flops, the multiplication of $D_{y_{k-1}}^{-1}$ and H needing $O((2r+2)p^2)$ flops. In fact, using Levinson's algorithm [173], $D_{y_{k-1}}^{-1} H$ can be computed directly in $O((4+2r)p^2)$ flops.

The Toeplitz structure of D_l is not important since the number of flops connected with computations involving D_l is small, due to its small size $(r+1) \times (r+1)$. The gain, therefore, is to be expected primarily in the computation of $D_{y_{k-1}}^{-1} H$.

Skipping updates of D_y and D_l

In the algorithms above, D_y and D_l are updated each iteration; this is quite expensive, as the most costly operation is solving a linear system $D_{y_{k-1}}X = H$ or $(Q_2^T D_{y_{k-1}} Q_2)X = (Q_2^T D_{y_{k-1}} Q_1)v_k$. Yet $D_{l_{k-1}}$ and $D_{y_{k-1}}$ may still yield quite good estimates for D_{l_k} and D_{y_k}, and one might consider keeping D_y and D_l fixed for a couple of iterations. One may update D_l and D_y at a different rate, i.e. update D_l every n_l iterations, and D_y every n_y iterations. This is implemented in Alg. 5.6.
How many updates one should skip is hard to determine theoretically, and will depend on the data.

In the limit, choosing $n_y = n_l$ and letting the number of power iterations per update n_y grow towards infinity, one is performing an inverse power iteration between every two updates; this is equivalent to taking an eigenvalue decomposition of $(H^T D_{y_k}^{-1} H)^{-1} D_{l_k}$ each iteration and taking the eigenvector belonging to the biggest eigenvalue for the next iterate y_{k+1}. Therefore, Alg. 5.6 becomes equivalent to Alg. 5.1 as $n_y \to \infty$.

Algorithm 5.6

$k = 0$
Initialize $y_0,\ l_0$
do {

$$\begin{array}{rcl}
\textbf{if} \ (\ k \in n_y\mathbb{Z}\ ) & & \{\ y \leftarrow y_k\} \\
\textbf{if} \ (\ k \in n_l\mathbb{Z}\ ) & & \{\ l \leftarrow l_k\} \\
k & \leftarrow & k+1 \\
y_k & = & (H^T D_y^{-1} H)^{-1} D_l y_{k-1} \\
y_k & \leftarrow & \dfrac{y_k}{\|y_k\|} \\
l_k & = & D_y^{-1} H y_k
\end{array}$$

} **while** $\|y_k - y_{k-n_y}\| > \epsilon$
$\hat{h} = h - T_{l_k}^T y_k$

Skipping updates can also be applied to Alg. 5.5, which of course turns out to be equivalent to Alg. 5.6, and also to Alg. 5.4, which 'converges to' Alg. 5.2 as the number of iterations per update grows.

5.3 Multiple Input Multiple Output

Even though SISO (or Single Sensor Single Actuator) controllers are certainly not uncommon in practice, the ability of modern control theory of handling truly multivariable controllers is commonly regarded as one of its important advantages. A natural question, therefore, is whether the STLS-based model reduction method described here can be extended to Multiple Input Multiple Output systems.

This is possible for Single Input Multiple Output (SIMO) systems $h(z) \in \mathbb{R}^{m\times 1}$, which is shown in the next paragraph; reduction of a Multiple Input Single Output (MISO) system $h(z) \in \mathbb{R}^{m\times 1}$, can of course be recast trivially as a SIMO problem by reducing the transpose $(h(z))^T$.

Unfortunately a straightforward generalization is impossible in the truly multivariable case. Indeed, expressing that the order of $\hat{h}(z)$ does not exceed r as a rank constraint on the first $r+1$ columns of the Hankel matrix is only possible for systems with one input; for more inputs, the last *block* column should be linearly dependent on the preceding block columns, hence y should be a matrix with as many columns as there are inputs. Lemmas 5.1 and 5.2 can then no longer be generalized.

For a system with one input and m outputs, y remains a $(r+1)\times 1$ vector, while the Markov parameters h_i as well as the corresponding Lagrange multipliers l_i are now $m \times 1$ vectors instead of scalars. Thus, the vectors $h \triangleq \begin{bmatrix} h_1^T & \dots & h_p^T \end{bmatrix}^T$ and $l \triangleq \begin{bmatrix} l_1^T & \dots & l_p^T \end{bmatrix}^T$ exhibit block structure in the column space. There are no essential changes in the derivation of the algorithm; it shall not be duplicated completely here. The result of Lemma 5.1, $H_p y = T_y h$, translates trivially into $H_p y = (T_y \otimes I_m)h$, while $H_p^T l = T_l h$ is still valid if the definition of T_l is generalized to

$$T_l \triangleq \begin{bmatrix} l_1^T & l_2^T & \cdots & l_p^T & 0 & \cdots & 0 \\ 0 & l_1^T & l_2^T & \cdots & l_p^T & \ddots & \vdots \\ \vdots & \cdots & l_1^T & l_2^T & \cdots & l_p^T & 0 \\ 0 & \cdots & 0 & l_1^T & l_2^T & \cdots & l_p^T \end{bmatrix},$$

where $T_l \in \mathbb{R}^{(r+1)\times m(p+r)}$.

On the other hand, one has $h-\hat{h} = l\star y = T_l^T y = (T_y^T \otimes I_m)l$; hence the necessary conditions (5.13) and (5.14) become $Hy = (H-\hat{H})y = (T_y \otimes I_m)(T_y^T \otimes I_m)l = (D_y \otimes I_m)l$ and $H^T l = (H^T - \hat{H}^T)l = T_l T_l^T y$.
Alg. 5.3 needs but minor adaptations, and turns into Alg. 5.7.

Algorithm 5.7

$k = 0$
Initialize y_0, l_0
do {

$$\begin{array}{rcl} k & \leftarrow & k+1 \\ y_k & = & (H^T(D_{y_{k-1}}^{-1} \otimes I_m)H)^{-1} D_{l_{k-1}} y_{k-1} \\ l_k & = & (D_{y_{k-1}}^{-1} \otimes I_m)Hy_k \\ \begin{bmatrix} y_k \\ l_k \end{bmatrix} & \leftarrow & \frac{1}{\|y_k\|} \begin{bmatrix} y_k \\ l_k \end{bmatrix} \end{array}$$

} **while** $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - T_{l_k}^T y_k$

Note that because of the fact that $(D_{y_{k-1}} \otimes I_m)^{-1} = (D_{y_{k-1}}^{-1} \otimes I_m)$, the flop count per iteration is not much different from the SISO case, as the most expensive operation may be done at roughly the same cost.

5.4 z-domain algorithm

Although the algorithms given in Subsection 5.2.3 offer some improvement with respect to the computational burden, it should be noted that in all of them the number of flops per iteration is still heavily dependent on the number of Markov parameters p considered in the minimization criterion. Ultimately we are interested in the infinite horizon problem here, i.e. $p \to \infty$, and thus we are confronted with a tradeoff between accuracy of the solution on the one hand and computation time and computer memory requirements on the other hand.

Representing the iterates l_k of length p by their z-transform $l_p^{[k]}(z) \triangleq \sum_{i=1}^{p} l_i z^{-i}$, one might study the power series

$$l^{[k]}(z) \triangleq \lim_{p\to\infty} l_p^{[k]}(z),$$

and examine whether this limit, if it exists, displays any structure that can be taken advantage of. Indeed experiments quickly show that an IIR realization of $l_p^{[k]}(z)$ has some poles that get closer to the poles of $h(z)$ as p grows; a closer look will substantiate this, and reveal more: $l_k(z)$ turns out to be a rational function of z, of which the poles are known. Using this knowledge, the algorithms presented in Section 5.2 can be expressed in terms of $y(z)$ and a finite number of parameters specifying $l_k(z)$, instead of the infinite number of Lagrange multipliers l_i.

In order to show this, first some notation, definitions and facts concerning power series are introduced. $h(z)$ is a SISO strictly proper system of order n, and $\hat{h}(z)$ a strictly proper system of order r.

$$\begin{aligned} \text{Let } h(z) &= \frac{b(z)}{a(z)} \\ \hat{h}(z) &= \frac{\hat{b}(z)}{\hat{a}(z)} \end{aligned}$$

where $b(z)$, $a(z)$, $\hat{b}(z)$, $\hat{a}(z)$ are real polynomials in z with $\deg b(z) = n-1$, $\deg a(z) = n$, $\deg \hat{b}(z) = r-1$, and $\deg \hat{a}(z) = r$. Let α_i, $i = 1,\ldots,n$, be the poles of $h(z)$, and $\hat{\alpha}_i$, $i = 1,\ldots,r$, be the poles of $\hat{h}(z)$.

$\mathcal{Z}$ is the function mapping a vector x on a corresponding power series $x(z)$ as follows:

$$\mathcal{Z} : x \in \mathrm{I\!R}^{q\times 1} \to x(z) = \sum_{i=1}^{q} x_i z^{-i} \ .$$

The following notation defined for a formal series $x(z) \triangleq \sum_{-\infty}^{\infty} x_i z^{-i}$:

$(x(z))^{\sim} \triangleq x(z^{-1})$,

$[x(z)]_i \triangleq x_i$: i-th Markov parameter of $x(z)$,

$[x(z)]_{0+} \triangleq \sum_{i=0}^{\infty} x_i z^{-i}$: proper or *causal* part of $x(z)$,

$[x(z)]_{+} \triangleq \sum_{i=1}^{\infty} x_i z^{-i}$: strictly proper or *strictly causal* part of $x(z)$,

$[x(z)]_{i\ldots j} \triangleq \sum_{s=i}^{j} x_i z^{-i}$: Markov parameters i through j of $x(z)$.

T is the unit circle in the complex plane.

A polynomial is *Hurwitz* if its roots are all lying inside the unit disk $|z| < 1$, and *anti-Hurwitz* if its roots are all lying outside the unit disk.

$L_2 \triangleq \{x(z) | \; \|x(z)\|_2 < \infty\}$
$L_2^+ \triangleq \{x(z) \in L_2 | \; x(z) = [x(z)]_+\}$ (the set of strictly proper functions $\in L_2$)
$L_2^{0+} \triangleq \{x(z) \in L_2 | \; z^{-1}x(z) \in L_2^+\}$ (i.e. the Hardy space H_2)
$RL_2 : \{x(z) \in L_2 \mid x(z)$ converges to a rational function on $T\}$
$RL_2^+ : RL_2 \cap L_2^+$
$RL_2^{0+} : RL_2 \cap L_2^{0+}$

A series $x(z) \triangleq \sum_{-\infty}^{\infty} x_i z^{-i}$ is *absolutely convergent* in a complex number z if the series $\sum_{-\infty}^{\infty} |x_i||z|^{-i}$ is convergent.

5.4.1 Some facts about power series

The following facts are standard and can be found e.g. in [158].

Lemma 5.3
Let $x(z) \triangleq \sum_{i=-\infty}^{\infty} x_i z^{-i}$ *be a formal series,* $y(z) \triangleq \sum_{i=-\infty}^{\infty} y_i z^{-i}$ *be absolutely convergent in* z*, and let* $|x_i| \leq |y_i|$*,* $\forall i$*,*
Then $x(z)$ *is absolutely convergent in* z*.*

Proof: see [158]. $\square$

Lemma 5.4
Let $x_i z^{-i}$ *be bounded in* z_1*, i.e.* $|x_i z_1^{-i}| < A < \infty$*.*
Then $x(z) \triangleq \sum_{i=0}^{\infty} x_i z^{-i}$ *is absolutely convergent for all* z *such that* $|z| > |z_1|$*.*

Proof: $|x_i| = |x_i z_1^{-i}||z_1^i| < A|z_1^i|$, and $\sum_{i=0}^{\infty} z_1^i z^{-i}$ is absolutely convergent for $|z| > |z_1|$. $\square$

Lemma 5.5
Let $x(z) \triangleq \sum_{i=0}^{\infty} x_i z^{-i}$ *be convergent in* z_1*,*
Then $x(z)$ *is absolutely convergent for all* z *such that* $|z| > |z_1|$*.*

Proof: Since $x(z)$ is convergent in z_1, $x_i z_1^{-i}$ is bounded. $\square$

Lemma 5.6
a) A series $x(z) \in L_2^+$ is absolutely convergent for all z satisfying $|z| > 1$.
b) A series $x(z) \in RL_2^+$ converges for all z satisfying $|z| > \alpha$ (for some real $\alpha < 1$).
c) A series $x(z) \in RL_2$ converges for all z satisfying $\alpha < |z| < \beta$ (where $\alpha, \beta \in \mathbb{R}_0^+$ and $\alpha < 1 < \beta$).

Proof:
a) Since $\|x(z)\|_2 = A < \infty$, $|x_i| \leq \sqrt{A}$, and absolute convergence follows from Lemma 5.4.
b) It is easily seen, by expanding $x(z)$ into partial fractions, that α is the largest of the moduli of the poles p_k of $x(z)$;

indeed, the series $\displaystyle\sum_{i=1}^{\infty} p_k^{i-1} z^{-i}$ converges to $\dfrac{1}{z - p_k}$ wherever $|z| > |p_k|$

by application of Lemma 5.4.
c) This follows from b) applied to $[x(z)]_+$ and $\left[x(z^{-1})\right]_+$. □

Lemma 5.7
If two series $x(z)$ and $y(z)$ converge in open regions S_x resp. S_y such that $S_x \cap S_y$ is nonempty, then the convolution $x(z)y(z)$ has well-defined coefficients and converges in $S_x \cap S_y$.

5.4.2 Translation of the necessary conditions

In examining how the quantities in Alg. 5.3 will behave as the time horizon p grows towards infinity, one should understand how the operations of Lemma 5.1 and Lemma 5.2 translate to the z-domain.

Let $x \in \mathbb{R}^{m\times 1}$ with $x(z) = \mathcal{Z}[x]$, $v \in \mathbb{R}^{(p+m-1)\times 1}$ with $v(z) = \mathcal{Z}[v]$, then it is but a formal manipulation to show that

$$\mathcal{Z}[H(v,p)x] = \mathcal{Z}[T(x,p)v] = \left[z^{-1}v(z)x(z^{-1})\right]_{1\ldots p} .$$

$$\text{Indeed, } \left[z^{-1}v(z)x(z^{-1})\right]_s = \sum_{i=1}^{m} x_i v_{i+s} = [H(v,p)x]_s .$$

Similarly, if $w \in \mathbb{R}^{p\times 1}$, with $w(z) = \mathcal{Z}[w]$, then

$$\mathcal{Z}[(T(x,p))^T w] = \mathcal{Z}[(T(w,m))^T x] = z^{-1}x(z)w(z) .$$

$$\text{Defining } y(z) \triangleq z\mathcal{Z}[y] = \sum_{i=0}^{r} y_i z^{-i} ,$$

application of the formulas above yields

$$\begin{aligned}\mathcal{Z}[H_p y] &= \left[h(z)y(z^{-1})\right]_{1\ldots p}\\ \mathcal{Z}[H_p^T l] &= \left[h(z)l_p(z^{-1})\right]_{0\ldots r}\\ \mathcal{Z}[T_y T_y^T l] &= \left[(y(z)l_p(z^{-1}))y(z^{-1})\right]_{1\ldots p}\\ \mathcal{Z}[T_l T_l^T y] &= \left[(l_p(z)y(z))l_p(z^{-1})\right]_{0\ldots r}\end{aligned}$$

This gives little more than an alternative notation for eqs. (5.13)-(5.14). It is another matter to take the limit $p \to \infty$, and to state, with $l(z) \triangleq \lim_{p\to\infty} l_p(z)$,

that $\lim\limits_{p\to\infty} \mathcal{Z}[H_p y] = \left[h(z)y(z^{-1})\right]_+$ and $\lim\limits_{p\to\infty} \mathcal{Z}[H_p^T l] = \left[h(z)l(z^{-1})\right]_{0\ldots r}$.

For this, one needs to prove that all these limits exist, or in other words, that the power series converge, at least on some region of the complex plane.

We will go rather lightly over this matter, by assuming that $l(z) \in L_2^+$, i.e. that $l(z)$ is square integrable. This is certainly the case in a minimum. Indeed, for the object function to exist, $h(z) - \hat{h}(z)$ must be square integrable, and since it is a rational function of z it follows that $h(z) - \hat{h}(z) \in RL_2^+$. Any $x(z) \in L_2^+$ is convergent in $\{z \in \mathbb{C} \mid |z| > 1\}$, hence the formal series $l(z)y(z) = \lim_{p\to\infty} l_p(z)y(z)$ is convergent when $|z| > 1$.

Therefore, in a stationary point of the H_2-optimal model reduction problem, the following conditions must be satisfied :

$$\begin{aligned}\hat{h}(z) &= h(z) - y(z)l(z)\\ \left[h(z)y(z^{-1})\right]_+ &= \left[y(z^{-1})y(z)l(z)\right]_+\\ \left[h(z)l(z^{-1})\right]_{0\ldots r} &= \left[l(z^{-1})l(z)y(z)\right]_{0\ldots r}\\ \|y(z)\| &= 1\end{aligned}$$

Note that the order constraint $\mathcal{Z}[\hat{H}_p y] = 0$ translates into $\left[\hat{h}(z)y(z^{-1})\right]_+ = 0$ as p goes to infinity. $y(z^{-1})$ is a polynomial in z, and in fact $y(z^{-1}) = \hat{a}(z)$ satisfies this condition. Indeed, every $y(z^{-1})$ satisfying $\left[\hat{h}(z)y(z^{-1})\right]_+ = 0$ obeys $y(z^{-1}) = c\hat{a}(z)$ for some $c \in \mathbb{R}$. This is easily seen by requiring that the residues of $\left[\hat{h}(z)y(z^{-1})\right]_+$ vanish in the poles $\hat{\alpha}_i$ (see Lemma A.1); i.e. $\lim_{z\to\hat{\alpha}_i}(z - \hat{\alpha}_i)\hat{h}(z)(z)y(\hat{\alpha}_i^{-1})$. As the optimal approximant $\hat{h}(z)$ is never non-minimal, $\lim_{z\to\hat{\alpha}_i}(z - \hat{\alpha}_i)\hat{h}(z) \neq 0$, which means that $y(\hat{\alpha}_i^{-1}) = 0$, $i = 1, \ldots, r$, and $y(z^{-1})$ must have the form $c\hat{a}(z)$.

By analogy, we introduce the notation $\hat{a}^{[k]}(z) \triangleq y^{[k]}(z^{-1})$.
Furthermore, $\hat{a}_r^{[k]}(z) \triangleq z^r \hat{a}^{[k]}(z^{-1}) = z^r y^{[k]}(z)$.

5.4.3 Translation of the algorithm

In a similar way one can see that, with the definitions $y^{[k]}(z) \triangleq \lim_{p\to\infty} \mathcal{Z}[y_k]$ and $l^{[k]}(z) \triangleq \lim_{p\to\infty} \mathcal{Z}[l_k]$, (5.24) translates into

$$\begin{aligned} \left[h(z)y^{[k]}(z^{-1})\right]_+ &= \left[y^{[k-1]}(z^{-1})y^{[k-1]}(z)l^{[k]}(z)\right]_+ \\ \left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} &= \left[l^{[k-1]}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_{0\ldots r} \end{aligned}$$

or equivalently :

$$\left[\frac{b(z)}{a(z)}\hat{a}^{[k]}(z)\right]_+ = \left[z^{-r}\hat{a}_r^{[k-1]}(z)\hat{a}^{[k-1]}(z)l^{[k]}(z)\right]_+ \tag{5.29}$$

$$\left[\frac{b(z)}{a(z)}l^{[k]}(z^{-1})\right]_{0\ldots r} = \left[l^{[k-1]}(z^{-1})l^{[k-1]}(z)\hat{a}^{[k-1]}(z^{-1})\right]_{0\ldots r} \tag{5.30}$$

Just to make life easier, we shall adopt the technical assumption that $h(z)$ and $\hat{h}(z)$ have no multiple poles, and that $h(z)$ has no poles at $z = 0$. Furthermore we assume for the moment that $l^{[k]}(z) \in RL_2^+$; it will be shown that at least there exists a solution $l^{[k]}(z)$ for which this is true.

The left hand side of (5.29) is rational and has only the zeroes α_i of $a(z)$ for poles; hence on the right hand side, the multiple pole at $z = 0$ must be canceled by zeroes of $l^{[k]}(z)$. The poles of the right hand side are those poles of $l^{[k]}(z)$ that are not canceled by the LHP zeroes of $\hat{a}_r^{[k-1]}(z)\hat{a}^{[k-1]}(z)$, as $l^{[k]}(z)$ is stable. If $\hat{a}^{[k-1]}(z)$ has n_a RHP roots, it can be factored as $\hat{a}^{[k-1]}(z) = \hat{a}_a^{[k-1]}(z)\hat{a}_s^{[k-1]}(z)$, where $\hat{a}_s^{[k-1]}(z)$ and $\hat{a}_{ar}^{[k-1]}(z) \triangleq z^{n_a}\hat{a}_a^{[k-1]}(z^{-1})$ are Hurwitz polynomials; the poles of $l^{[k]}(z)$ must therefore either be zeroes of $a(z)$ or of $\hat{a}_{ar}^{[k-1]}(z)\hat{a}_s^{[k-1]}(z)$, and $l^{[k]}(z)$ must be of the form :

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{a(z)\hat{a}_{ar}^{[k-1]}(z)\hat{a}_s^{[k-1]}(z)}$$

where $f^{[k]}(z)$ is a real polynomial with $\deg f^{[k]}(z) = n - 1$.

When the algorithm has converged to a minimum, $\hat{a}^{[k-1]}(z)$ is Hurwitz, since a minimal approximant must be stable; hence, in this case, the 'Hurwitz factor' of $\hat{a}_r^{[k-1]}(z)\hat{a}^{[k-1]}(z)$ is $\hat{a}^{[k-1]}(z)$. Before convergence has been reached no such guarantee can be given; yet at least in a region round the minimum $\hat{a}^{[k-1]}(z)$ will be Hurwitz. In practice, $\hat{a}^{[k-1]}(z)$ is rarely observed to have RHP roots, and if it happens for a certain system $h(z)$, it will only be non-Hurwitz for the first few iterations. Therefore, in order not to make the notation more complicated than necessary, we shall assume that $l^{[k]}(z)$ has the form :

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)} \tag{5.31}$$

even though this is not quite general. Note that this assumption is not necessary, but only adopted for simplicity of notation; while implementing the algorithm one should employ the more general formula, which can be done without any further complications.

Plugging expression (5.31) into (5.29) yields

$$\left[\frac{b(z)\hat{a}^{[k]}(z)}{a(z)}\right]_+ = \left[\frac{\hat{a}^{[k-1]}(z^{-1})f^{[k]}(z)}{a(z)}\right]_+ .$$

This is fulfilled if the residues of both expressions in the stable poles are equal, i.e. if $\hat{a}^{[k-1]}(z^{-1})f^{[k]}(z) - b(z)\hat{a}^{[k]}(z) = 0$ in the roots α_i of $a(z)$.
This gives us n equations in $n+r+1$ unknowns.

Eq. (5.30) may also be expressed as :

$$\left[h(z)l^{[k]}(z^{-1})\right]_+ = \left[l^{[k-1]}(z^{-1})l^{[k]}(z)y(z)\right]_+ + e(z) , \tag{5.32}$$

where $e(z) \in L_2^+$ is such that $[e(z)]_{0\ldots r} = 0$. As both $\left[h(z)l^{[k]}(z^{-1})\right]_+$ and $\left[l^{[k-1]}(z^{-1})l^{[k]}(z)y(z)\right]_+$ are rational and have no other poles than the roots of $a(z)\hat{a}^{[k-2]}(z)$, it is clear that

$$e(z) = \frac{s^{[k]}(z)}{a(z)\hat{a}^{[k-2]}(z)}$$

for some real polynomial $s^{[k]}(z)$ of degree at most $n-2$.
Again (5.32) is equivalent to requiring equality of the residues in the poles α_i and $\hat{\alpha}_i^{[k-2]}$ of both sides of the equation; i.e.

$$\begin{aligned}&\left[\frac{b(z)z^{-r-1}f^{[k]}(z^{-1})\hat{a}^{[k-2]}(z)}{a(z^{-1})\hat{a}^{[k-1]}(z^{-1})} + s^{[k]}(z)\right]_{z=\alpha_i,\hat{\alpha}_i^{[k-2]}} \\ = \quad &\left[\frac{z^{-1}f^{[k-1]}(z)f^{[k-1]}(z^{-1})\hat{a}^{[k-1]}(z^{-1})}{a(z^{-1})\hat{a}^{[k-2]}(z^{-1})}\right]_{z=\alpha_i,\hat{\alpha}_i^{[k-2]}} ,\end{aligned}$$

yielding $n+r$ further equations, but also $n-1$ extra unknowns, i.e. the coefficients of $s^{[k]}(z)$.

Let $\hat{a}^{[k]}(z) \triangleq \sum_{i=0}^{r} \hat{a}_i^{[k]} z^i$, $f^{[k]}(z) \triangleq \sum_{i=0}^{n-1} f_i^{[k]} z^i$, and $s^{[k]}(z) \triangleq \sum_{i=0}^{n-2} s_i^{[k]} z^i$, then define the vectors

$$\hat{a}_k \triangleq \begin{bmatrix} \hat{a}_0^{[k]} \\ \vdots \\ \hat{a}_r^{[k]} \end{bmatrix}, \quad f_k \triangleq \begin{bmatrix} f_0^{[k]} \\ \vdots \\ f_{n-1}^{[k]} \end{bmatrix}, \quad s_k \triangleq \begin{bmatrix} s_0^{[k]} \\ \vdots \\ s_{n-2}^{[k]} \end{bmatrix}.$$

The n linear conditions arising from (5.29) can then be expressed as a matrix equation

$$M_1(\hat{a}_{k-1}) f_k + M_2 \hat{a}_k = 0 ,$$

where $M_1(\hat{a}_{k-1}) \in \mathbb{R}^{n\times n}$ depends only on $\hat{a}_{k-1}$, and $M_2 \in \mathbb{R}^{n\times(r+1)}$ is fixed. Likewise the $n+r$ conditions coming from (5.32) can be written as

$$M_3(\hat{a}_{k-1}, \hat{a}_{k-2}) f_k + M_4(\hat{a}_{k-2}) s_k = F_1(f_{k-1}, \hat{a}_{k-1}, \hat{a}_{k-2}) ,$$

where $M_3(\hat{a}_{k-1}, \hat{a}_{k-2}) \in \mathbb{R}^{(n+r)\times n}$, $M_4(\hat{a}_{k-2}) \in \mathbb{R}^{(n+r)\times(n-1)}$,
and $F_1(f_{k-1}, \hat{a}_{k-1}, \hat{a}_{k-2}) \in \mathbb{R}^{(n+r)\times 1}$.
In total, we have $2n+r$ equations in $2n+r$ unknowns.

The overall algorithm can be summarized as shown in Alg. 5.8.

This algorithm follows rather naturally from the formulas (5.29), (5.30) and (5.31), but it has a drawback. The precise formulas for $M_1(\hat{a}_{k-1})$, M_2, $M_3(\hat{a}_{k-1}$, $M_4(\hat{a}_{k-2})$ and $F_1(f_{k-1}, \hat{a}_{k-1}, \hat{a}_{k-2})$ involve Vandermonde matrices and diagonal matrices; especially the presence of the Vandermonde matrices tends to make the condition number of M very large, greatly impairing the practical relevance of the algorithm given above. A more useful algorithm, which we do not duplicate here, will be given in Chapter 6. Therefore, we shall omit the detailed expressions for $M_1(\hat{a}_{k-1})$, M_2, $M_3(\hat{a}_{k-1}$, $M_4(\hat{a}_{k-2})$ and $F_1(f_{k-1}, \hat{a}_{k-1}, \hat{a}_{k-2})$ here.

Note that with respect to global or local convergence, no proof of either can be given here, any more than for Alg. 5.3, to which it is equivalent.

5.4.4 Structure of the optimum

Upon convergence one has

$$h(z) - \hat{h}(z) \quad = \quad l(z)y(z) = \frac{z^r f^{[k]}(z) z^{-r} \hat{a}_r^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)}$$

Algorithm 5.8

$k = 0$
Initialize $\hat{a}_{-1}$ $\hat{a}_0$, f_0
Compute M_2
do {

$$\begin{aligned} k &\leftarrow k+1 \\ M_1 &\leftarrow M_1(\hat{a}_{k-1}) \\ M_3 &\leftarrow M_3(\hat{a}_{k-1}, \hat{a}_{k-2}) \\ M_4 &\leftarrow M_4(\hat{a}_{k-2}) \\ F_1 &\leftarrow F_1(f_{k-1}, \hat{a}_{k-1}, \hat{a}_{k-2}) \\ M &\leftarrow \begin{bmatrix} M_1 & M_2 & 0 \\ M_3 & 0 & M_4 \end{bmatrix} \\ \textit{Solve}\ M \begin{bmatrix} f_k \\ \hat{a}_k \\ s_k \end{bmatrix} &= \begin{bmatrix} 0 \\ F_1 \end{bmatrix}\ \textit{for}\ f_k,\ \hat{a}_k \\ \begin{bmatrix} \hat{a}_k \\ f_k \end{bmatrix} &\leftarrow \frac{1}{\|\hat{a}_k\|} \begin{bmatrix} \hat{a}_k \\ f_k \end{bmatrix} \end{aligned}$$

} **while** $\|\hat{a}_k - \hat{a}_{k-1}\| > \epsilon$
$\hat{h}(z) = h(z) - \frac{f^{[k]}(z)\hat{a}_r^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)}$

$$= \frac{f(z)}{a(z)} \frac{\hat{a}_r(z)}{\hat{a}(z)} ,$$

where $f(z) \triangleq \lim_{k\to\infty} f^{[k]}(z)$. It is immediately apparent that the modeling error $h(z) - \hat{h}(z)$ contains an all-pass factor $\frac{\hat{a}_r(z)}{\hat{a}(z)}$.

Closer scrutiny actually reveals some more structure : the necessary condition $\left[\hat{h}(z)l(z^{-1})\right]_{0\ldots r}$ can be expressed as

$$\sum_{i=1}^{r} \hat{H}_i l(\hat{\alpha}_i^{-1})\hat{\alpha}_i^{s-1} = 0 , \qquad s = 0, \ldots, r .$$

This can only be satisfied if $l(\hat{\alpha}_i^{-1}) = 0$, $i = 1, \ldots, r$, which means that in a stationary point $l(z)$ has the form

$$l(z) = \frac{z^r \hat{a}_r(z) \bar{f}(z)}{a(z)\hat{a}(z)} \text{ ; hence, } h(z) - \hat{h}(z) = \frac{\bar{f}(z)\hat{a}(z)}{a(z)} \left(\frac{\hat{a}_r(z)}{\hat{a}(z)}\right)^2 ,$$

or in words, the approximation error in the optimum contains an all-pass factor squared. This result, equivalent to the interpolation conditions of [190], may also be found by application of Beurling's theorem [219, 223].

5.4.5 Initialization

In general, the algorithm does not need a close initial guess, but this may reduce the number of iterations needed.

For the time-domain algorithm, with finite p, a natural initial guess for y, x and σ is provided by the singular triplet corresponding to the smallest singular value σ_{r+1} of H, i.e. $H = \sum_{i=1}^{r+1} u_i v_i^* \sigma_i$ is approximated by the rank r matrix $\sum_{i=1}^{r} u_i v_i^* \sigma_i$. For a $p \times p$ Hankel matrix, as $p \to \infty$, this is equivalent to balanced truncation [160, 317, 145]; but for a $\infty \times (r+1)$ Hankel matrix, as in this case, this is not true.

A method perhaps more suitable for the z-domain algorithm is to choose a cheaper suboptimal approximation $\hat{h}_0(z) \triangleq \frac{\hat{b}_0(z)}{\hat{a}_0(z)}$ for starter, and then find $y_0(z)$ and $l_0(z)$ corresponding to this estimate. As the result of the algorithm, aborted after k iterations, is $\hat{h}^{[k]}(z) = h(z) - l^{[k]}(z) y^{[k]}(z)$, $y_0(z)$ and $l_0(z)$ should obey $h(z) - \hat{h}_0(z) = l_0(z) y_0(z)$. This is generally not possible when $\hat{h}_0(z)$ is suboptimal, since then $h(z) - \hat{h}_0(z)$ does not necessarily vanish in the zeros of $y_0(z)$. However, with $y_0(z) = \hat{a}_0(z^{-1})$, one may take $l_0(z) \triangleq \left[(y_0(z))^{-1} (h(z) - \hat{h}_0(z)) \right]_+$; when $\hat{h}_0(z)$ is a good guess, $h(z) - \hat{h}_0(z) = l_0(z) y_0(z)$ will be satisfied approximately.

To obtain $h_0(z)$, one may use e.g. modal reduction, or alternatively Balanced Truncation. Kabamba [151] notes that the Hankel singular values do not always reflect the importance of the contributions of the balanced states to the H_2 norm well : indeed, in a balanced realization

$$\begin{aligned} \|h(z)\|_2^2 &= \text{Tr } (C \Sigma C^T)) = \sum_i \sigma_i C_{(:,i)}^T C_{(:,i)} \\ &= \text{Tr } (B^T \Sigma B)) = \sum_i \sigma_i B_{(i,:)} B_{(i,:)}^T \ . \end{aligned}$$

He proposes a small modification, i.e. to truncate those states x_i corresponding to the smallest quantities $\sigma_i C^T_{(:,i)} C_{(:,i)} = \sigma_i B_{(i,:)} B^T_{(i,:)}$. It is plausible that this will often be better, but this cannot be made quite rigorous; in many cases it will of course yield exactly the same reduced order model. However, examples may easily be found where Kabamba's scheme yields a significantly larger H_2 error; see the example at the end of this chapter.

A combinatorial search over all r-dimensional truncations of the balanced realization, evaluating the H_2 error in each truncated model, is another possibility; because of the cost of an H_2 norm evaluation, and in the light of the fact that our algorithm generally converges fast from any starting point, it is rather doubtful whether such a procedure is justified even when r is relatively small.

5.4.6 Which Algorithm?

The algorithm given above is the translation of Alg. 5.3 to the z-domain; naturally Alg. 5.4 results in the similar procedure Alg. 5.9.

Furthermore, Subsection 5.2.3 gave three ways of speeding up Alg. 5.3, which is the origin of the algorithm above; let's see if they are of any use now.

Recall that among the time domain algorithms, Alg. 5.3 was slightly more costly with respect to the operation count than Alg. 5.5. One could raise the question whether a translation of the latter procedure to the z-domain does also result in an algorithm more efficient with respect to number of operations per iteration. This is indeed the approach taken in [73]. In a similar way as above, rational expressions are derived for the quantities appearing in Alg. 5.5; the only unknown parameters in these expressions appear in the numerators as in the algorithm given above, and linear equations are derived from which these parameters can be found. However, it is not surprising that in this approach, more linear equations have to be solved for more unknowns, as more variables appear in Alg. 5.5 than in Alg. 5.3; in fact, a $(3n + 2r) \times (3n + 2r)$ linear system has to be solved each iteration there, compared to a $(2n + r) \times (2n + r)$ system for the algorithm presented here. Not only does the number of flops needed to solve this system grow proportionally to $(3n + 2r)^3$, but also the condition number of this system is generally observed to grow very fast as it gets larger.

It is evident that further speedup using Toeplitz structure of matrices D_y and D_l as in subsection 5.2.3 is impossible for the z-domain algorithm Alg. 5.8; indeed Alg. 5.8 is exploiting precisely this structure.

The technique of skipping updates of D_y of subsection 5.2.3 is equally possible in the z-domain, and amounts to keeping the matrix M unchanged for a couple of iterations, updating only F_1. Skipping updates of D_l is equivalent to holding f_K fixed in $F_1(f_K, \hat{a}_{k-1}, \hat{a}_{k-2})$ for some iterations.

5.4.7 Continuous time

A continuous time H_2 model reduction problem is readily converted into a discrete-time one. This is true also for the frequency weighted extension of the algorithm above, treated in Chapter 6. As the unweighted problem is but a special instance of the frequency weighted one, we refer to Section 6.2 for an explanation of this conversion.

5.5 Example

Consider the 8th order system $G(s) \triangleq \sum_{i=1}^{4} G_i(s)$, where

$$
\begin{aligned}
G_1(s) &\triangleq \frac{5s}{s^2 + 0.08s + 1} \\
G_2(s) &\triangleq \frac{20s}{s^2 + 0.32s + 16} \\
G_3(s) &\triangleq \frac{40s}{s^2 + 1.28s + 256} \\
G_4(s) &\triangleq \frac{80s}{s^2 + 5.12s + 4096}
\end{aligned}
$$

which is to be reduced to order $r = 6$. The magnitude Bode plot of $G(s)$ is shown by the solid line in Fig. 5.1.

This function is constructed in such a way that it has 4 local minima, of which one is the global one. Evidently $G(s) - G_i(s)$, $i = 1, 2, 3, 4$, are good initial guesses for these local minima. Using a general-purpose quasi-Newton method with the gradients above, locally optimal approximants $G_i^o(s)$ are indeed found, where $G_i^o(s)$ is obtained with $G(s) - G_i(s)$ as an initial guess. Magnitude Bode plots of $G_i^o(s)$ and $G(s) - G_i(s)$ are shown in Fig. 5.1. The H_2 error norms of these approximants, defined $E_i^o \triangleq \|G(s) - G_i^o(s)\|_2$, are $E_1^o = 12.39$, $E_2^o = 24.77$, $E_3^o = 24.81$, and $E_4^o = 24.99$. $G_1^o(s)$ is the global optimum. If Alg. 5.8 is run with $G_i^o(s)$, $i = 1, 2, 3, 4$, as initial guess, not surprisingly it does not diverge from these local optima; however, they seem to have a rather small zone of attraction. Taking as an initial system :

$$
G_0(s) \triangleq \frac{5.1s}{s^2 + 0.1s + 0.99} + \frac{40s}{s^2 + 1.25s + 255} + \frac{79s}{s^2 + 5.1s + 4100} ,
$$

the general-purpose optimization algorithm quite naturally yields $G_2^o(s)$ as optimum, whereas Alg. 5.8 finds $G_1^o(s)$.
In Fig. 5.2, it is shown how the approximant $G_r^{(n)}$ obtained after n steps of the STLS algorithm, starting with initial guess $G_r^{(0)} = G_0(s)$, evolves towards the optimum.

When the denominator of an approximant is fixed in advance, and optimal values are only sought for the numerator coefficients, the latter are easily found as a function of the denominator coefficients. Indeed, in many algorithms this is used to eliminate the numerator coefficients from the problem.
In Fig. 5.2 it can be seen that the numerator of these intermediate approximants is certainly not optimal; the shape of the intermediate functions is often very far off, even though their poles are sometimes very close to (locally) optimal locations. This feature may be regarded as a drawback, but obviously, without this property one could never evolve towards the global optimum when the initial point is so near another local optimum.

We certainly do not claim that the STLS-based algorithms presented above are guaranteed to return the global optimum in every case; in fact counterexamples may be constructed. When the object function values in the local minima are not much higher than in the global one, it is in general harder for the algorithm to 'find' the global minimum. But in many cases, e.g. the one above, it has an advantage over descent algorithms, which are certain to remain stuck in a local minimum once they get close to it.

The convergence of the denominator coefficients $\|\hat{a}_k - \hat{a}\|_2$ and the convergence of the poles $\sqrt{\sum_i(\hat{\alpha}^{[k]_i} - \hat{\alpha}_i)^2}$ are plotted in Fig. 5.3. Observe that the denominator coefficients converge sooner than the poles, illustrating that changes in the pole locations may be a better basis for a stopping criterion than $\|\hat{a}_k - \hat{a}\|_2$. After a while the convergence becomes linear, as is not surprising for a scheme inspired by power iteration : as soon as one is close enough to the optimum, the solution-dependent matrices D_y and D_l change very little, and the scheme behaves almost in the same way as inverse power iteration.
For the reduction done here, the quasi-Newton optimization requires on average 25 times more operations than the STLS algorithm, i.e. 37×10^6 vs. 1.4×10^6 flops. In general, the operation count for STLS compares very favorably to state-space gradient-based H_2 optimization methods.

It might be argued that for a system that consists of so lightly damped modes as in this case, a suboptimal method like Balanced Truncation will do nearly as well, while requiring much less computation. Indeed, the BT model is computed

using 160000 flops or so, almost 10 times less expensive as the STLS-based reduction. However, BT picks the wrong modes : the resulting reduced order model is close to $G_4^o(s)$ rather than $G_1^o(s)$.
The modification of BT proposed by Kabamba is also off the mark : it yields a model G_k with $\|G - G_k\|_2 = 38.2$.

5.6 Conclusions

The widely applicable STLS framework [73] can be used for SISO model estimation from measured impulse response data as well as for SISO H_2 model reduction from exactly known Markov parameters. A family of iterative schemes for solving this type of STLS problems, inspired by inverse power iteration, was treated.

For model reduction, these algorithms are plagued with an annoying tradeoff : perfect accuracy requires infinitely many Markov parameters, and hence also infinitely many associated Lagrange multipliers and an infinite amount of computation. Fortunately, for finite-dimensional systems, the aforementioned algorithms can be translated to the z-domain, i.e. the infinite sequence of Markov parameters and the associated Lagrange parameters may be represented by their rational z-transform. The resulting algorithm is much less expensive, and is not inherently inaccurate, since there is no truncation of the impulse response.

For general MIMO systems, these schemes are not applicable, but a generalization for SIMO and MISO systems is straightforward.

Convergence issues are unresolved; examples suggest that even when the iteration is started with vectors $y(z)$ and $l(z)$ very close to those corresponding to a non-global local optimum, the algorithm will often still converge towards the global optimum.

Algorithm 5.9

$k = 0$
Initialize $\hat{a}_{-1},\ \hat{a}_0,\ f_0$
Compute M_2
do {

$$\begin{array}{rcl}
k & \leftarrow & k+1 \\
M_1 & \leftarrow & M_1(\hat{a}_{k-1}) \\
M_3 & \leftarrow & M_3(\hat{a}_{k-1}, \hat{a}_{k-2}) \\
M_4 & \leftarrow & M_4(\hat{a}_{k-2}) \\
F_1 & \leftarrow & F_1(f_{k-1}, \hat{a}_{k-1}, \hat{a}_{k-2}) \\
M & \leftarrow & \begin{bmatrix} M_1 & M_2 & 0 \\ M_3 & 0 & M_4 \end{bmatrix} \\
\textit{Solve } M \begin{bmatrix} f_k \\ \hat{a}_k \\ s_k \end{bmatrix} & = & \begin{bmatrix} 0 \\ F_1 \end{bmatrix} \textit{ for } f_k,\ \hat{a}_k \\
\gamma & \leftarrow & \sqrt{\left\| \dfrac{f^{[k]}(z)\hat{a}_r^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)} \right\|_2} \\
\alpha & \leftarrow & \dfrac{\left\| \frac{f^{[k]}(z) z^r}{a(z)\hat{a}^{[k-1]}(z)} \right\|_2}{\|\hat{a}_{k-1}\|} \\
f_k & \leftarrow & \alpha\gamma^{-1} f_k \\
\hat{a}_k & \leftarrow & \alpha^{-1}\gamma^{-1}\hat{a}_k \\
\tau_k & = & \left[h(z) \dfrac{f^{[k]}(z^{-1})\hat{a}_r^{[k]}(z^{-1})}{a(z^{-1})\hat{a}^{[k-1]}(z^{-1})} \right]_0
\end{array}$$

} while $\|\hat{a}_k - \hat{a}_{k-1}\| > \epsilon$
$\hat{h}(z) = h(z) - \frac{f^{[k]}(z)\hat{a}_r^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)}$

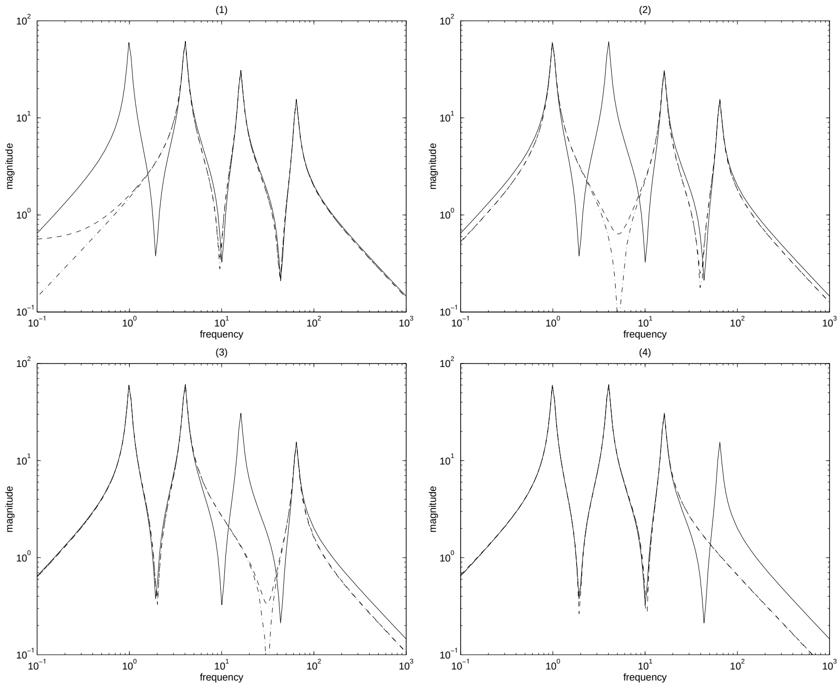


Figure 5.1: Magnitude Bode plots of $G_i^o(s)$ (dashed line) and $G(s) - G_i(s)$ (dash-dotted line) for $i = 1, 2, 3, 4$
Full line: original system.

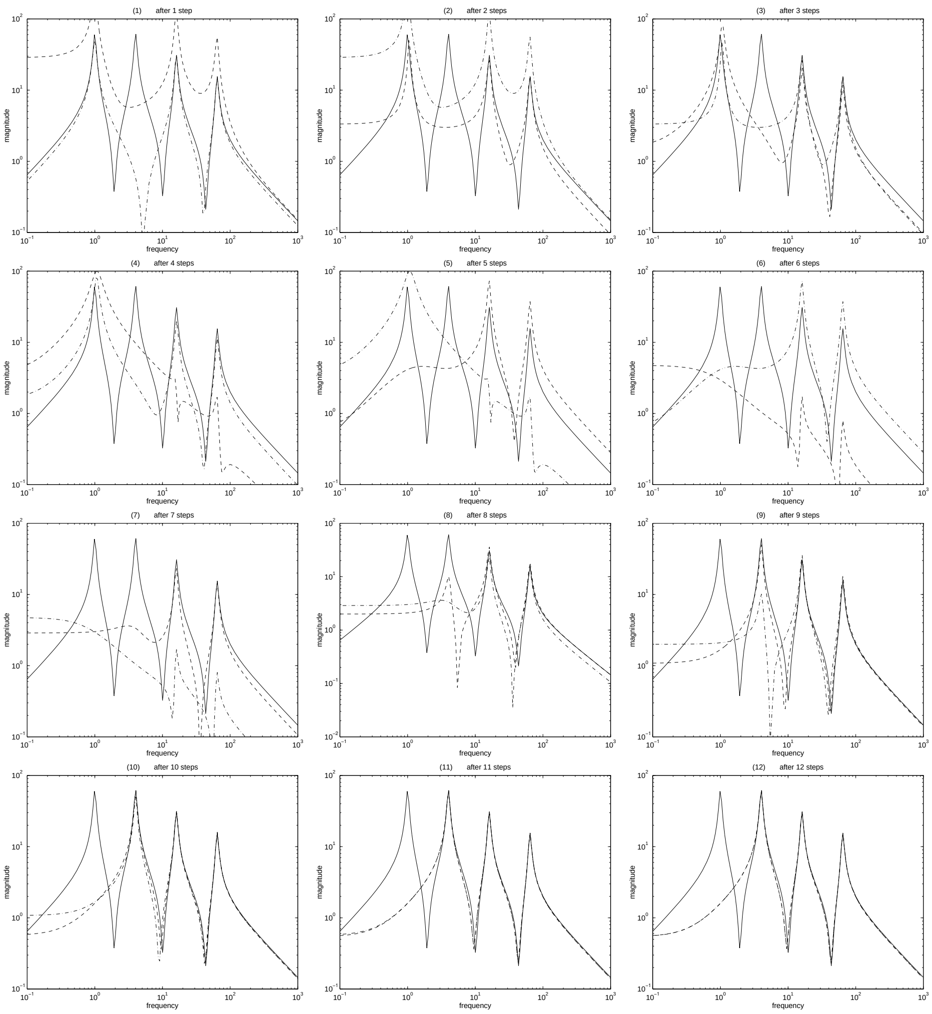


Figure 5.2: Plot (n) contains Bode plot of approximant after n steps of the algorithm (dashed line) and after $n-1$ steps (dash-dotted line).
Full line: original system.

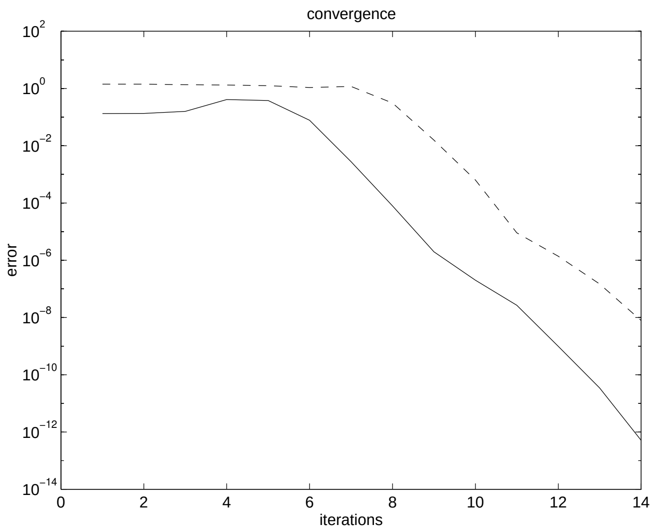


Figure 5.3: Convergence of $\|\hat{a}_k - \hat{a}\|_2$ (full line) and $\|\hat{\alpha}^{[k]} - \hat{\alpha}\|_2$ (dashed line).

Chapter 6

Frequency weighted H_2-optimal model reduction

The problem considered in this chapter is a generalization of the one studied in the previous one, i.e.

$$\min_{\hat{h}(z)} \|w(z)(h(z) - \hat{h}(z))\|_2$$

$$\text{s.t. McMillan } \deg \hat{h}(z) \leq r$$

where $w(z)$ is a given frequency weighting function. The weight $w(z)$ is used to put more emphasis on some frequencies than on others; one should choose $w(z)$ large in frequency intervals where the modeling error should be small and conversely take $w(z)$ small wherever a greater discrepancy can be allowed. Ways of constructing such weighting filters relevant for plant and controller reduction have been given in Section 2.4.

The results in this chapter are also obtained using the STLS framework and the algorithms are a generalization of those of the previous chapter, but the frequency domain methods are dealt with in more detail here. The derivation of necessary conditions and reformulation into 'solution-dependent SVD'-like equations is actually done directly in the z-domain; a treatment starting from the finite horizon time domain formulation analogous to that of the previous chapter is equally possible, but does not enhance clarity. Instead, the analogies will be briefly mentioned in Subsection 6.1.3.

This chapter is organized as follows : in Section 6.1 we treat weighted H_2 reduction with strictly proper approximants, whereas the slightly different case of non-strictly proper approximants is treated in Section 6.3. Section 6.2 gives a trick to convert continuous-time problems to discrete-time ones, which can be solved with the algorithms of Sections 6.1 and 6.3. Some non-generic cases needing special care are touched upon in Section 6.4. In Section 6.5, the frequency weighted algorithm is illustrated with an example.

6.1 Strictly proper systems

In this section, we consider the problem of approximating a discrete time linear time invariant *strictly proper* single-input single-output system of order n by one of order at most r ($r < n$), minimizing the frequency weighted H_2 norm of the modeling error. Recall that a discrete-time system $h(z) \triangleq \sum_{i=0}^{\infty} h_i z^{-i}$ is strictly proper if $h_0 = 0$. For unweighted H_2 approximation, the distinction between the strictly proper and the non-strictly proper case is trivial: as $\hat{h}_0$ has no influence on the order of $\hat{h}(z)$, it is easily seen that for non-strictly proper models $\hat{h}_0$ must equal h_0, and thus the approximation problem reduces to finding the optimal strictly proper reduced-order approximant to the strictly proper part of $h(z)$. For frequency weighted H_2 approximation however, this is no longer true: h_0 and $\hat{h}_0$ need no longer be equal, and finding the optimal approximant over the set of strictly proper systems of order maximally r, or over the set of proper systems of the same order, become two different problems.

It may seem somewhat exaggerated to develop these two cases separately, as these developments must be, conceptually at least, very similar. However, the non-strictly proper case is technically somewhat more complicated, and will therefore be treated in a later subsection; in the remainder of this section the strictly proper case will be expounded.

Again Lagrange multipliers will be used to derive a set of necessary conditions, which are solved iteratively in a way inspired by the inverse power iteration algorithm for singular value decomposition.

Let $h(z)$ be a strictly proper discrete-time stable LTI system of order n and $w(z)$ a stable, causal invertible and minimum-phase weighting function of order w, i.e. $w^{-1}(z) \triangleq (w(z))^{-1}$ is also stable.
We wish to find the system with impulse response $\hat{h}(z) = \sum_{i=1}^{\infty} \hat{h}_i z^{-i}$ such that $J \triangleq \|w(z)(h(z) - \hat{h}(z))\|_2^2 \triangleq \frac{1}{2}\sum_{i=1}^{\infty}([w(z) \star h(z)]_i - [w(z) \star \hat{h}(z)]_i)^2$ is minimized, and $\hat{h}(z)$ has order less or equal to r, where $r < n$. Recall that

$$J = \frac{1}{4\pi j} \oint_T z^{-1} \left(h(z^{-1}) - \hat{h}(z^{-1})\right) w(z^{-1}) w(z) \left(h(z) - \hat{h}(z)\right) dz \; .$$

Furthermore, suppose the systems have the following representations :

$$
\begin{aligned}
h(z) &\triangleq \frac{b(z)}{a(z)} \triangleq \sum_{i=1}^{n} \frac{H_i}{z-\alpha_i} , \\
\hat{h}(z) &\triangleq \frac{\hat{b}(z)}{\hat{a}(z)} \triangleq \sum_{i=1}^{n} \frac{\hat{H}_i}{z-\hat{\alpha}_i} , \\
w(z) &\triangleq \frac{n_w(z)}{d_w(z)} \triangleq \sum_{i=1}^{w} \frac{W_i}{z-\alpha_{wi}} , \\
w^{-1}(z) &\triangleq \frac{d_w(z)}{n_w(z)} \triangleq \sum_{i=1}^{w} \frac{\bar{W}_i}{z-\beta_{wi}} ,
\end{aligned}
$$

where $b(z)$, $a(z)$, $\hat{b}(z)$, $\hat{a}(z)$, $n_w(z)$, $d_w(z)$ are polynomials in z, with $\deg b(z) = n-1$, $\deg a(z) = n$, $\deg \hat{b}(z) = r-1$, $\deg \hat{a}(z) = r$, $\deg n_w(z) = w$, $\deg d_w(z) = w$, and where H_i, $\hat{H}_i$, W_i and $\bar{W}_i$ are residuals.

The requirement that $w(z)$ be minimum phase is not restrictive (except for the fact that it excludes weighting functions with zeroes on the unit circle), since only $w(z^{-1})w(z)$ appears in the optimization criterion, and every stable spectral factor of $w(z^{-1})w(z)$ is an equivalent weight function.
For simplicity, we consider only systems $h(z)$ and $w(z)$ with distinct poles (and zeroes for $w(z)$).

6.1.1 SIMO and MISO weightings

The fact that only the SISO case is dealt with here is restrictive only with respect to the system $h(z)$; taking $w(z)$ to be SISO is not restrictive. Indeed, suppose one wishes to minimize $\|W_o(z)(h(z)-\hat{h}(z))W_i(z)\|_2$ over all $\hat{h}(z)$ of order r or less, where $W_o(z)$ and $W_i(z)$ are $m_o \times 1$ and $1 \times m_i$ systems, respectively. Note that $4\pi jJ$ is equal to the contour integral

$$
\mathrm{Tr} \oint_T z^{-1} W_i^T(z^{-1})(h(z^{-1})-\hat{h}(z^{-1}))W_o^T(z^{-1})W_o(z)(h(z)-\hat{h}(z))W_i(z)dz
$$

$$
= \oint_T z^{-1}(h(z^{-1})-\hat{h}(z^{-1}))W_i(z)W_i^T(z^{-1})W_o^T(z^{-1})W_o(z)(h(z)-\hat{h}(z))dz.
$$

Using spectral factorization, it is possible to find a SISO stable and inverse stable $w(z)$ such that $w(z^{-1})w(z) = W_i(z)W_i^T(z^{-1})W_o^T(z^{-1})W_o(z)$; $w(z)$ is an equivalent SISO frequency weight. Indeed, $w(z)$ is the product of the (SISO) outer factor $w_i(z)$ of $W_i(z)$, which is such that $w_i(z)w_i(z^{-1}) = W_i(z)W_i^T(z^{-1})$, end the outer factor $w_o(z)$ of $W_o(z)$, for which $w_o(z)w_o(z^{-1}) = W_o^T(z^{-1})W_o(z)$.

This remark is important in the light of the fact that although single actuator single sensor (SASS) plants are very common, performance-oriented weights are likely to be multivariable.

6.1.2 Necessary conditions

As in Chapter 5, the constraint on the order of $\hat{h}(z)$ is equivalent to the rank-deficiency of the $\infty \times (r+1)$ Hankel matrix $\hat{H}^\infty$ constructed from the Markov parameters $\hat{h}_i$, which is expressed through the existence of a nonzero null vector, i.e. $\exists y\ : \|y\| = 1$ and $\hat{H}^\infty y = 0$. Using Lagrange multipliers, we may formulate (using somewhat informal notation) the constrained minimization problem as

$$\min_{\hat{h},y,l,\lambda} \frac{1}{2}\sum_{i=1}^{\infty}([w(z) \star h(z)]_i - [w(z) \star \hat{h}(z)]_i)^2 + l^T \hat{H}^\infty y + \lambda(y^T y - 1) \ .$$

With $y(z) \triangleq \sum_{i=0}^{r} y_i z^{-i}$ and $l(z) \triangleq \sum_{i=1}^{\infty} l_i z^{-i}$, taking the derivatives of the Lagrangian with respect to $\hat{h}, l, y$ and λ yields the following necessary conditions (see appendix A):

$$\left[w(z^{-1})w(z)(h(z) - \hat{h}(z))\right]_+ = l(z)y(z) \ ,$$

$$\left[\hat{h}(z)y(z^{-1})\right]_+ = 0 \ , \quad \left[\hat{h}(z)l(z^{-1})\right]_{0\ldots r} = 0 \ , \quad \|y\| = 1 \ .$$

We find the first condition to be equivalent to :

$$h(z) - \hat{h}(z) = w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ \tag{6.1}$$

using the following

Lemma 6.1

$$\text{If} \ \ x(z) \in L_2^+, \ w(z), \ w^{-1}(z) \in RL_2^{0+} \ ,$$

$$\text{Then} \ \left[w^{-1}(z^{-1})\left[w(z^{-1})x(z)\right]_+\right]_+ = x(z) \ , \qquad |z| > 1 \ .$$

Proof : see App. A.2. □

By convolution of (6.1) with $y(z^{-1})$ and $l(z^{-1})$ respectively, and taking the causal part, $\hat{h}(z)$ (and $\hat{H}$) is eliminated from the above necessary conditions:

$$\begin{aligned} \left[h(z)y(z^{-1})\right]_+ &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ y(z^{-1})\right]_+ \ , \\ \left[h(z)l(z^{-1})\right]_{0\ldots r} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ l(z^{-1})\right]_{0\ldots r} . \end{aligned} \tag{6.2}$$

Note that the first right-hand side is linear in y and the second equation is linear in l. If these equations can be solved for $y(z)$ and $l(z)$, $\hat{h}(z)$ can be recomputed from $y(z)$ and $l(z)$ using Eq. (6.1).

Note that these equations give only necessary conditions for stationary points; the number of solutions may grow combinatorially with the problem size. Therefore at least some heuristics will be necessary to pick a good candidate for the global minimum.

6.1.3 Time domain interpretation

In Chapter 5, we started by minimizing a quadratic error over a *finite* horizon $p+r$, where the rank constraint was replaced by the requirement that a *finite* Hankel matrix be rank-deficient. It was shown that this finite-horizon problem could be solved by finding a 'nonlinear' singular value triplet, and then the infinite horizon H_2 model reduction problem of interest, i.e. the limit case of this problem as $p \to \infty$, was translated into the z-domain. In this chapter, the infinite-horizon problem is immediately tackled, and it is a valid question whether the necessary equations (6.2) arrived at here can be viewed as a limit case of a similar finite-horizon 'nonlinear' or 'solution-dependent' S.V.D. problem.

For this purpose we first consider a modified problem, where the rank deficiency of a *finite* Hankel matrix is demanded; furthermore, only *autoregressive* or *all-pole* weighting functions $w(z)$ are allowed, i.e. let $w(z)$ be a stable weighting function of the form $w(z) \triangleq (w_0 + w_1 z^{-1} + \ldots + w_q z^{-q})^{-1}$, or $w^{-1}(z) \triangleq \sum_{i=1}^{\infty} w_i z^{-i}$, with the convention that $w_i \triangleq 0$, $i > q$. Note that $w(z)$ is inherently minimum-phase. Given a stable system with impulse response $h(z)$, the object is now to find the approximating system with impulse response $\hat{h}(z) = \sum_{i=1}^{p+r+q} \hat{h}_i z^{-i}$, such that the quadratic criterion $\frac{1}{2}\sum_{i=1}^{\infty}[w(z)(h(z)-\hat{h}(z))]_i^2)$ is minimized, subject to the condition that the $p \times (r+1)$ Hankel matrix constructed with its first $p+r$ Markov parameters be rank deficient, i.e.

$$\exists y \in \mathbb{R}^{(r+1)\times 1} \neq 0 \;\text{ s.t. }\; \hat{H}y \triangleq \begin{bmatrix} \hat{h}_1 & \hat{h}_2 & \ldots & \hat{h}_{r+1} \\ \hat{h}_2 & \hat{h}_3 & \ldots & \hat{h}_{r+2} \\ \vdots & & & \vdots \\ \hat{h}_p & \ldots & \ldots & \hat{h}_{r+p} \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_r \end{bmatrix} = 0 \, .$$

Again using the same Lagrange multiplier approach, one finds the conditions:

$$\left[w(z^{-1})w(z)(h(z)-\hat{h}(z))\right]_i = [l(z)y(z)]_i \, , \qquad i = 1, \ldots, p+r+q \, ,$$

where this time $l(z)$ has finite length p, i.e. $l(z) \triangleq \sum_{i=1}^{p} l_i z^{-i}$.
Rewriting this as $\left[w(z^{-1})w(z)(h(z)-\hat{h}(z))\right]_+ = [l(z)y(z)]_+ + z^{-p-q-r}e_1(z)$, where $e_1(z) \in L_2^+$, and taking into account that $w^{-1}(z)$ is a FIR filter of length q, so that $z^{-q}w^{-1}(z^{-1})$ is causal, Lemma 6.1 (see Appendix A.2) can again be used to find :

$$
\begin{aligned}
h(z) - \hat{h}(z) &= w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ && (6.3)\\
&\quad + z^{-p-r}w^{-1}(z)e_1(z)(z^{-q}w^{-1}(z^{-1})) && (6.4)\\
&= w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ + z^{-p-r}e_2(z)\ , && (6.5)
\end{aligned}
$$

$$\text{where} \qquad e_2(z) \triangleq w^{-1}(z)e_1(z)(z^{-q}w^{-1}(z^{-1})) \in L_2^+\ .$$

Taking derivatives of the Lagrangian with respect to l_i and y_i yields $\left[\hat{h}(z)y(z^{-1})\right]_i = 0$, $i = 1,\ldots,p$, and $\left[\hat{h}(z)l(z^{-1})\right]_i = 0$, $i = 0,\ldots,r$, as before, and combining these expressions with equation (6.5) one eliminates $\hat{h}(z)$ from the equations :

$$\left[h(z)y(z^{-1})\right]_i = \left[y(z^{-1})w^{-1}(z)\left[w^{-1}(z^{-1})y(z)l(z)\right]_+\right]_i, \qquad i = 1,\ldots,p\ ,$$

$$\left[h(z)l(z^{-1})\right]_i = \left[l(z^{-1})w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+\right]_i, \qquad i = 0,\ldots,r\ .$$

In order to rewrite this finite number of equations in terms of Toeplitz matrices, we introduce

$$
T_w \triangleq \begin{bmatrix}
w_0 & w_1 & \ldots & & w_{p+r-1}\\
0 & w_0 & w_1 & \ldots & \vdots\\
\vdots & \ddots & \ddots & \ddots & \\
0 & \ldots & 0 & w_0 & w_1\\
0 & \ldots & & 0 & w_0
\end{bmatrix},
$$

with $T_w \in \mathbb{R}^{(r+p)\times(r+p)}$. Recall that $T_y \triangleq T(y,p) \in \mathbb{R}^{p\times(r+p)}$, and $T_l \triangleq T(l,r+1) \in \mathbb{R}^{(r+1)\times(r+p)}$.
Then the necessary conditions are equivalent to the condition $y^T y = 1$ and the matrix equations :

$$
\begin{aligned}
Hy &= D_y l \qquad \text{where} \qquad D_y \triangleq T_y T_w^T T_w T_y^T\ ,\\
H^T l &= D_l y \qquad \text{where} \qquad D_l \triangleq T_l T_w^T T_w T_l^T\ .
\end{aligned}
$$

D_y and D_l are positive definite symmetric matrices depending quadratically on the elements of y and l. By scaling the variable $l = x\sigma$ such that $\|x\| = 1$, these equations are again put in the form

$$\begin{aligned} Hy &= D_y x\sigma \qquad \text{where} \qquad D_y \triangleq T_y T_w^T T_w T_y^T \ , \\ H^T x &= D_x y\sigma \qquad \text{where} \qquad D_x \triangleq T_x T_w^T T_w T_x^T \ . \end{aligned}$$

(T_x being defined exactly as T_l, but using x instead of l), which is exactly the singular value decomposition of H except for the presence of the matrices D_y and D_x.

As in the previous chapter, this 'solution-dependent S.V.D.' may have many more solutions than an ordinary S.V.D.; obviously, the solution corresponding to the global minimum of the optimization criterion is what one is after. To our knowledge, no algorithm can guarantee to find the globally optimal H_2 approximant except through enumeration of all solutions, whose number may grow exponentially with the order n of $h(z)$. The algorithm proposed here is not capable of this either. Yet, in our formulation, the frequency weighted approximation error may be expressed as $\|T_w^{-T}(h - \hat{h})\| = \|T_w(x \star y)\|\sigma$, where $\|x\| = \|y\| = 1$ and T_w is fixed. This indicates that a solution with a small 'singular value' σ will correspond to a small approximation error and vice versa, which makes it natural to look for the solution with the smallest σ.

6.1.4 Back to the z-domain

With the modified definitions for D_y and D_l, Alg. 5.3 may still be used as a scheme for solving the necessary conditions above. This iteration can again be expressed in the z-domain :

$$\begin{aligned} &\left[h(z)y^{[k]}(z^{-1})\right]_{1,\dots,p} \\ &\quad = \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_+ y^{[k-1]}(z^{-1})\right]_{1,\dots,p} \ , \qquad (6.6) \\ &\left[h(z)l^{[k]}(z^{-1})\right]_{0\dots r} \\ &\quad = \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ l^{[k-1]}(z^{-1})\right]_{0\dots r} \ . \qquad (6.7) \end{aligned}$$

Executing this iteration for ever larger p, $l^{[k]}(z)$ is still found to converge to a rational function, i.e. $\lim_{p\to\infty} l^{[k]}(z)$ has a finite number of poles, and indeed these poles are found to be equal to the poles of $h(z)$ and $w(z)$, and the roots of $y^{[k-1]}(z^{-1})$, and therefore $l^{[k]}(z)$ may be represented in a more compact manner than by its Markov parameters l_i. Indeed, if one can prove this statement, the denominator of $l^{[k]}(z)$ represented as a fraction of polynomials is known, and equations (6.6)(6.7) may be expressed in terms of the parameters defining the numerator. If these translated equations can be solved, they will yield a reduced model without loss of accuracy through truncation after time p. In Chapter 5 (and in [73]), this is done for the unweighted problem, using the coefficients of the numerator polynomial as unknowns. There, equations (6.6)(6.7) translate into a set of $2n+r$ $(3n+2r)$ linear equations in as many unknowns. While this offers a considerable improvement over the time domain method, this is still a sizeable linear system to be solved each iteration. The size of the linear system can be further reduced, as will be shown below. For this purpose, we introduce some definitions :

$\hat{a}^{[k]}(z) \triangleq y^{[k]}(z^{-1})$ (i.e. a polynomial in z of degree r)
$\alpha_i \triangleq$ roots of $a(z)$ (= poles of $h(z)$)
$\hat{\alpha}_i^{[k]} \triangleq$ roots of $\hat{a}^{[k]}(z)$
$\alpha_{wi} \triangleq$ roots of $d_w(z)$ (= poles of $w(z)$)
$\{a_i^{[k]}\} \triangleq \{\alpha_i, \hat{\alpha}_i^{[k]}, \alpha_{wi}\}$
$\beta_{wi} \triangleq$ roots of $n_w(z)$ (= zeroes of $w(z)$)

$$\xi^{[k]}(z) \triangleq \left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_+$$

6.1.5 Properties of $l^{[k]}(z)$ and $\xi^{[k]}(z)$

In a way analogous to Chapter 5, rational expressions can be found for $l^{[k]}(z)$ and $\xi^{[k]}(z)$. To give a more formal proof this time, we employ the following

Lemma 6.2

$$\left.\begin{array}{l} x(z) \in L_2^+ \\ n_s(z),\ \ d_s(z)\ \textit{Hurwitz polynomials} \\ n_a(z),\ \ d_a(z)\ \textit{anti-Hurwitz polynomials} \\ \left[\frac{n_s(z)n_a(z)}{d_s(z)d_a(z)}x(z)\right]_+ = \frac{y(z)}{a(z)} \in L_2^+ \\ n_s(z)a(z)\ \textit{and}\ d_s(z)\ \textit{have no common roots.} \end{array}\right\} \Rightarrow x(z) = \frac{d_s(z)f(z)}{n_s(z)a(z)}$$

where $f(z)$ *is a polynomial with* $\deg f(z) < \deg n_s(z) + \deg a(z) - \deg d_s(z)$

Proof : see App. A.2. □

With this lemma, and using

$$\begin{aligned}\left[h(z)y^{[k]}(z^{-1})\right]_+ = \left[\frac{b(z)}{a(z)}\hat{a}^{[k]}(z)\right]_+ &= \left[w^{-1}(z)\xi^{[k]}(z)y^{[k-1]}(z^{-1})\right]_+ , \\ &= \left[\frac{d_w(z)}{n_w(z)}\xi^{[k]}(z)\hat{a}^{[k-1]}(z)\right]_+ ,\end{aligned}$$

we find

$$\xi^{[k]}(z) = \frac{n_w(z)\phi^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)d_w(z)}, \quad \phi^{[k]}(z) \text{ a polynomial of degree } n+r-1 \,, \tag{6.8}$$

since $\xi^{[k]}(z)$ is causal. Again, as in Chapter 5, p. 157, the assumption is made that $\hat{a}^{[k-1]}(z)$ is Hurwitz, which might occasionally be violated during the first few iterations; whenever this does not hold, the same fix may be applied.

From the definition of $\xi^{[k]}(z)$:

$$\xi^{[k]}(z) \triangleq \left[w^{-1}(z^{-1})l^{[k]}(z)z^{-r}\hat{a}_r^{[k-1]}(z)\right]_+ , \tag{6.9}$$

again using Lemma 6.2, we find an expression for $l^{[k]}(z)$:

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)d_w(z)}, f^{[k]}(z) \text{ a polynomial of degree } n+w-1 \,,$$

as $l^{[k]}(z)$ must cancel the r-multiple pole at $z=0$ and have the same poles as $\xi^{[k]}(z)$.

One could generalize the 'polynomial' approach of [73] and Chapter 5 to the case with frequency weighting $w(z)$. In that case, one expresses equation (6.7) by requiring that $\left[z^{-1}w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ l^{[k-1]}(z^{-1})\right]_+$ and $[z^{-1}h(z)l^{[k]}(z^{-1})]_+$ need only be equal in the first $r+1$ Markov parameters, so the difference between both can be expressed as $\frac{s^{[k]}(z)}{a(z)\hat{a}^{[k-2]}(z)n_w(z)}$, where $s^{[k]}(z)$ is a polynomial in z of degree $n+w-2$. The unknowns are then the coefficients of $f^{[k]}(z)$, $\phi^{[k]}(z)$, $\hat{a}^{[k]}(z)$ and $s^{[k]}(z)$, amounting to $(n+w)+(n+r)+(r+1)+(n+w-1) = 3n+2r+2w$ unknowns. The equations come from interpolation of equation (6.6) in the roots of $a(z)$ and interpolation of equation (6.7) and equation (6.9) in the roots of $a(z)$, $\hat{a}^{[k-2]}(z)$ and $n_w(z)$, resulting in $n+(n+r+w)+(n+r+w)$ equalities, i.e. a linear system of size $3n+2r+2w$ is to be solved each iteration. This approach works, but the rapidly growing size of the linear system makes it rather slow; moreover the linear system becomes ill-conditioned even for moderate values of n, r and w, e.g. for random systems with $n = 9, r = 4, w = 0$, the condition number will generally be in the range $O(10^{12}) - O(10^{15})$.

We will develop an approach that results in a smaller number of unknowns and linear equations, and a better condition number.

6.1.6 Partial fraction representation

We have shown that $l^{[k]}(z)$ can be written as:

$$l^{[k]}(z) \triangleq \sum_{i=1}^{n} \frac{C_{\alpha i}^{[k]}}{z-\alpha_i} + \sum_{i=1}^{r} \frac{C_{\hat{\alpha} i}^{[k]}}{z-\hat{\alpha}_i^{[k-1]}} + \sum_{i=1}^{w} \frac{C_{wi}^{[k]}}{z-\alpha_{wi}} \triangleq \sum_i \frac{C_i^{[k]}}{z-a_i^{[k-1]}}$$

The following lemma is helpful for determining the residues $C_i^{[k]}$:

Lemma 6.3

$$\textit{If } X(z) \in RL_2^{+},\ Y(z) \in RL_2^{0+}\ , \textit{with } X(z) \triangleq \sum_{i=1}^{n_x} \frac{X_i}{z-\alpha_i}$$

$$\textit{Then } \left[X(z)Y(z^{-1})\right]_+ = \sum_i \frac{X_i Y(\alpha_i^{-1})}{z-\alpha_i}$$

Proof : follows immediately from Lemma A.1, to be found in the appendix.

By this 'interpolation in the poles', we find:

$$\begin{aligned}\xi^{[k]}(z) &\triangleq \sum_i \frac{X_i^{[k]}}{z-a_i^{[k-1]}} = \left[w^{-1}(z^{-1}) l^{[k]}(z) y^{[k-1]}(z)\right]_+ \\ &= \sum_{i=1}^{n+r+w} \frac{w^{-1}((a_i^{[k-1]})^{-1})\hat{a}^{[k-1]}((a_i^{[k-1]})^{-1})C_i^{[k]}}{z-a_i^{[k-1]}}\ ; \qquad (6.10)\end{aligned}$$

hence $\quad X_i^{[k]} = w^{-1}((a_i^{[k-1]})^{-1})\hat{a}^{[k-1]}((a_i^{[k-1]})^{-1})C_i^{[k]}\ ;$

$$\text{and} \qquad C_i^{[k]} = \gamma_i^{[k-1]} X_i^{[k]}\ , \quad \text{with} \quad \gamma_i^{[k]} \triangleq \frac{w((a_i^{[k]})^{-1})}{\hat{a}^{[k]}((a_i^{[k]})^{-1})}\ . \qquad (6.11)$$

The equality $\left[h(z)y^{[k]}(z^{-1})\right]_+ = \left[w^{-1}(z)\hat{a}^{[k-1]}(z)\xi^{[k]}(z)\right]_+$ yields the following interpolation conditions in the poles α_i :

$$\hat{a}^{[k]}(\alpha_i) = H_i^{-1} w^{-1}(\alpha_i)\hat{a}^{[k-1]}(\alpha_i^{-1})X_i^{[k]} \triangleq D_i^{[k-1]} X_i^{[k]}\ . \qquad (6.12)$$

Therefore, $V_\alpha^T \hat{a}^{[k]} = D^{[k-1]} X^{[k]}$, defining $D^{[k-1]}$ to be a diagonal matrix with $[D^{[k-1]}]_{(i,i)} \triangleq D_i^{[k-1]}$ and where V_α consists of the first $r+1$ rows of the Vandermonde matrix with second row $\begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_n \end{bmatrix}$, i.e.

$$V_\alpha \triangleq \begin{bmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \dots & \alpha_n^2 \\ \vdots & & & \vdots \\ \alpha_1^r & \alpha_2^r & \dots & \alpha_n^r \end{bmatrix} .$$

If V_α^T has QR decomposition $V_\alpha^T = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix}$, then

$$R^{-1} Q_1^H D^{[k-1]} X_\alpha^{[k]} = \hat{a}^{[k]} , \tag{6.13}$$
$$Q_2^H D^{[k-1]} X_\alpha^{[k]} = 0 . \tag{6.14}$$

Equation (6.14) gives $n-r-1$ conditions on $X_i^{[k]}$.

Because $l^{[k]}(z)$ has an r-multiple zero at $z=0$, $l^{[k]}(z)$ and its first $r-1$ derivatives vanish at $z=0$:

$$\sum_{i=1}^{n+r+w} \frac{C_i^{[k]}}{(a_i^{[k-1]})^s} = \sum_{i=1}^{n+r+w} \frac{\gamma_i^{[k-1]} X_i^{[k]}}{(a_i^{[k-1]})^s} = 0 ,$$

$$\text{i.e.} \quad P_\alpha^{[k-1]} X_\alpha^{[k]} + P_{\hat{\alpha}}^{[k-1]} X_{\hat{\alpha}}^{[k]} + P_w^{[k-1]} X_w^{[k]} = P^{[k-1]} X^{[k]} = 0 , \tag{6.15}$$

where $P_{ij}^{[k]} \triangleq \gamma_i^{[k]}((a_i^{[k]})^j)^{-1}$. These are r conditions on $X_i^{[k]}$.

Since $\xi^{[k]}(z) = \frac{n_w(z)\phi^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)d_w(z)}$, $\xi^{[k]}(z)$ vanishes at $z=\beta_{wi}$, $i=1,\dots,w$.

Substitution into (6.10) renders :
$$\sum_{j=1}^{n+r+w} \frac{X_j^{[k]}}{\beta_{wi} - a_j^{[k-1]}} = 0 \qquad i=1,\dots,w ; \tag{6.16}$$

these w linear conditions on $X_i^{[k]}$ can be rewritten as a matrix equation :

$$G_\alpha^{[k-1]} X_\alpha^{[k]} + G_{\hat{\alpha}}^{[k-1]} X_{\hat{\alpha}}^{[k]} + G_w^{[k-1]} X_w^{[k]} = G^{[k-1]} X^{[k]} = 0 , \tag{6.17}$$

where $G_{ij}^{[k]} \triangleq (\beta_{wi} - a_j^{[k]})^{-1}$.

Finally (6.7) provides the missing $r+1$ conditions :
$\left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} = \left[\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ w^{-1}(z)l^{[k-1]}(z^{-1})\right]_{0\ldots r}$,

where
$$\begin{aligned}\left[h(z)l^{[k]}(z^{-1})\right]_s &= \left[\sum_i \frac{H_i}{z-\alpha_i} l^{[k]}(z^{-1})\right]_s \\ &= \sum_i H_i \frac{l^{[k]}(\alpha_i^{-1})}{\alpha_i}\left[\frac{1}{1-\alpha_i z^{-1}}\right]_s \\ &= \sum_i H_i l^{[k]}(\alpha_i^{-1})\alpha_i^{s-1}\end{aligned}$$

and
$$\left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ l^{[k-1]}(z^{-1})\right]_+ = \sum_i \frac{R_{\beta i}}{z-\beta_{wi}} + \sum_i \frac{R_{\alpha i}}{z-\alpha_i} + \sum_i \frac{R_{\hat{\alpha} i}}{z-\hat{\alpha}_i^{[k-2]}} \; .$$

Again, interpolation in the poles β_{wi}, α_i and $\hat{\alpha}_i$ gives (see Lemma 6.3) :

$$\begin{aligned}R_{\beta i} &\triangleq l^{[k-1]}(\beta_{wi}^{-1})\sum_j \frac{w^{-1}((a_j^{[k-2]})^{-1}).\hat{a}^{[k-1]}((a_j^{[k-2]})^{-1}).C_j^{[k-1]}}{\beta_{wi}^{-1}-a_j^{[k-2]}} W_i \\ R_{\alpha i} &\triangleq w^{-1}(\alpha_i)w^{-1}(\alpha_i^{-1}).l^{[k-1]}(\alpha_i^{-1}).\hat{a}^{[k-1]}(\alpha_i^{-1}).C_{\alpha i}^{[k-1]} \\ R_{\hat{\alpha} i} &\triangleq w^{-1}(\hat{\alpha}_i^{[k-2]}).w^{-1}((\hat{\alpha}_i^{[k-2]})^{-1}).l^{[k-1]}((\hat{\alpha}_i^{[k-2]})^{-1}).\hat{a}^{[k-1]}((\hat{\alpha}_i^{[k-2]})^{-1}).C_{\hat{\alpha} i}^{[k-1]}\end{aligned}$$

So we find for $s = 0, \ldots, r$:

$$\sum_i H_i \alpha_i^s \sum_j \frac{\gamma_j^{[k-1]} X_j^{[k]}}{1-a_j^{[k]}\alpha_i} = \sum_i R_{\beta i}\beta_{wi}^{s-1} + \sum_i R_{\alpha i}\alpha_i^{s-1} + \sum_i R_{\hat{\alpha} i}(\hat{\alpha}_i^{[k-2]})^{s-1} \triangleq V_s^{[k-1]} \; ,$$

or as a matrix equation :

$$J_\alpha^{[k-1]} X_\alpha^{[k]} + J_{\hat{\alpha}}^{[k-1]} X_{\hat{\alpha}}^{[k]} + J_w^{[k-1]} X_w^{[k]} = J^{[k-1]} X^{[k]} = V^{[k-1]} \; , \tag{6.18}$$

$$\text{where } [J^{[k]}]_{(s,j)} \triangleq \sum_i H_i \alpha_i^s \gamma_j^{[k]} (1-a_j^{[k]}\alpha_i)^{-1} \; .$$

This gives $r+1$ conditions on $X_i^{[k]}$.

Summing it all up, we find equations (6.14)(6.15)(6.17)(6.18) yield $n+r+w$ linear conditions on $n+r+w$ unknowns $X_i^{[k]}$, i.e. the following $(n+r+w)$-by-$(n+r+w)$ linear system with complex coefficients has to be solved each iteration :

$$\begin{bmatrix} Q_2^H D^{[k-1]} & 0 & 0 \\ P_\alpha^{[k-1]} & P_{\hat{\alpha}}^{[k-1]} & P_w^{[k-1]} \\ G_\alpha^{[k-1]} & G_{\hat{\alpha}}^{[k-1]} & G_w^{[k-1]} \\ J_\alpha^{[k-1]} & J_{\hat{\alpha}}^{[k-1]} & J_w^{[k-1]} \end{bmatrix} \begin{bmatrix} X_\alpha^{[k]} \\ X_{\hat{\alpha}}^{[k]} \\ X_w^{[k]} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ V^{[k-1]} \end{bmatrix} . \tag{6.19}$$

The solution contains only real numbers and complex conjugate pairs.

The condition number of this system is much better than in the polynomial method mentioned on p. 180 : for a random system with $n=9, r=4, w=0$, the condition number will be in the range $O(10^4)-O(10^7)$. Moreover, the smaller size of the linear system to be solved each iteration results in a dramatic speed improvement over this method. The computational complexity of one iteration is generally in the range of $O(20(n+r+w)^3)$ to $O(25(n+r+w)^3)$.

6.1.7 Computation of $\hat{h}(z)$

Upon convergence, the best approximant $\hat{h}(z)$ is retrieved using eq. (6.1) :

$$\hat{h}(z) = \sum_{i=1}^{n} \frac{\hat{H}_i}{z-\hat{\alpha}_i} = h(z) - w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ = h(z) - w^{-1}(z)\xi(z) \ ,$$

Interpolation in the poles (Lemma 6.3) again yields an expression for the residues :

$$\hat{H}_i = -w^{-1}(\hat{\alpha}_i)X_i \ . \tag{6.20}$$

6.1.8 Initialization

It is not necessary to provide a close initial guess, but naturally, it reduces the number of iterations needed. For the initial $\hat{h}_0(z)$ one can take the model produced by any model reduction method; frequency weighted balanced truncation à la Enns [93], is a suitable candidate. This immediately yields the initial $y_0(z) = \hat{a}_0(z^{-1})$.
As one has :

$$h(z) - \hat{h}^{[k]}(z) = w^{-1}(z)\xi^{[k]}(z) \ ,$$

the initial $\xi_0(z)$ corresponding to $\hat{h}_0(z)$ is $\xi_0(z) \triangleq w(z)(h(z)-\hat{h}_0(z))$; the initial $l(z)$ is computed as $l_0(z) \triangleq \left[y_0^{-1}(z)w(z^{-1})\xi_0(z)\right]_+$.

6.1.9 Algorithm

Algorithm 6.1

$k = 0$
Compute an initial approximation $\hat{h}_0(z) = \frac{\hat{b}_0(z)}{\hat{a}_0(z)}$
Initialize $y(z)$, $\xi(z)$ *and* $l(z)$ *as in paragraph 6.1.8.*
do {

1. $k \leftarrow k + 1$
2. *Solve the linear system (6.19) to find* $l^{[k]}(z)$
3. *Find* $\hat{a}^{[k]}$ *from (6.13) to find* $y^{[k]}(z)$

} while $\|\hat{a}^{[k]} - \hat{a}^{[k-1]}\| > \epsilon$
Compute $\hat{h}(z)$ *using (6.20)*

Remark that other stopping criteria may be used as well, e.g. one may stop when the decrease of the object function $\|h(z) - \hat{h}(z)\| = \|w^{-1}(z)\xi(z)\|$, which can be computed cheaply using Lemma A.3 (see Appendix A.2), or the quantity $\|H^T l_{k-1} - D_{l_{k-1}} y_{k-1}\|_2 = \|H^T(l_{k-1} - l_k)\|_2 = \|h(z)(l^{[k-1]}(z) - l^{[k]}(z))\|_2$ get smaller than a certain tolerance.

As convergence issues are unresolved in the unweighted case, *a fortiori* global convergence cannot be proven here either. Again no counterexamples are known. In rare cases, convergence will be very slow and sometimes the linear convergence sets in only after an apparently chaotic phase.

6.2 Continuous time H_2 model reduction

One could think that as the CT H_2 norm only makes sense for strictly proper systems, and the bilinear transform maps CT strictly proper systems into DT *non*strictly proper systems, a translation of the CT H_2 model reduction problem into a DT problem will result in a frequency weighted reduction problem of a non-strictly proper model. But we cannot simply use the bilinear transformation on the CT model $h(s)$ to map the problem into a DT one.

Recall that some norms, such as the Hankel norm and the H_∞ norm, are invariant under the bilinear transform $s \leftarrow \frac{z-1}{z+1}$; hence, as the bilinear transform does not change the order of a system either, optimal model reduction with respect to those norms for continuous-time systems and for discrete-time systems is equivalent : a CT system $H_c(s)$ is approximated optimally by $\mathcal{B}_T^{-1}(\hat{H}_d)$, where $\hat{H}_d$ is the optimal approximant of order r to $H_d(z) \stackrel{\triangle}{=} \mathcal{B}_T(H_c)$. But for H_2 model reduction, this does not work : the bilinear transform does not preserve the H_2 norm.

Fortunately, frequency weighted H_2 model reduction problems for continuous time systems can easily be converted into discrete time equivalents. Let $H_c(s)$ be a continuous-time strictly proper stable system of order n, and $W_c(s)$ be a continuous-time, stable, causal invertible, minimum phase weight of order w,

$$\begin{aligned} \text{then } \|W_c(s)H_c(s)\|_2^2 &= \frac{1}{2\pi}\int_{-\infty}^{\infty} |W_c(j\omega)H_c(j\omega)|^2 d\omega \\ &= \frac{1}{2\pi j}\int \text{Tr } (H_c^T(-s)W_c^T(-s)W_c(s)H_c(s))ds \ . \end{aligned}$$

The last integral is taken over the imaginary axis $s = j\omega$, ω going from $-\infty$ to ∞. Substituting $s \leftarrow \frac{z-1}{z+1}$, one has $ds = \frac{2dz}{(z+1)^2}$, and defining $H(z) \stackrel{\triangle}{=} H_c(\frac{z-1}{z+1})$, $H_1(z) \stackrel{\triangle}{=} \frac{1}{z+1}H_c(\frac{z-1}{z+1})$ and $W(z) \stackrel{\triangle}{=} W_c(\frac{z-1}{z+1})$, one has :

$$\begin{aligned} &\|W_c(s)H_c(s)\|_2^2 \\ &\quad = \frac{1}{2\pi j}\oint_T \frac{2z^{-1}}{(z+1)(z^{-1}+1)}\text{Tr } (H^T(z^{-1})W^T(z^{-1})W(z)H(z))dz \\ &\quad = \frac{1}{2\pi j}\oint_T 2z^{-1}\text{Tr } (H_1^T(z^{-1})W^T(z^{-1})W(z)H_1(z))dz \ . \end{aligned}$$

Remark that that $H(z)$ has a transmission zero at $z = 1$, and that $H_1(z)$ is strictly proper and has order n, since the additional pole at $z = 1$ is cancelled by the transmission zero at $z = 1$. Now find the system $\hat{H}_1(z)$ of order at most r that minimizes $\|W(z)(H_1(z) - \hat{H}_1(z))\|_2$. Then the best approximant of order at most r for the continuous-time system is found to be $\hat{H}_c(s) \stackrel{\triangle}{=} (1+z)\hat{H}_1(z) = \frac{2}{1-s}\hat{H}_1(\frac{1+s}{1-s})$ (note that $\hat{H}_c(s)$ has the same order as $\hat{H}_1(z)$, because $\hat{H}_1(\frac{1+s}{1-s})$ has a zero at $s = 1$).
Indeed, suppose there is a better approximant $\hat{G}_c(s)$ of $H_c(s)$ in the CT frequency weighted H_2 norm, then $(1+z)^{-1}\hat{G}_c(\frac{z-1}{z+1})$ is a better approximant of $H(z)$ in the DT frequency weighted H_2 norm.

Note furthermore that if $H_c(s) = [A, B, C, 0]$, then a state-space representation for $H_1(z)$ is

$$H_1(z) = \left[\begin{array}{c|c} (I+A)(I-A)^{-1} & B \\ \hline C\,(I-A)^{-1} & 0 \end{array}\right] ,$$

and if $\hat{H}_1(z) = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & 0 \end{array}\right]$, then $\hat{H}_c(s) = 2\left[\begin{array}{c|c} (\hat{A}-I)(\hat{A}+I)^{-1} & \hat{B} \\ \hline \hat{C}(I+\hat{A})^{-1} & 0 \end{array}\right]$.

6.3 Non-strictly proper systems

Consider the problem

$$\min_{\hat{h}(z)} \|w(z)(h(z)-\hat{h}(z))\|_2$$

with $w(z)$ satisfying the same conditions as before and where $\hat{h}(z) \triangleq \sum_{i=0}^{\infty} \hat{h}_i z^{-i}$ is stable. The only difference with Section 6.1 is that there is one extra variable, i.e. $\hat{h}_0$, with respect to which the optimization needs to be done; in other words, we consider optimization over the set of proper approximants, in contrast with Section 6.1, where we optimized over all *strictly* proper approximants. Note that this distinction was unimportant for the unweighted H_2 approximation of Chapter 5; in that case, the optimal $\hat{h}_0$ equals h_0.
The parameter $\hat{h}_0$ has no influence on the order of $\hat{h}(z)$, i.e. the order constraint can be formulated as

$$\left[\left[\hat{h}(z)\right]_+ y(z^{-1})\right]_+ \equiv \left[\hat{h}(z)y(z^{-1})\right]_+ = 0 \; .$$

Using the same machinery as in the previous section, the necessary conditions are easily found to be :

$$\left[w(z^{-1})w(z)(h(z)-\hat{h}(z))\right]_{0+} = l(z)y(z) \; ,$$

$$\left[\hat{h}(z)y(z^{-1})\right]_+ = 0 \; , \qquad \left[\hat{h}(z)l(z^{-1})\right]_{0\ldots r} = 0 \; , \qquad \|y\| = 1 \; ,$$

where now $h(z)$ and $\hat{h}(z) \in RL_2^{0+}$, whereas $l(z)$ is still in RL_2^{+}.

We need a lemma slightly different from Lemma 6.1, with similar proof :

Lemma 6.4

$$\text{If } x(z) \in RL_2^{0+},\; w(z),\; w^{-1}(z) \in RL_2^{0+}$$

$$\text{Then } \left[w^{-1}(z^{-1})\left[w(z^{-1})x(z)\right]_{0+}\right]_{0+} = x(z),\; |z| > 1$$

Therefore

$$h(z) - \hat{h}(z) = w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_{0+} . \tag{6.21}$$

and $\hat{h}(z)$ can be eliminated as before :

$$\begin{aligned}\left[h(z)y(z^{-1})\right]_+ &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_{0+} y(z^{-1})\right]_+ , \\ \left[h(z)l(z^{-1})\right]_{0\ldots r} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l(z)y(z)\right]_{0+} l(z^{-1})\right]_{0\ldots r} . \end{aligned} \tag{6.22}$$

The iteration now becomes :

$$\begin{aligned}\left[h(z)y^{[k]}(z^{-1})\right]_+ &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_{0+} y^{[k-1]}(z^{-1})\right]_+ , \\ \left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} &= \left[w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_{0+} l^{[k-1]}(z^{-1})\right]_{0\ldots r} .\end{aligned}$$

or, introducing $\xi^{[k]}(z) \triangleq \left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_{0+}$, this translates into

$$\xi^{[k]}(z) = \left[w^{-1}(z^{-1})l^{[k]}(z)\hat{a}^{[k-1]}(z^{-1})\right]_{0+} \tag{6.23}$$

$$\left[h(z)\hat{a}^{[k]}(z)\right]_+ = \left[w^{-1}(z)\xi^{[k]}(z)\hat{a}^{[k-1]}(z)\right]_+ \tag{6.24}$$

$$\left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} \tag{6.25}$$

$$= \left[l^{[k-1]}(z^{-1})w^{-1}(z)\left[w^{-1}(z^{-1})l^{[k-1]}(z)\hat{a}^{[k-1]}(z^{-1})\right]_{0+}\right]_{0\ldots r} \tag{6.26}$$

From (6.24) one finds :

$$\xi^{[k]}(z) = w(z)\frac{\phi^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)} , \tag{6.27}$$

where $\phi^{[k]}(z)$ is a polynomial of degree $n + r$, and from (6.23) :

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)d_w(z)} , \tag{6.28}$$

where $f^{[k]}(z)$ is a polynomial of degree $n + w - 1$.
The poles of $\xi^{[k]}(z)$ and $l^{[k]}(z)$ are therefore contained in the set $\{a_i^{[k-1]}\} \triangleq \{\alpha_i\} \cup \{\hat{\alpha}_i\} \cup \{\alpha_{wi}\}$. We shall write a_i for $a_i^{[k-1]}$ to simplify the notation.

Again one may express these quantities in pole-residue form :

$$\begin{aligned}\xi^{[k]}(z) &= X_0^k + \sum_{i=1}^{n+r+w} \frac{X_i^k}{z-a_i}\\ l^{[k]}(z) &= \sum_{i=1}^{n+r+w} \frac{C_i^k}{z-a_i}\end{aligned}$$

Clearly an extra unknown X_0^k is introduced here.

As in Subsection 6.1.6, one must still express the knowledge about the zeroes :

$$\xi^{[k]}(\beta_{wi}) = 0 \qquad i = 1, \ldots, w \tag{6.29}$$

$$\frac{d^s}{dz^s}\left[l^{[k]}(\beta_{wi})\right]_0 = 0 \qquad s = 0, \ldots, r-1 \tag{6.30}$$

At this point, some lemmas are presented to be used in the sequel.
Define the polynomial $p(z) \triangleq \sum_i^{n_p} p_i z^i$, and $X(z), Y(z) \in RL_2^{0+}$,

with $\quad X(z) \triangleq X_0 + \sum_{i=1}^{n_x} \frac{X_i}{z-\alpha_i}, \qquad Y(z) \triangleq Y_0 + \sum_{j=1}^{n_y} \frac{Y_j}{z-\beta_j}.$

Then we can state :

Lemma 6.5

$$[X(z)Y(z)]_{0+} = X_0Y_0 + \sum_i \frac{X_iY(\alpha_i)}{z-\alpha_i} + \sum_j \frac{Y_jX(\beta_j)}{z-\beta_j}$$

Proof : obtained by ordinary decomposition into partial fractions.

Lemma 6.6

$$\begin{aligned}\left[X(z)Y(z^{-1})\right]_{0+} &= X_0Y_0 + \sum_i\sum_j \frac{X_iY_j}{1-\alpha_i\beta_j} + \sum_i \frac{X_iY(\alpha_i^{-1})}{z-\alpha_i}\\ &= Y_0X(0) + \sum_i Y(\alpha_i^{-1})\alpha_i^{-1}X_i + \sum_i \frac{Y(\alpha_i^{-1})X_i}{z-\alpha_i}\\ &= Y_0X(0) + \sum_i \alpha_i^{-1}Y(\alpha_i^{-1})X_i\left[\frac{1}{1-\alpha_i z^{-1}}\right]_{0+}\end{aligned}$$

Proof : by Lemma 6.3 : $\left[X(z)Y(z^{-1})\right]_+ = \sum_i \frac{X_i Y(\alpha_i^{-1})}{z-\alpha_i}$,
so only $\left[X(z)Y(z^{-1})\right]_0 = X_0Y_0 + \sum_i \sum_j \frac{X_iY_j}{1-\alpha_i\beta_j}$ needs to be proven.

$$\text{Since} \quad X(z) = X_o + \sum_i X_i z^{-1}(1 + \alpha_i z^{-1} + \alpha_i^2 z^{-2} + \ldots)$$
$$\text{and} \quad Y(z^{-1}) = Y_o + \sum_j Y_i z(1 + \beta_j z + \beta_j^2 z^2 + \ldots) ,$$

$$\begin{aligned} \text{we have } \left[X(z)Y(z^{-1})\right]_0 &= X_0Y_0 + \sum_i \sum_j X_iY_j(1 + \alpha_i\beta_j + \alpha_i^2\beta_j^2 + \ldots) \\ &= X_0Y_0 + \sum_i \sum_j \frac{X_iY_j}{1-\alpha_i\beta_j} \\ &= X_0Y_0 + \sum_i X_i(Y(\alpha_i^{-1})\alpha_i^{-1} - Y_0\alpha_i^{-1}) \\ &= Y_0X(0) + \sum_i Y(\alpha_i^{-1})\alpha_i^{-1}X_i \ . \end{aligned}$$ □

Lemma 6.7

$$[p(z)X(z)]_{0+} = p(0)X(0) + \sum_i p(\alpha_i)\alpha_i^{-1}X_i + \sum_i \frac{p(\alpha_i)X_i}{z-\alpha_i}$$

Proof : this is just a particular instance of Lemma 6.6.

$$\text{If} \quad h(z) \stackrel{\triangle}{=} H_0 + \sum_i \frac{H_i}{z-\alpha_i} \quad \text{and} \quad w^{-1}(z) \stackrel{\triangle}{=} \bar{W}_0 + \sum_i \frac{\bar{W}_i}{z-\beta_{wi}} ,$$

$$\text{then, by Lemma 6.7 :} \qquad \left[h(z)\hat{a}^{[k]}(z)\right]_+ = \sum_i \frac{\hat{a}^{[k]}(\alpha_i)H_i}{z-\alpha_i}$$

and by Lemmas 6.5 and 6.7 :

$$\left[\hat{a}^{[k-1]}(z)w^{-1}(z)\xi^{[k]}(z)\right]_+ = \sum_i \frac{\hat{a}^{[k-1]}(\alpha_i)w^{-1}(\alpha_i)X_{\alpha i}^k}{z-\alpha_i}$$

and hence $\hat{a}^{[k]}(\alpha_i)H_i = \hat{a}^{[k-1]}(\alpha_i)w^{-1}(\alpha_i)X_{\alpha i}^k$, giving n conditions on the unknowns X_i^k and $\hat{a}_i^{[k]}$.

$$\text{By lemma 6.5 :} \qquad \hat{a}^{[k-1]}(z^{-1})l^{[k]}(z) = \sum_i \frac{\hat{a}^{[k-1]}(a_i^{-1})C_i^k}{z-a_i} \ .$$

Note that terms in z^{-1} do not appear in this expression, since $l^{[k]}(z)$ has a zero with multiplicity r at $z=0$. Using Lemma 6.6, one finds :

$$\begin{aligned}&\Big[w^{-1}(z^{-1})\hat{a}^{[k-1]}(z^{-1})l^{[k]}(z)\Big]_{0+}\\ &\qquad = \sum_{ij}\frac{\bar{W}_j\hat{a}^{[k-1]}(a_i^{-1})C_i^k}{1-a_i\beta_{wj}}+\sum_i\frac{w^{-1}(a_i^{-1})\hat{a}^{[k-1]}(a_i^{-1})C_i^k}{z-a_i}\ ,\end{aligned}$$

$$\begin{aligned}\text{hence}\ \ X_i^k &= w^{-1}(a_i^{-1})\hat{a}^{[k-1]}(a_i^{-1})C_i^k\ ,\\ X_0^k &= \sum_i\Big(\sum_j\frac{\bar{W}_j}{1-a_i\beta_{wj}}\Big)w(a_i^{-1})X_i^k=\sum_i a_i^{-1}(1-\bar{W}_0w(a_i^{-1}))X_i^k\ .\end{aligned}$$

The former relation is identical to (6.11), and is used to eliminate C_i^k. The latter equation may be used to get rid of the new unknown X_0^k; in fact, the only place where X_0^k appears is in condition (6.29), which can thus be reformulated as :

$$\sum_{i=1}^{n+r+w}\Big(a_i^{-1}(1-\bar{W}_0w(a_i^{-1}))+\frac{1}{\beta_{wj}-a_i}\Big)X_i^{[k]}=0 \qquad i=1,\ldots,w\ . \tag{6.31}$$

The only expression that has not been used is (6.26); by Lemma 6.6,

$$\Big[h(z)l^{[k]}(z^{-1})\Big]_{0+}=\sum_{ij}\frac{H_iC_j^k}{1-\alpha_ia_j}+\sum_i\sum_j\frac{C_j^k}{\alpha_i^{-1}-a_j}\frac{H_i}{z-\alpha_i}\ ,$$

and therefore

$$\Big[h(z)l^{[k]}(z^{-1})\Big]_s=\sum_i H_i\alpha_i^s\sum_j\frac{w(a_j^{-1})(\hat{a}^{[k-1]}(a_j^{-1}))^{-1}X_j^k}{1-\alpha_ia_j}\ .$$

For the calculation of the right hand side of equation (6.26), the following definitions are introduced :

$$\begin{aligned}R^{[k-1]}(z) &\triangleq \Big[w^{-1}(z^{-1})\hat{a}^{[k-1]}(z^{-1})l^{[k-1]}(z)\Big]_{0+}\\ Y^{[k-1]}(z) &\triangleq w^{-1}(z)R^{[k-1]}(z)\\ V^{[k-1]}(z) &\triangleq \Big[l^{[k-1]}(z^{-1})Y^{[k-1]}(z)\Big]_{0+}\\ \bar{a}_i &\triangleq a_i^{[k-2]}\end{aligned}$$

It is straightforward to find, using Lemmas 6.5 and 6.6, that

$$\begin{aligned} R^{[k-1]}(z) &= R_0^{k-1} + \sum_i \frac{R_i^{k-1}}{z-\bar{a}_i} \\ &= \sum_{ij} \frac{\bar{W}_j \hat{a}^{[k-1]}(\bar{a}_i^{-1}) C_i^{k-1}}{1-\bar{a}_i \beta_{wj}} + \sum_i \frac{w^{-1}(\bar{a}_i^{-1}) \hat{a}^{[k-1]}(\bar{a}_i^{-1}) C_i^{k-1}}{z-\bar{a}_i} \\ &= \sum_i \bar{a}_i^{-1} (w^{-1}(\bar{a}_i^{-1}) - \bar{W}_0) \hat{a}^{[k-1]}(\bar{a}_i^{-1}) C_i^{k-1} \\ &\qquad\qquad + \sum_i \frac{w^{-1}(\bar{a}_i^{-1}) \hat{a}^{[k-1]}(\bar{a}_i^{-1}) C_i^{k-1}}{z-\bar{a}_i} . \end{aligned}$$

By Lemma 6.5, and with obvious definitions for Y_i and $\bar{b}_i$,

$$\begin{aligned} Y^{[k-1]}(z) &\triangleq Y_0^{k-1} + \sum_{i=1}^{n+r+w} \frac{Y_i^{k-1}}{z-\bar{b}_i} = \bar{W}_0 R_0^{k-1} \\ &+ \sum_{i=1}^{n} \frac{w^{-1}(\alpha_i) R_{\alpha i}^{k-1}}{z-\alpha_i} + \sum_{i=1}^{r} \frac{w^{-1}(\hat{\alpha}_i^{k-2}) R_{\hat{\alpha} i}^{k-1}}{z-\hat{\alpha}_i^{k-2}} + \sum_{i=1}^{w} \frac{\bar{W}_i R^{[k-1]}(\beta_{wi})}{z-\beta_{wi}} , \end{aligned}$$

and finally, by Lemma 6.6,

$$\begin{aligned} V^{[k-1]}(z) &\triangleq V_0^{k-1} + \sum_{i=1}^{n+r+w} \frac{V_i^{k-1}}{z-\bar{b}_i} \\ &= \sum_i l^{[k-1]}(\bar{b}_i^{-1}) \bar{b}_i^{-1} Y_i^{k-1} + \sum_i \frac{l^{[k-1]}(\bar{b}_i^{-1}) Y_i^{k-1}}{z-\bar{b}_i} \\ &= \sum_i \bar{b}_i^{-1} l^{[k-1]}(\bar{b}_i^{-1}) Y_i^{k-1} \left[\frac{1}{1-\bar{b}_i z^{-1}} \right]_{0+} . \end{aligned}$$

Hence $\left[h(z) l^{[k]}(z^{-1})\right]_s = \left[V^{[k-1]}(z)\right]_s$, for $s = 0, \ldots, r$, yields the following $r+1$ linear equations in X_i^k:

$$\sum_{i=1}^{n} H_i \alpha_i^s \sum_{j=1}^{n+r+w} \frac{w(a_j^{-1}) (\hat{a}^{[k-1]}(a_j^{-1}))^{-1} X_j^k}{1-\alpha_i a_j} = \sum_{i=1}^{n+r+w} V_i^{k-1} \bar{b}_i^{s-1} , \quad s = 0, \ldots, r .$$

Note that this is identical to Eq. (6.18), only the computation of the residues V_i^{k-1} is different.

Summing up, $n+r+w$ linear equations in $n+r+w$ unknowns have to be solved each iteration, as in Subsection 6.1.6; equations (6.13), (6.14), (6.15) and the left hand side of Eq. (6.18) are unchanged, whereas (6.17) is replaced by (6.31), and the right hand side of Eq. (6.18) is computed as shown above.

6.4 Some special cases

The algorithms presented above have the disadvantage that they break down in some special cases, e.g. when poles or zeroes have a multiplicity greater than one or when poles of $h(z)$ coincide with zeroes of $w(z)$. Many of these conditions occur exactly for the weightings proposed in Section 2.4. One way out is approximation of the weight $w(z)$ by a lower order function; this will in general move the poles and zeroes sufficiently to be able to use the original algorithm. Below, it is shown that often, knowledge of coinciding poles or zeroes may be exploited, sometimes yielding smaller linear systems to be solved within each iteration.

6.4.1 Cancelation of poles and zeroes of the weighting.

The attentive reader will have noticed that when some of the poles of the system to be reduced $h(z)$ coincide with transmission zeroes of the weighting filter, the developments given above do not carry through. One might try to waive this objection by giving the standard lame excuse, i.e. that this situation is *non-generic*, and therefore can never occur in practice and should not be an object of study in an engineering thesis. Calamitously however, it occurs more often than not, as the reader will see when referring to Section 2.4; this is e.g. the case with the weighting function $W_s = (I + GC)^{-1}G$ for robust stability when replacing a controller C for a plant G by a reduced-order controller C_r. Fortunately, this failing is easily mended, and it turns out that it amounts to a simplification rather than a complication, as will be shown in the sequel of this section.

Suppose $w(z)$ has c zeroes that are also poles of $h(z)$, i.e. $n_w(z) = n(z)n_{w1}(z)$ and $a(z) = n(z)a_1(z)$, where $n(z)$ is a Hurwitz polynomial of degree c; suppose furthermore that α_i, $i = 1, \ldots, c$, are zeroes of $w(z)$, and that β_{wi}, $i = 1, \ldots, c$, are poles of $h(z)$.
In this case, it is readily found that

$$\xi^{[k]}(z) = \frac{n_{w1}(z)\phi^{[k]}(z)}{a_1(z)\hat{a}^{[k-1]}(z)d_w(z)} , \qquad \phi^{[k]}(z) \text{ a polynomial of degree } n + r - 1 ,$$

and hence

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{a_1(z)\hat{a}^{[k-1]}(z)d_w(z)} , \qquad f^{[k]}(z) \text{ a polynomial of degree } n+w-c-1 ;$$

i.e. the c common zeroes do not appear as poles of $\xi^{[k]}(z)$ and $l^{[k]}(z)$, and the number of unknowns decreases by c. The pole-residue expressions of $\xi^{[k]}(z)$ and

$l^{[k]}(z)$ are of the same form as before, but with fewer terms :

$$\begin{aligned} \xi^{[k]}(z) &\triangleq \sum_i \frac{X_i^{[k]}}{z - a_i^{[k-1]}} \\ l^{[k]}(z) &\triangleq \sum_i \frac{C_i^{[k]}}{z - a_i^{[k-1]}} \end{aligned}$$

where $\{a_i^{[k-1]}, i = 1, \ldots, n - c + r + w\}$
$= \{\alpha_i, i = c + 1, \ldots, n\} \cup \{\hat{\alpha}_i^{[k-1]}, i = 1, \ldots, r\} \cup \{\alpha_{wi}, i = 1, \ldots, w\}$.

On the other hand, $\xi^{[k]}(\beta_{wi}) = 0$ is not valid anymore in the roots of $n(z)$, therefore one loses c conditions; equation 6.17 changes into

$$\begin{bmatrix} 0 & I_{w-c} \end{bmatrix} G^{[k-1]} X^{[k]} = 0 \; .$$

Furthermore condition (6.12), i.e. $\hat{a}^{[k]}(\alpha_i) = H_i^{-1} w^{-1}(\alpha_i) \hat{a}^{[k-1]}(\alpha_i) X_i^{[k]}$, is now only valid in the $n - c$ poles of $h(z)$ that are not in common with $w^{-1}(z)$, and should be replaced with $\hat{a}^{[k]}(\alpha_i) = H_i^{-1} \bar{W}_i \hat{a}^{[k-1]}(\alpha_i) \xi^{[k]}(\alpha_i)$ in the c common poles. Again these can be expressed in matrix form $V_\alpha^T \hat{a}^{[k]} = \Delta^{[k-1]} X^{[k]}$,

$$\text{where} \quad \Delta^{[k-1]} \triangleq \begin{bmatrix} \begin{bmatrix} 0 & I_{n-c} \end{bmatrix} D^{[k-1]} \\ E^{[k-1]} \end{bmatrix}, \text{ with } \; E_{ij}^{[k-1]} \triangleq \frac{H_i \bar{W}_i \hat{a}^{[k-1]}(\alpha_i)}{\alpha_i - a_j} \; .$$

However, $\Delta^{[k-1]}$ is no longer a diagonal matrix like $D^{[k-1]}$ in (6.13) and (6.14). As before, $R^{-1} Q_1^* \Delta^{[k-1]} X_\alpha^{[k]} = \hat{a}^{[k]}$ can be used to eliminate $\hat{a}^{[k]}$;
$Q_2^* \Delta^{[k-1]} X_\alpha^{[k]} = 0$ are the $n - r - 1$ unexploited conditions.

The conditions (6.15) and the calculation of $V^{[k-1]}(z)$ remain unchanged, keeping in mind that $l^{[k-1]}(z)$ has only $n - c + r + w$ terms now, and has no poles in common with $w^{-1}(z)$.

Thus each iteration, the following linear system of $(n-r-1)+(w-c)+r+(r+1) = n + r + w - c$ equations in $n + r + w - c$ unknowns is to be solved :

$$\begin{bmatrix} Q_2^H \Delta^{[k-1]} & 0 & 0 \\ P_\alpha^{[k-1]} & P_{\hat{\alpha}}^{[k-1]} & P_w^{[k-1]} \\ \begin{bmatrix} 0 & I_{w-c} \end{bmatrix} G_\alpha & \begin{bmatrix} 0 & I_{w-c} \end{bmatrix} G_{\hat{\alpha}}^{[k-1]} & \begin{bmatrix} 0 & I_{w-c} \end{bmatrix} G_w \\ J_\alpha^{[k-1]} & J_{\hat{\alpha}}^{[k-1]} & J_w^{[k-1]} \end{bmatrix} X^{[k]} = \bar{V}^{[k-1]} \; , \tag{6.32}$$

$$\text{with } X^{[k]} \triangleq \begin{bmatrix} X_\alpha^{[k]} \\ X_{\hat{\alpha}}^{[k]} \\ X_w^{[k]} \end{bmatrix} \quad \text{and} \quad \bar{V}^{[k]} \triangleq \begin{bmatrix} 0 \\ 0 \\ 0 \\ V^{[k]} \end{bmatrix} .$$

Furthermore, $\hat{a}^{[k]} = R^{-1} Q_1^H \Delta^{[k-1]} X_\alpha^{[k]}$.

6.4.2 H_2-optimal relative error model reduction.

The cancelation case of the previous section can be worked out for non-strictly proper systems as in Section 6.3 as well. We shall omit this treatment in the general case, and instead deal with only one specific case, i.e. the H_2-optimal relative error model reduction problem.

Let $h(z)$ be a stable and causal invertible SISO system. In H_2 relative error model reduction, we wish to optimize $\| (h_o(z))^{-1} \left(h(z) - \hat{h}(z)\right) \|_2$ where $h_o(z)$ is an outer spectral factor of $h(z)$, i.e. $h_o(z)$ is stable and minimum phase, and $h(z)h(z^{-1}) = h_o(z)h_o(z^{-1})$.
If $h(z) = \frac{b_s(z)b_a(z)}{a(z)}$, where $b_s(z)$ and $a(z)$ are Hurwitz polynomials of degree $n-m$ and n, resp., and $b_a(z)$ is an anti-Hurwitz polynomial of degree m, then $w(z) = \frac{a(z)}{b_s(z)b_{ar}(z)}$, with $b_{ar}(z) \triangleq z^m b_a(z^{-1})$.

It is seen that this is an extreme case of the situation of Subsection 6.4.1 : *all* the poles of $h(z)$ coincide with the zeroes of $w(z)$.

The expressions for $\xi^{[k]}(z)$ and $l^{[k]}(z)$ specialize to :

$$\xi^{[k]}(z) = \frac{\phi^{[k]}(z)}{\hat{a}^{[k-1]}(z)b_s(z)b_{ar}(z)} = \sum_i \frac{X_i^k}{z - a_i^{k-1}} + X_0^k ,$$

where $\deg \phi^{[k]}(z) = n + r$ and

$$l^{[k]}(z) = \frac{z^r f^{[k]}(z)}{\hat{a}^{[k-1]}(z)b_s(z)b_{ar}(z)} = \sum_i \frac{C_i^k}{z - a_i^{k-1}} ,$$

where $\deg f^{[k]}(z) = n - 1$. The only differences with Section 6.3 are that there are w unknowns less, and that the w conditions (6.29) or (6.31) do not hold anymore. As before, X_0^k may be eliminated, and the expressions for $R^{[k-1]}(z)$, $Y^{[k-1]}(z)$ and $V^{[k-1]}(z)$ remain unchanged, except for the fact that the terms in $(z - \alpha_i)^{-1}$ are zero in $Y^{[k-1]}(z)$ and $V^{[k-1]}(z)$ and thus do not need to be computed.

6.4.3 Weighting functions with poles or zeros on the unit circle

If $w(z)$ has marginally stable poles or zeroes (i.e. poles or zeroes on the unit circle), it is not clear whether it should be considered as a stable and inverse stable weighting function. This is also a non-generic but significant case : when mapping a CT problem with a strictly proper weight $w_c(s)$ into a DT problem, one obtains a discrete weight $w(z)$ with a zero at $z = -1$; similarly, zeroes at $s = 0$ are mapped into $z = 1$.

By replacing the marginally stable zeroes z_m and poles p_m with ρz_m and ρp_m, where $\rho < 1$, and taking the limit $\rho \to 1$, one can see that no changes are required for this case.

6.4.4 Strictly proper weighting functions

A strictly proper weighting function $w(z)$ can be interpreted as a weighting function with one or more transmission zeroes on $z = \infty$, i.e. a non-minimum phase weighting function; the minimum phase equivalent is found by replacing those zeroes by zeroes on $z = 0$. In other words, a weighting equivalent to

$$w_{sp}(z) = \frac{n_w(z)}{d_w(z)} \ , \text{ with } d \triangleq \deg d_w(z) - \deg n_w(z) > 0 \ , \text{ is } w(z) = \frac{z^d n_w(z)}{d_w(z)} \ .$$

6.4.5 Multiple poles.

Again, it can be argued that the presence of a multiple pole is a non-generic condition. Of course, while this is true, one should ask oneself the question how the algorithm will behave whenever one is close to such a non-generic condition, i.e. when $|p_i - p_j|$ is small. Indeed breakdown through numerical problems or limit cycling may occur when two poles nearly coincide. How much distance there has to be between the poles to prevent this is hard to predict : no problems have been experienced for systems with poles more than 10^{-4} apart, while other systems with two poles at a distance 10^{-7} of each other do not give trouble. In practice, one is not very likely to encounter this kind of breakdown problems because of almost coinciding poles.

Unfortunately, as in Subsection 6.4.1, this non-generic condition is precisely a feature of the weighting function constructed as in Subsection 6.1.1 from the performance-oriented weighting functions of Subsection 2.4.2.

Multiple poles are not intractable, but merely more complicated. It is not elaborated here. Using Lemma A.1, formulas may be derived for this case. Section 7.3

in Chapter 7 will give a flavor of how the presence of multiple poles makes the derivation more tedious and involved : there, a series $l^{[k]}(z)$ will appear, of which the poles all have multiplicity two.

6.4.6 Poles at $z = 0$.

Another special case is occurring when $h(z)$ has a (possibly multiple) pole at $z = 0$. This is the case e.g. when $h(z)$ contains a 'pure delay', i.e. when $h(z)$ can be factored as $z^{-d}h_1(z)$, or when $h(z)$ is a FIR filter. This can be classified among the cases dealt with in the previous subsection, but it has one extra effect : it influences the form of $l^{[k]}(z)$: some or all of the r zeroes of $l^{[k]}(z)$ will be canceled by these poles.

6.4.7 Weighting functions with multiple zeroes.

A non-generic case similar to the one above occurs when one or more transmission zeroes of $w(z)$ have multiplicity greater than one, but it is somewhat easier : it only affects Eq. (6.16) and the computation of $R_{\beta i}$.
If the multiplicity of β_{wi} is m_i, one now has

$$\frac{d^{s-1}}{dz^{s-1}}(\xi^{[k]}(z))\bigg|_{\beta_{wi}} = 0 \ , s = 1, \ldots, m_i \ ,$$

yielding :

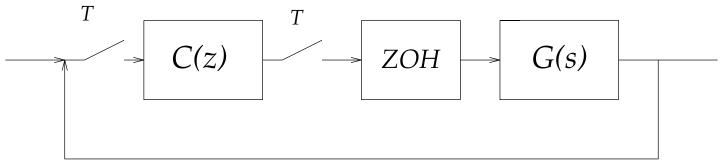
$$\sum_{j=1}^{n+r+w} \frac{X_j^{[k]}}{(\beta_{wi} - a_j^{[k-1]})^s} = 0 \qquad s = 1, \ldots, m_i, \ i = 1, \ldots, w \ .$$

6.4.8 Coinciding or multiple poles in the approximant.

One of the assumptions that are prerequisite for the above algorithms is also that no roots of $\hat{a}^{[k]}(z)$ should coincide with zeroes or poles of $w(z)$, or with poles of $h(z)$, or should have multiplicity greater than one.
To deal with these special cases is more cumbersome, since it cannot be determined in advance whether they occur or not, and each iteration one should test for it; one needs to fix tolerances for these tests, etc.
It seems somewhat of an overkill to do this, especially in the case that this non-generic condition only occurs in one of the iterates $\hat{a}^{[k]}(z)$, but not in the optimum $\hat{a}(z)$; it is advisable then to restart from another initial point. If the non-generic condition *does* occur in the optimum though, the iterates $\hat{a}^{[k]}(z)$ will come ever closer to this too, and one may have premature breakdown of the iteration, whatever the starting point.

6.5 Example

Consider the following sampled-data control system, taken from a Matlab Robust Control Toolbox demo :



$$\text{Actuator transfer function } G(s) = \frac{1000}{s^3 + 40s^2 + 800s + 1000}$$

With a sampling period of $T = 0.01$ s, one finds that the discrete-time system describing the zero order hold and the plant $G(s)$ is

$$G(z) = 10^{-4}\frac{1.51z^2 + 5.44z + 1.23}{z^3 - 2.6z^2 + 2.28z - 0.67}$$

An H_∞ design with suitable weighting functions on the sensitivity function $(I + GC)^{-1}$ and the complementary sensitivity function $GC(I + GC)^{-1}$ yields the following controller :

$$C(z) = \frac{160z^6 - 373.6z^5 + 108.7z^4 + 334.7z^3 - 279.1z^2 + 29.8z + 19.5}{z^6 - 1.522z^5 + 0.166z^4 + 0.29z^3 + 0.045z^2 + 0.018z + 0.0037}$$

We reduce this controller to lower order using balanced truncation (BT) and H_2-optimal reduction.

Choosing $w(z) = 1$, the smallest order to which these methods can reduce the controller so that it is still stabilizing, is $r = 3$; the H_2-optimally reduced controller found is

$$\hat{C}_{H_2} = \frac{160z^3 - 95.6z^2 - 141.6z + 107.9}{z^3 + 0.21z^2 + 0.014z + 0.066}$$

The balanced truncation (BT) reduced controller

$$\hat{C}_{BT} = \frac{160z^3 - 186.1z^2 - 139.3z + 171.8}{z^3 - 0.34z^2 - 0.38z - 0.23}$$

is not stabilizing.

Convergence is reached in 11 iterations and after 317660 flops; for comparison, the BT reduction requires 73100 flops. The convergence behavior of $\|\hat{a}_k - \hat{a}\|$ is shown in Figure 6.1 (d).

Let $H_{H_2} \triangleq (I + GC_{H_2})^{-1}GC_{H_2}$,

$$\text{then} \quad \frac{\|H_{H_2} - H\|_2}{\|H\|_2} = 0.19 \quad \text{and} \quad \frac{\|H_{H_2} - H\|_\infty}{\|H\|_\infty} = 0.25$$

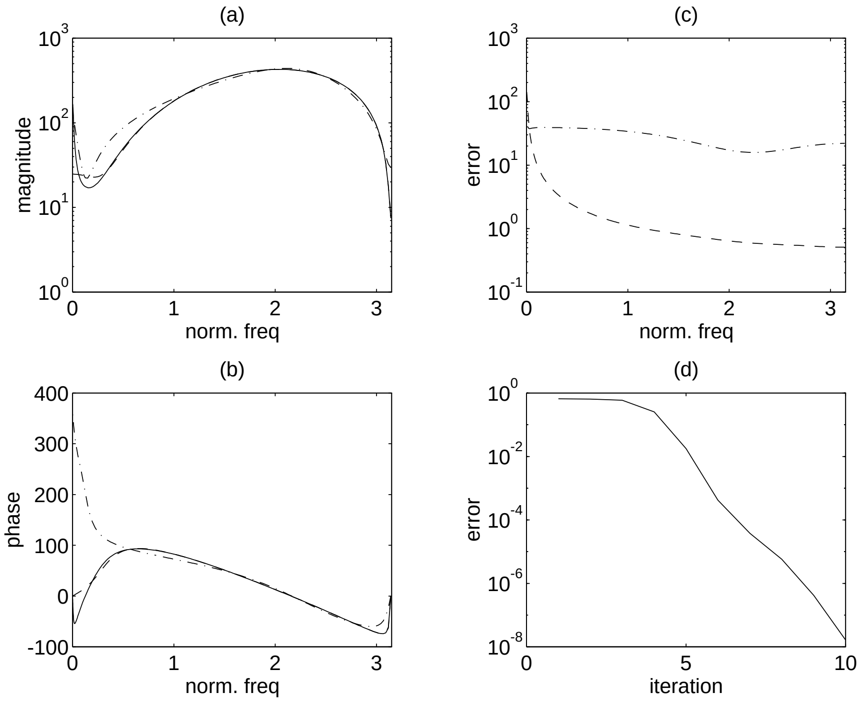


Figure 6.1: Unweighted model reduction.
Dashed line : H_2 reduced model, dash-dotted line : BT model . (a) magnitude Bode plots, (b) phase Bode plots, (c) error magnitude, (d) convergence of $\|\hat{a}_k - \hat{a}\|$

The Bode plots, and error magnitude versus frequency are shown in figure 6.1 (a)(b)(c).

If we choose the weighting function to be[1] $w(z) = (I + GC)^{-2}G$, reduced to order 5 by BT, the smallest order that can be obtained is $r = 2$, both for the

[1]This weighting function aims at minimizing the variation in the closed loop transfer function $(I + GC)^{-1}GC$

optimal controller $\hat{C}_{wH_2}$ and the controller $\hat{C}_{wBT}$ reduced by frequency weighted balanced truncation (FWBT) [93]. One finds :

$$\hat{C}_{wH_2} = \frac{160z^2 - 272.6z + 123.3}{z^2 - 0.544z + 0.01} \qquad \hat{C}_{wBT} = \frac{160z^2 - 294z + 140.1}{z^2 - 0.853z + 0.0074}$$

$$\frac{\|H_{wH_2} - H\|_2}{\|H\|_2} = 0.12 \qquad \frac{\|H_{wH_2} - H\|_\infty}{\|H\|_\infty} = 0.16$$

$$\frac{\|H_{wBT} - H\|_2}{\|H\|_2} = 0.36 \qquad \frac{\|H_{wBT} - H\|_\infty}{\|H\|_\infty} = 0.37$$

This time convergence is reached in 26 iterations.

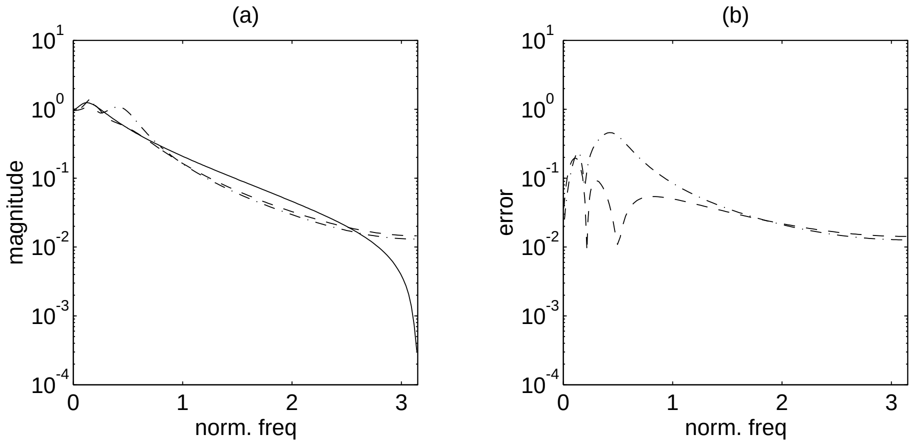


Figure 6.2: Frequency weighted model reduction.
Magnitude Bode plots (a) and errors (b) of the closed-loop transfer function $(I + GC)^{-1}GC$
(dashed line : FW H_2 reduced model, dash-dotted line : FWBT model).

6.6 Conclusions

The STLS algorithm of Chapter 5 was generalized for frequency weighted H_2 model reduction; a time-domain version may be given, but the z-domain formulation is more suitable. A more practical formulation for the z-domain version was presented, greatly reducing problems of numerical conditioning.

The algorithm is shown to be useful also for continuous-time H_2-optimal approximation.

A separate treatment is given for non-strictly proper H_2-optimal approximants.

We show that special care is required for some non-generic but meaningful cases.

Chapter 7

Time weighted model reduction

7.1 Introduction

A lot of control specifications and objectives can be translated to (possibly sufficient) conditions formulated in the frequency domain, i.e. conditions on the closed-loop transfer $H_{zw}(j\omega) = L_f(P(j\omega), C(j\omega))$; see e.g. [56] for a review of such specs. Observe that $H_{zw}(j\omega)$ is not influenced by $C(j\omega')$, where $\omega' \neq \omega$. Thus, to retain good properties of high order controllers with respect to to these objectives after controller reduction, frequency weighted model reduction is arrived at naturally, since $L_f(P(j\omega), C(j\omega)) - L_f(P(j\omega), C_r(j\omega)) \approx$
$P_{zu}(j\omega)(I - C(j\omega)P_{yu}(j\omega))^{-1}(C(j\omega) - C_r(j\omega))(I - P_{yu}(j\omega)C(j\omega))^{-1}P_{yw}(j\omega)$.

Another class of control specifications are formulated in terms of $H_{zw}(t)$, such as overshoot and undershoot, rise time, dead time, etc. How these specifications are affected by model reduction techniques is a matter on which but a few results are available [165]; likewise, there is a lack of reduction methods which can be moulded or adapted in such a way that these time-domain properties are preserved after reduction. It seems natural to surmise that for preservation of these properties, a time weighted model reduction method is needed, i.e. a method making some measure or norm of the time weighted approximation error $W_o(t)(C(t) - C_r(t))W_i(t)$ small. However, a simple relation between $C(t)$ and $H_{zw}(t)$ as before is missing; there is no guarantee whatsoever that a reduced order controller C_r matching the impulse response of the full order controller well in a certain time interval, will result in a good approximation of the closed-loop behavior in that interval. Yet time-domain properties of the controller, such as overshoot, may be interesting in itself; and certainly for plant model reduction, time weighted model reduction may have its use.

One particular time weighted norm, the Hilbert-Schmidt-Hankel norm, may be of independent importance, because of its topological equivalence with the H_∞ norm [130]; see also Subsection 2.2.7.

This chapter offers some, perhaps rather unconnected, contributions to the area of time weighted model reduction. One technique may be classified among the balancing methods, like those of Chapters 3 and 4, while the other schemes are very similar to the ones of Chapter 5 and 6.

First, it will be shown how the STLS framework can also be useful for finite-horizon H_2-optimal model reduction with time weighting.
For the special case of Hilbert-Schmidt-Hankel norm optimal model reduction, i.e. H_2-optimal model reduction with weight $w(t) = t$, an algorithm similar to the one of Chapter 6 is presented in Section 7.3.
Finally, Section 7.4 introduces time weighted gramians, which may be used for computing time weighted norms. This concept also offers the possibility of a suboptimal balancing-type method for model reduction in these norms, which may be useful in itself, or serve for initialization of optimal algorithms such as the ones in the other sections. The latter technique applies to MIMO systems, the former ones, being STLS derivates, can handle only SIMO or MISO systems.

7.2 Time weighted H_2-optimal model reduction

In this section, time weighted H_2 norms are defined, and it is shown how the STLS approach of Section 5.2 may easily be generalized to allow for a weighting function in the time domain.

7.2.1 Time weighted H_2 norms

Let $F(t)$ be a time-dependent piecewise continuous positive definite symmetric matrix, and let $G(s)$ be a CT LTI system with impulse response $G(t)$. Define the time weighted H_2 norm $\|G(s)\|_F$ by

$$\|G(s)\|_F^2 \triangleq \mathrm{Tr} \int_0^\infty G(t)^* F(t) G(t) dt \ .$$

When F_i is a positive definite symmetric matrix for every nonnegative integer i, and $G(z)$ is a DT LTI system with impulse response G_i, define $\|G(z)\|_F$ by

$$\|G(z)\|_F^2 \triangleq \mathrm{Tr} \sum_{i=0}^{\infty} G_i^* F_i G_i \ .$$

It is easily verified that both expressions really do define a norm : since $F(t)$ is positive definite symmetric, there exists a matrix function $R(t)$ such that $F(t) = R^*(t)R(t)$, and defining $\tilde{G}(s)$ to be the system with impulse response $R(t)G(t)$, $\|G(s)\|_F = \|\tilde{G}(s)\|_2$; an analogous argument holds for discrete-time systems.

A special case of interest is the one where $F(t) = f(t)I$, where $f(t)$ is a scalar function with $f(t) > 0$, $\forall\, t \geq 0$, or $F_i = f_i I$ for discrete-time systems, where $f_i > 0$, $\forall\, i \geq 0$.

7.2.2 STLS for time weighted H_2 model reduction

In Chapter 5, an STLS-based iterative algorithm was presented for finite-horizon discrete-time unweighted SISO H_2 model reduction. A time weighting f_k may be incorporated in the procedure of Section 5.2 with very few changes : the object function with Lagrange multipliers now becomes

$$\min_{\hat{h}_i, y, l, \lambda} \frac{1}{2} \sum_{i=1}^{p+r} f_i (h_i - \hat{h}_i)^2 + l^T \hat{H}_p y + \lambda (y^T y - 1) \ . \tag{7.1}$$

Differentiation of $\frac{1}{2}\sum_{i=1}^{p+r} f_i(h_i - \hat{h}_i)^2$ with respect to $\hat{h}_i$ now yields $f_i(\hat{h}_i - h_i)$, so a stationary point of the optimization problem (7.1) must satisfy the necessary conditions :

$$\begin{aligned} f_i(h_i - \hat{h}_i) &= (l \star y)_i \\ \hat{H}_p y &= 0 \\ \hat{H}_p^T l &= 0 \\ y^T y &= 1 \end{aligned}$$

Define D_f as the diagonal matrix with $[D_f]_{(i,i)} \triangleq \frac{1}{f_i}$, then these necessary conditions are equivalent to :

$$\begin{aligned} h - \hat{h} &= D_f (l \star y) = D_f T_l^T y = D_f T_y^T l \\ Hy &= (H - \hat{H}) y = T_y (h - \hat{h}) = T_y D_f T_y^T l \\ H^T l &= (H - \hat{H})^T l = T_l (h - \hat{h}) = T_l D_f T_l^T y \end{aligned}$$

In fact, a frequency domain weighting as in Subsection 6.1.3 can be taken into account at the same time without additional difficulties, i.e. one may minimize Tr $(\sum_{k=0}^{\infty} f_k (w_k \star \tilde{h}_k)^* (w_k \star \tilde{h}_k)$, with $\tilde{h} \triangleq h - \hat{h}$. With T_w defined as in Subsection 6.1.3, this results in the conditions :

$$\begin{aligned} h - \hat{h} &= T_w^T D_f T_w (l \star y) \\ Hy &= D_y l \\ H^T l &= D_l y \end{aligned}$$

where $D_y \triangleq T_y T_w^T D_f T_w T_y^T$ and $D_l \triangleq T_l T_w^T D_f T_w T_l^T$.

Using the same rationale as before, one arrives at the following iteration scheme to solve these necessary conditions :

Algorithm 7.1

$k = 0$
Initialize y_0, l_0
do {

$$\begin{aligned} D_{y_k} &= T_{y_k} T_w^T D_f T_w T_{y_k}^T \\ D_{l_k} &= T_{l_k} T_w^T D_f T_w T_{l_k}^T \\ k &\leftarrow k+1 \\ y_k &= (H^T D_{y_{k-1}}^{-1} H)^{-1} D_{l_{k-1}} y_{k-1} \\ l_k &= D_{y_{k-1}}^{-1} H y_k \\ \begin{bmatrix} y_k \\ l_k \end{bmatrix} &\leftarrow \frac{1}{\|y_k\|} \begin{bmatrix} y_k \\ l_k \end{bmatrix} \end{aligned}$$

} while $\|y_k - y_{k-1}\| > \epsilon$
$\hat{h} = h - T_w^T D_f T_w T_{l_k}^T y_k$

A translation of this algorithm to the z-domain as in Chapter 5 has not been attempted in the general case; one special case is translated in the next section, which will show how matters become pretty involved even for simple time weightings.

7.3 Hilbert-Schmidt-Hankel model reduction

The Hilbert-Schmidt-Hankel norm is a special case of the time weighted H_2 norms defined in Subsection 7.2.1, with $f(t) = t$ for CT systems or $f_i = i$ for DT systems. Hence the algorithm of Subsection 7.2.2 can be used for model reduction minimizing the Hilbert-Schmidt-Hankel norm of the approximation error. In the next subsection, an expression for the gradient of the (squared) error criterion and necessary conditions like those of Subsection 5.1.3 for H_2-optimal model reduction are derived; to our knowledge, this derivation is new. The gradient can be used to solve the minimization problem with a vanilla optimization routine as for H_2 approximation, but the necessary conditions are different from those of Chapter 5 in that they do not have a projection of dynamics structure.
In Subsection 7.3.2, it will be shown how the approach of Chapter 6 can be carried through to translate Alg. 7.1 to the z-domain for this case.

7.3.1 Optimal HSH norm model reduction

Recall that the HSH norm is defined by

$$\|G(s)\|_{HSH} \stackrel{\triangle}{=} \sqrt{\text{Tr} \int_0^\infty t\, G^T(t)G(t)\; dt}$$

for CT systems and

$$\|G(z)\|_{HSH} \stackrel{\triangle}{=} \sqrt{\text{Tr} \sum_{i=0}^{\infty} i\, G_i^T G_i}$$

for DT systems. It is therefore a special case of the norms considered in Section 7.2, with $f(t) = t$ or $f_i = i$.

Optimal model reduction in the HSH norm minimizes the HSH norm of the modeling error, i.e.

$$\min_{G_r \in \mathcal{S}_r} \|G - G_r\|_{HSH}$$

Necessary conditions for optimality may easily be found by deriving the gradient of the square of the HSH norm with respect to the state-space parameters.

Let $G(s) \stackrel{\triangle}{=} [A, B, C, D]$ be a stable continuous-time system, P and Q be its controllability and observability gramians, and the matrices P_1 and Q_1 be the solutions of $AP_1 + P_1A^T + P = 0$ and $A^TQ_1 + Q_1A + Q = 0$. In Subsection 2.2.7, it was shown that $\|G\|_{HSH}^2 = \text{Tr}\ PQ$ (actually, the same equality holds in the discrete-time case). Evidently the HSH norm is insensitive to changes in the D-matrix. An alternative expression for $J \triangleq \|G(s)\|^2_{HSH}$ is Tr $P_1 C^T C$, since

$$\text{Tr } PQ = -\text{Tr } ((AP_1 + P_1 A^T)Q) = -\text{Tr } (P_1(A^T Q + QA)) = \text{Tr } P_1 C^T C \ ;$$

analogously, one has $\|G(s)\|^2_{HSH} = \text{Tr } Q_1 B B^T$.
With this, the gradients with respect to B and C are readily found :

$$\frac{\partial}{\partial B_{ij}} J = 2[Q_1 B]_{(i,j)} \ , \qquad \frac{\partial}{\partial C_{ij}} J = 2[C P_1]_{(i,j)} \ ,$$

$$\frac{\partial}{\partial A_{ij}} J = \text{Tr } (\frac{\partial P}{\partial A_{ij}} Q + P \frac{\partial Q}{\partial A_{ij}}) \ .$$

$$\begin{array}{lrcl} \text{Now} & A\frac{\partial P}{\partial A_{ij}} + \frac{\partial P}{\partial A_{ij}} A^T + e_i e_j^T P + P e_j e_i^T & = & 0 \\ \text{and} & A^T \frac{\partial Q}{\partial A_{ij}} + \frac{\partial Q}{\partial A_{ij}} A + Q e_i e_j^T + e_j e_i^T Q & = & 0 \ . \end{array}$$

where e_i is the vector with 1 as the i-th element and zeros elsewhere;

$$\begin{aligned} \text{hence } \frac{\partial}{\partial A_{ij}} J &= -\text{Tr } \left(\frac{\partial P}{\partial A_{ij}} (A^T Q_1 + Q_1 A) + (AP_1 + P_1 A^T) \frac{\partial Q}{\partial A_{ij}} \right) \\ &= -\text{Tr } \left((A \frac{\partial P}{\partial A_{ij}} + \frac{\partial P}{\partial A_{ij}} A^T) Q_1 + P_1 (A^T \frac{\partial Q}{\partial A_{ij}} + \frac{\partial Q}{\partial A_{ij}} A) \right) \\ &= \text{Tr } (e_i e_j^T P + P e_j e_i^T) Q_1 + P_1 (Q e_i e_j^T + e_j e_i^T Q) \\ &= e_j^T (P Q_1 + P_1 Q) e_i + e_i^T (Q_1 P + Q P_1) e_j \\ &= 2[Q_1 P + Q P_1]_{(i,j)} \ . \end{aligned}$$

Using these gradients, it is trivial to derive necessary conditions for (possibly frequency weighted) optimal HSH model reduction, HSH optimal static output feedback design, or HSH optimal fixed order control. Only the gradients for the optimization criterion for unweighted optimal HSH model reduction are given below, using the following definitions :

$$G_r(s) \triangleq \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array} \right] \ ,$$

$$\tilde{G}(s) \triangleq G(s) - G_r(s) = \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array} \right] = \left[\begin{array}{cc|c} A & 0 & B \\ 0 & \hat{A} & \hat{B} \\ \hline C & -\hat{C} & D - \hat{D} \end{array} \right] \ ,$$

$$\text{with}\quad \begin{array}{rcl} \tilde{A}\tilde{P}+\tilde{P}\tilde{A}^T+\tilde{B}\tilde{B}^T &=& 0 \\ \tilde{A}\tilde{P}_1+\tilde{P}_1\tilde{A}^T+\tilde{P} &=& 0 \end{array} \qquad \begin{array}{rcl} \tilde{A}^T\tilde{Q}+\tilde{Q}\tilde{A}+\tilde{C}^T\tilde{C} &=& 0 \\ \tilde{A}^T\tilde{Q}_1+\tilde{Q}_1\tilde{A}+\tilde{Q} &=& 0 \end{array}$$

Let $\tilde{J} \triangleq \|\tilde{G}(s)\|^2_{HSH}$, then

$$\begin{aligned} \frac{\partial}{\partial \hat{A}_{ij}}\tilde{J} &= 2\left[\begin{bmatrix} 0 & I \end{bmatrix}(\tilde{Q}_1\tilde{P}+\tilde{Q}\tilde{P}_1)\begin{bmatrix} 0 \\ I \end{bmatrix}\right]_{(i,j)} \\ \frac{\partial}{\partial \hat{B}_{ij}}\tilde{J} &= 2\left[\begin{bmatrix} 0 & I \end{bmatrix}\tilde{Q}_1\tilde{B}\right]_{(i,j)} \\ \frac{\partial}{\partial \hat{C}_{ij}}\tilde{J} &= 2\left[\tilde{C}\tilde{P}_1\begin{bmatrix} 0 \\ I \end{bmatrix}\right]_{(i,j)} \end{aligned}$$

Partitioning the gramians in the same way as $\tilde{A}$, i.e.
$\tilde{P}=\begin{bmatrix} P & \bar{P} \\ \bar{P}^T & \hat{P} \end{bmatrix}$, $\tilde{Q}=\begin{bmatrix} Q & \bar{Q} \\ \bar{Q}^T & \hat{Q} \end{bmatrix}$, $\tilde{P}_1=\begin{bmatrix} P_1 & \bar{P}_1 \\ \bar{P}_1^T & \hat{P}_1 \end{bmatrix}$ and $\tilde{Q}_1=\begin{bmatrix} Q_1 & \bar{Q}_1 \\ \bar{Q}_1^T & \hat{Q}_1 \end{bmatrix}$,
one finds the following necessary conditions for optimality :

$$\begin{aligned} \bar{Q}_1^T\bar{P}+\bar{Q}^T\bar{P}_1 &= -\hat{Q}_1\hat{P}-\hat{Q}\hat{P}_1 \\ \bar{Q}_1^T B &= -\hat{Q}_1\hat{B} \\ C\bar{P}_1 &= \hat{C}\hat{P}_1 \end{aligned} \tag{7.2}$$

In contrast with H_2-optimal model reduction, no projection of dynamics structure can be derived from these relations.

If $G(z) \triangleq [A,B,C,D]$ is a stable discrete-time system, P and Q are its controllability and observability gramians, and the matrices P_1 and Q_1 the solutions of $AP_1A^T-P_1+P=0$ and $A^TQ_1A-Q_1+Q=0$, then only the gradient with respect to A_{ij} is different from the continuous-time gradient :

$$\begin{aligned} \frac{\partial}{\partial A_{ij}}J &= \operatorname{Tr}\left(\frac{\partial P}{\partial A_{ij}}(Q_1-A^TQ_1A)+(P_1-AP_1A^T)\frac{\partial Q}{\partial A_{ij}}\right) \\ &= \operatorname{Tr}\left((\frac{\partial P}{\partial A_{ij}}-A\frac{\partial P}{\partial A_{ij}}A^T)Q_1+P_1(\frac{\partial Q}{\partial A_{ij}}-A^T\frac{\partial Q}{\partial A_{ij}}A)\right) \\ &= \operatorname{Tr}\left((e_ie_j^TPA^T+APe_je_i^T)Q_1+P_1(A^TQe_ie_j^T+e_je_i^TQA)\right) \\ &= 2[Q_1AP+QAP_1]_{(i,j)}\ . \end{aligned}$$

Hence for the discrete-time HSH norm, condition (7.2) should be replaced by

$$\bar{Q}_1^TA\bar{P}+\bar{Q}^T\hat{A}\bar{P}_1=-\hat{Q}_1A\hat{P}-\hat{Q}\hat{A}\hat{P}_1\ .$$

7.3.2 STLS formulation and iteration scheme

For the special case of discrete-time HSH model reduction, a z-domain algorithm may again be presented (as in Chapters 5 and 6) for the limit case of procedure of Subsection 7.2.2 as the time horizon p is taken to infinity.

Since $\sum_{i=1}^{\infty} i h_i z^{-i} = -z\frac{d}{dz}h(z)$, the definition of the HSH norm can be rewritten

$$\|h(z)\|_{HSH}^2 \triangleq \sum_{i=1}^{\infty} i h_i^2 = -\frac{1}{2\pi j}\oint_T h(z^{-1})\frac{d}{dz}h(z)dz \ .$$

For frequency weighted HSH model reduction, the minimization criterion is

$$\begin{aligned} J &= \frac{1}{2}\|w(z)h(z)\|_{HSH}^2 \\ &= \frac{1}{2}\sum_{i=1}^{\infty} i\left(w(z)\star\left(h(z)-\hat{h}(z)\right)\right)_i^2 \\ &= -\frac{1}{4\pi j}\oint_T w(z^{-1})\left(h(z^{-1})-\hat{h}(z^{-1})\right)\frac{d}{dz}\left[w(z)\left(h(z)-\hat{h}(z)\right)\right]dz \ . \end{aligned}$$

Using Lagrange multipliers as in Section 6.1.2, one derives the following necessary conditions for optimality :

$$\left[w(z^{-1})z\frac{d}{dz}\left[w(z)\left(h(z)-\hat{h}(z)\right)\right]\right]_+ + \ l(z)y(z) \ = \ 0 \ ,$$

$$\left[\hat{h}(z)y(z^{-1})\right]_+ = 0 \ , \qquad \left[\hat{h}(z)l(z^{-1})\right]_{0\ldots r} = 0 \ , \qquad \|y\| = 1 \ .$$

By Lemma 6.1, the first condition is equivalent to :

$$h(z)-\hat{h}(z) = -w^{-1}(z)\int z^{-1}\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ dz \ .$$

The indefinite integral is taken such that the constant term is zero, i.e. it vanishes as $z\to\infty$. This expression again enables us to eliminate $\hat{h}(z)$, to find the following equations :

$$\begin{aligned} \left[h(z)y(z^{-1})\right]_+ &= -\left[w^{-1}(z)y(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ dz\right]_+ \ , \\ \left[h(z)l(z^{-1})\right]_{0\ldots r} &= -\left[w^{-1}(z)l(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ dz\right]_{0\ldots r} \ . \end{aligned}$$

In this context, the iteration of Alg. 7.1 reads as follows :

$$\begin{aligned}
&\left[h(z)y^{[k]}(z^{-1})\right]_+ \\
&\quad = -\left[w^{-1}(z)y^{[k-1]}(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k-1]}(z)\right]_+ dz\right]_+ , \\
&\left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} \\
&\quad = -\left[w^{-1}(z)l^{[k-1]}(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ dz\right]_{0\ldots r} .
\end{aligned}$$

$\hat{a}^{[k]}(z) = y^{[k]}(z^{-1})$ as before, and again we have :

$$\hat{a}^{[k]}(\alpha_i) = H_i^{-1}w^{-1}(\alpha_i)\hat{a}^{[k-1]}(\alpha_i^{-1})X_i^{[k]} \triangleq D_i^{[k-1]}X_i^{[k]} ,$$

so that (6.13) and (6.14) still hold.

With the definition $\xi^{[k]}(z) \triangleq -\int z^{-1}\left[w^{-1}(z^{-1})\hat{a}^{[k-1]}(z^{-1})l^{[k]}(z)\right]_+ dz$, the equation $\left[h(z)y^{[k]}(z^{-1})\right]_+ = \left[w^{-1}(z)\hat{a}^{[k-1]}(z)\xi^{[k]}(z)\right]_+$ still holds, and the form of $\xi^{[k]}(z)$ is the same as in (6.8), i.e. $\xi^{[k]}(z) = w(z)\frac{\phi^{[k]}(z)}{a(z)\hat{a}^{[k-1]}(z)}$, and (6.17) is still valid as well.

A different expression is found, however, for $l^{[k]}(z)$:

$$\begin{aligned}
\left[w^{-1}(z^{-1})\hat{a}^{[k-1]}(z^{-1})l^{[k]}(z)\right]_+ &= -z\frac{d}{dz}\xi^{[k]}(z) = \sum_i \frac{zX_i^{[k]}}{(z-a_i^{[k-1]})^2} \\
&= \sum_i \frac{a_i^{[k-1]}X_i^{[k]}}{(z-a_i^{[k-1]})^2} + \sum_i \frac{X_i^{[k]}}{z-a_i^{[k-1]}} ,
\end{aligned}$$

$$\text{hence} \quad l^{[k]}(z) = \frac{z^r\psi^{[k]}(z)}{(a(z)\hat{a}^{[k-1]}(z)d_w(z))^2} = \sum_i \frac{\gamma_i^{[k-1]}X_i^{[k]}}{(z-a_i^{[k-1]})^2} + \sum_i \frac{\delta_i^{[k-1]}X_i^{[k]}}{z-a_i^{[k-1]}}$$

where $\gamma_i^{[k]}$, $\delta_i^{[k]}$ are easily found to be :

$$\begin{aligned}
\gamma_i^{[k]} &\triangleq \frac{a_i^{[k]}w((a_i^{[k]})^{-1})}{\hat{a}^{[k]}((a_i^{[k]})^{-1})} , \\
\delta_i^{[k]} &\triangleq \gamma_i^{[k]}(a_i^{[k]})^{-1}\left(1-\gamma_i^{[k]}\frac{d}{dz}\left(w^{-1}(z^{-1})\hat{a}^{[k]}(z^{-1})\right)\Big|_{a_i^{[k]}}\right) \\
&= \gamma_i^{[k]}(a_i^{[k]})^{-1}\left(1+\left[z^2\gamma_i^{[k]}\frac{d}{dz}\left(w^{-1}(z)\hat{a}^{[k]}(z)\right)\right]_{(a_i^{[k]})^{-1}}\right)
\end{aligned}$$

$$\begin{aligned}
&= (a_i^{[k]})^{-1}\left[\gamma_i^{[k]} - \frac{d}{dz}\left(\frac{w(z)}{\hat{a}^{[k]}(z)}\right)\right]_{(a_i^{[k]})^{-1}} \\
&= \left[\frac{w(z^{-1})}{\hat{a}^{[k]}(z^{-1})} + z\frac{d}{dz}\left(\frac{w(z^{-1})}{\hat{a}^{[k]}(z^{-1})}\right)\right]_{a_i^{[k]}} \\
&= \left[\frac{w(z)}{\hat{a}^{[k]}(z)} - z\frac{d}{dz}\left(\frac{w(z)}{\hat{a}^{[k]}(z)}\right)\right]_{(a_i^{[k]})^{-1}} .
\end{aligned}$$

Since $l^{[k]}(z)$ contains an r-multiple zero at $z = 0$, it evaluates to zero at $z = 0$ together with its first $r-1$ derivatives, giving r conditions on $X_i^{[k]}$:

$$(-1)^{s+1}(s-1)!\left(s\sum_i \frac{\gamma_i^{[k-1]}X_i^{[k]}}{(z-a_i^{[k-1]})^{s+1}} - \frac{\delta_i^{[k-1]}X_i^{[k]}}{(z-a_i^{[k-1]})^{s}}\right)_{z=0} = 0$$

$$\text{hence :} \qquad s\sum_i \frac{\gamma_i^{[k-1]}X_i^{[k]}}{(a_i^{[k-1]})^{s+1}} - \sum_i \frac{\delta_i^{[k-1]}X_i^{[k]}}{(a_i^{[k-1]})^{s}} = 0 .$$

$$\text{or :} \qquad P_\alpha^{[k-1]}X_\alpha^{[k]} + P_{\hat{\alpha}}^{[k-1]}X_{\hat{\alpha}}^{[k]} + P_w^{[k-1]}X_w^{[k]} = P^{[k-1]}X^{[k]} = 0 , \tag{7.3}$$

where $P_{si}^{[k]} \triangleq (a_i^{[k]})^{-(s+1)}(s\gamma_i^{[k]} - a_i^{[k]}\delta_i^{[k]})$.

The $r+1$ still missing conditions are provided by the equation

$$\begin{aligned}
&\left[h(z)l^{[k]}(z^{-1})\right]_{0\ldots r} \\
&\quad = -\left[w^{-1}(z)l^{[k-1]}(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ dz\right]_{0\ldots r} ,
\end{aligned}$$

$$\begin{aligned}
\text{or :}\quad \left[h(z)l^{[k]}(z^{-1})\right]_s &= \left[\sum_i \frac{H_i}{z-\alpha_i} l^{[k]}(\alpha_i)\right]_s \\
&= \sum_i \frac{H_i}{\alpha_i}\alpha_i^s \left[\sum_j \frac{\gamma_j^{[k-1]}X_j^{[k]}}{(\alpha_i^{-1}-a_j^{[k-1]})^2} + \sum_j \frac{\delta_j^{[k-1]}X_j^{[k]}}{\alpha_i^{-1}-a_j^{[k-1]}}\right] , \; s = 1,\ldots,r .
\end{aligned}$$

This gives $r+1$ conditions on $X_i^{[k]}$, which can be written as

$$J^{[k-1]}X^{[k]} = J_\alpha^{[k-1]}X_\alpha^{[k]} + J_{\hat{\alpha}}^{[k-1]}X_{\hat{\alpha}}^{[k]} + J_w^{[k-1]}X_w^{[k]} = V^{[k-1]} , \tag{7.4}$$

where $J_{sj}^{[k]} \triangleq \sum_i H_i\alpha_i^s(a_j^{[k]}\gamma_j^{[k]}(1-a_j^{[k]}\alpha_i)^{-2} - \delta_j^{[k]}(1-a_j^{[k]}\alpha_i)^{-1})$
and where $V^{[k-1]} \in \mathbb{R}^{(r+1)\times 1}$ is defined by

$$V_s^{[k-1]} \triangleq -\left[w^{-1}(z)l^{[k-1]}(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l^{[k-1]}(z)y^{[k-1]}(z)\right]_+ dz\right]_s .$$

We now show how to compute $V^{[k]}$ from $l^{[k]}(z)$ and $y^{[k]}(z)$. In order not to overburden the notation, we shall write a_i for $a_i^{[k-1]}$ in the sequel.

$$\begin{aligned}
\exists\, T_i^{[k]}, S_i^{[k]} \text{ s.t. } \left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k]}(z)\right]_+ &= \sum_i \frac{zT_i^{[k]}}{(z-a_i)^2} + \sum_i \frac{a_i S_i^{[k]}}{z-a_i} \\
&= \left[w^{-1}(z^{-1})\hat{a}^{[k]}(z^{-1}) \sum_i \left(\frac{\gamma_i^{[k-1]}}{(z-a_i)^2} + \frac{\delta_i^{[k-1]}}{z-a_i}\right) X_i^{[k]}\right]_+
\end{aligned}$$

and interpolation in the poles will show :

$$\begin{aligned}
T_i^{[k]} &= \gamma_i^{[k-1]} X_i^{[k]} \left[z^{-1}w^{-1}(z^{-1})\hat{a}^{[k]}(z^{-1})\right]_{z=a_i} = \frac{\hat{a}^{[k]}(a_i^{-1})}{\hat{a}^{[k-1]}(a_i^{-1})} X_i^{[k]}\,, \\
S_i^{[k]} &= \delta_i^{[k-1]} \left[z^{-1}w^{-1}(z^{-1})\hat{a}^{[k]}(z^{-1})\right]_{z=a_i} \\
&\qquad + a_i \gamma_i^{[k-1]} \frac{d}{dz}\left[z^{-1}w^{-1}(z^{-1})\hat{a}^{[k]}(z^{-1})\right]_{z^{-1}=a_i} X_i^{[k]} \\
&= \left(\delta_i^{[k-1]} - a_i^{-1}\gamma_i^{[k-1]}\right)\left[z^{-1}w^{-1}(z^{-1})\hat{a}^{[k]}(z^{-1})\right]_{z=a_i} T_i^{[k]} \\
&\qquad - a_i^{-3}\gamma_i^{[k-1]} \frac{d}{dz}\left(w^{-1}(z)\hat{a}^{[k]}(z)\right)_{z=a_i^{-1}} X_i^{[k]} \\
&= \frac{d}{dz}\left[\frac{\frac{\hat{a}^{[k-1]}(z^{-1})}{\hat{a}^{[k-1]}(a_i^{-1})} - \frac{\hat{a}^{[k]}(z^{-1})}{\hat{a}^{[k]}(a_i^{-1})}}{\frac{w(z)}{w(a_i^{-1})}}\right]_{z=a_i^{-1}} T_i^{[k]}\,.
\end{aligned}$$

Note that $T_i^{[k]} \to X_i^{[k]}$ and $S_i^{[k]} \to 0$ as the algorithm converges. Indeed, omission of the cumbersome computation of the terms containing $S_i^{[k]}$ hardly affects the efficiency of the algorithm.

An expression for $V_s^{[k]}$ is now computed as follows :

$$-\int z^{-1}\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k]}(z)\right]_+ dz = \sum_i \frac{T_i^{[k]}}{z-a_i} - \sum_i S_i^{[k]} \log|1-a_i z^{-1}|$$

$$\begin{aligned}
\Rightarrow \qquad V_s^{[k]} &= -\left[w^{-1}(z)l^{[k]}(z^{-1})\int z^{-1}\left[w^{-1}(z^{-1})l^{[k]}(z)y^{[k]}(z)\right]_+ dz\right]_s \\
&= \sum_i T_i^{[k]}\left[w^{-1}(z)l^{[k]}(z^{-1})\frac{1}{z-a_i}\right]_s \\
&\qquad - \sum_i S_i^{[k]}\left[w^{-1}(z)l^{[k]}(z^{-1})\log|1-a_i z^{-1}|\right]_s
\end{aligned}$$

The terms of the first summation are the easiest ones :

$$\begin{aligned}\sum_i T_i^{[k]} &\left[w^{-1}(z)l^{[k]}(z^{-1})\frac{1}{z-a_i}\right]_s \\ &= \left[\sum_i \frac{R_{\beta i}^{[k]}}{z-\beta_{wi}} + \sum_i \frac{R_{\alpha i}^{[k]}}{z-\alpha_i} + \sum_i \frac{R_{\hat{\alpha} i}^{[k]}}{z-\hat{\alpha}_i^{[k-1]}}\right]_s\end{aligned}$$

where the residues $R_{\beta i}^{[k]}$, $R_{\alpha i}^{[k]}$ and $R_{\hat{\alpha} i}^{[k]}$ are again found by interpolation in the poles β_{wi}, α_i and $\hat{\alpha}_i^{[k-1]}$:

$$\begin{aligned}R_{\beta i}^{[k]} &\triangleq l^{[k]}(\beta_{wi}^{-1})W_i \sum_j \frac{T_j^{[k]}}{\beta_{wi}^{-1} - a_j^{[k-1]}} \ , \\ R_{\alpha i}^{[k]} &\triangleq w^{-1}(\alpha_i)l^{[k]}(\alpha_i^{-1})T_{\alpha i}^{[k]} \ , \\ R_{\hat{\alpha} i}^{[k]} &\triangleq w^{-1}(\hat{\alpha}_i)l^{[k]}(\hat{\alpha}_i^{-1})T_{\hat{\alpha} i}^{[k]} \ ,\end{aligned}$$

$$\begin{aligned}\text{hence} \quad \sum_i T_i^{[k]} &\left[w^{-1}(z)l^{[k]}(z^{-1})\frac{1}{z-a_i}\right]_s \\ &= \sum_i R_{\beta i}^{[k]}\beta_{wi}^{s-1} + \sum_i R_{\alpha i}^{[k]}\alpha_i^{s-1} + \sum_i R_{\hat{\alpha} i}^{[k]}(\hat{\alpha}_i^{[k-1]})^{s-1} \ .\end{aligned}$$

To compute the second summation, the following lemma is needed :

Lemma 7.1

$$\left[w(z)l(z^{-1})\log|1-az^{-1}|\right]_s = \left[\int z^{-1}\left[z^{-s}w(z^{-1})l(z)\right]_+ dz\right]_{z=a^{-1}}$$

Proof : see Appendix A.2. □

To compute the integral, the integrand is rewritten

$$\left[z^{-s}w^{-1}(z^{-1})l^{[k]}(z)\right]_+ = \sum_j \left[\frac{zM_j^{s[k]}}{(z-a_j)^2} - \frac{a_jN_j^{s[k]}}{z-a_j}\right]_+$$

where the $M_j^{s[k]}$, $N_j^{s[k]}$ are found using the recursion

$$\begin{aligned}M_j^{(s+1)[k]} &= a_j^{-1}M_j^{s[k]} \\ N_j^{(s+1)[k]} &= a_j^{-1}\left(N_j^{s[k]} + M_j^{(s+1)[k]}\right)\end{aligned}$$

starting from

$$
\begin{aligned}
M_j^{0[k]} &= a_j^{-1} w^{-1}(a_j^{-1}) \gamma_j^{[k-1]} X_j^{[k]} \\
N_j^{0[k]} &= a_j^{-1} \left(a_j^{-1} \gamma_j^{[k-1]} - \delta_j^{[k-1]} \right) w^{-1}(a_j^{-1}) X_j^{[k]} \\
&\qquad + a_j^{-3} \gamma_j^{[k-1]} X_j^{[k]} \frac{d}{dz} \left(w^{-1}(z) \right)_{a_j^{-1}} .
\end{aligned}
$$

As
$$
\begin{aligned}
&\int z^{-1} \left[z^{-s} w^{-1}(z^{-1}) l^{[k]}(z) \right]_+ dz \\
&\qquad = -\sum_j \left(\frac{M_j^{s[k]}}{z - a_j} + N_j^{s[k]} \log |1 - a_j z^{-1}| \right)
\end{aligned}
$$
and thus
$$
\begin{aligned}
&\int z^{-1} \left[z^{-s} w^{-1}(z^{-1}) l^{[k]}(z) \right]_+ dz \,\Big|_{z=a_i^{-1}} \\
&\qquad = -\sum_j \left(\frac{M_j^{s[k]}}{a_i - a_j} + N_j^{s[k]} \log |1 - a_j a_i| \right) ,
\end{aligned}
$$
an expression is found for the second summation :

$$
\begin{aligned}
&\sum_i S_i^{[k]} \left[w^{-1}(z) l^{[k]}(z^{-1}) \log |1 - a_i z^{-1}| \right]_s \\
&\qquad = -\sum_j M_j^{s[k]} \sum_i \frac{a_i S_i^{[k]}}{1 - a_i a_j} - \sum_j N_j^{s[k]} \sum_i \log |1 - a_i a_j| S_i^{[k]} .
\end{aligned}
$$

Piecing it all together, we find that (6.14)(7.3)(6.17)(7.4) yield $n + r + w$ linear conditions in $n+r+w$ unknowns $X_i^{[k]}$, i.e. the following $(n+r+w) \times (n+r+w)$ linear system has to be solved each time :

$$
\begin{bmatrix}
\Gamma_r D^{[k-1]} & 0 & 0 \\
P_\alpha^{[k-1]} & P_{\hat\alpha}^{[k-1]} & P_w^{[k-1]} \\
G_\alpha & G_{\hat\alpha}^{[k-1]} & G_w \\
J_\alpha^{[k-1]} & J_{\hat\alpha}^{[k-1]} & J_w^{[k-1]}
\end{bmatrix}
\begin{bmatrix}
X_\alpha^{[k]} \\
X_{\hat\alpha}^{[k]} \\
X_w^{[k]}
\end{bmatrix}
=
\begin{bmatrix}
0 \\ 0 \\ 0 \\ V^{[k-1]}
\end{bmatrix} . \tag{7.5}
$$

7.3.3 Initialization and computation of the reduced model

For the initial $\xi(z)$ we take an analogous approach as in the H_2 case :

$$
l^{[0]}(z) \triangleq \left[\frac{z w(z^{-1})}{y^0(z)} \frac{d}{dz} \left(w(z) \left(h(z) - \hat{h}(z) \right) \right) \right]_+ .
$$

Also the computation of $\hat{h}(z)$ is analogous to the H_2 case :

$$\begin{aligned}\hat{h}(z)=\sum_{i=1}^{n}\frac{\hat{H}_i}{z-\hat{\alpha}_i} &= h(z)+w^{-1}(z)\int z^{-1}\left[w^{-1}(z^{-1})l(z)y(z)\right]_+ dz \\ &= h(z)-w^{-1}(z)\xi(z) \\ \text{hence}\qquad \hat{H}_i &= -w^{-1}(\hat{\alpha}_i)X_{\hat{\alpha}i}\ . \end{aligned} \tag{7.6}$$

7.3.4 Algorithm

The two preceding subsections are summarized in Alg. 7.2.

Algorithm 7.2

$k=0$
Take an initial approximation $\hat{h}_0(z)=\frac{\hat{b}_0(z)}{\hat{a}_0(z)}$
Compute $y^{[0]}(z)$, $l^{[0]}(z)$
do {
1. $k \leftarrow k+1$
2. *Solve the linear system (7.5)* $\Rightarrow l^{[k]}(z)$
3. *Find* $\hat{a}^{[k]}$ *from (6.13)* $\Rightarrow y^{[k]}(z)$
} while $\|\hat{a}^{[k]}-\hat{a}^{[k-1]}\|>\epsilon$
Compute $\hat{h}(z)$ *using (7.6)*

7.3.5 Continuous-time systems

As the bilinear transform preserves the HSH norm, a continuous-time optimal HSH model reduction problem

$$G_r^*(s)=\arg\min_{G_r(s)\in\mathcal{S}_r}\|G(s)-G_r(s)\|_{HSH}$$

may be mapped into a discrete-time one by considering

$$G_{rd}^*(z)=\arg\min_{G_{rd}(z)\in\mathcal{S}_r}\|G_d(z)-G_{rd}(z)\|_{HSH}$$

where $G_d(z)\triangleq\mathcal{B}_T(G(s))$; evidently $G_r^*(s)=\mathcal{B}_T^{-1}(G_{rd}^*(z))$ is optimal.

Note that $G_r^*(s)$ and $G(s)$ have in general different D-matrices; but as the HSH norm of a system $[A, B, C, D]$ does not depend on the value of D, the D-matrix of $G_r^*(s)$ may be chosen equal to the D-matrix of $G(s)$, if desired.

7.4 Time Weighted Balanced Truncation.

In this section, a suboptimal balancing-type model reduction method is put forward that may be used to provide an initial guess for optimal time weighted H_2 model reduction. To this end, time weighted gramians are defined, both for continuous-time and discrete-time systems, and it is shown how these can be computed and used for model reduction.

For *frequency* weighted model reduction, Wortelboer et al. [299] propose an extension of Balanced Truncation with piecewise polynomial weighting functions; piecewise polynomial *time* weighting functions are considered here.

7.4.1 Time weighted gramians : continuous-time

Recall that for a continuous-time system $[A, B, C, D]$, the controllability gramian P and the observability gramian Q are defined as

$$P \triangleq \int_0^\infty e^{At} BB^T e^{A^T t} dt \qquad \text{and} \qquad Q \triangleq \int_0^\infty e^{A^T t} C^T C e^{A^t} dt \; .$$

These can be found as the solution of the Lyapunov equations

$$AP + PA^T + BB^T = 0 \qquad \text{and} \qquad A^T Q + QA + C^T C = 0 \; .$$

We wish to generalize the definition of these gramians to expressions of the type

$$P_f \triangleq \int_0^\infty f(t) e^{At} BB^T e^{A^T t} dt \qquad \text{and} \qquad Q_f \triangleq \int_0^\infty f(t) e^{A^T t} C^T C e^{A^t} dt \; ,$$

where $f(t)$ should be suitable to approximate any desired weighting function.

Let's first investigate functions of the type $f(t) = t^i$, where i is a positive integer, and define

$$X_i \triangleq \int_0^\infty t^i e^{A_1 t} D e^{A_2 t} dt \qquad \text{and} \qquad X_i(T) \triangleq \int_0^T t^i e^{A_1 t} D e^{A_2 t} dt \; ;$$

in particular, we define

$$P_i \triangleq \int_0^\infty t^i e^{At} BB^T e^{A^T t} dt \qquad \text{and} \qquad Q_i \triangleq \int_0^\infty t^i e^{A^T t} C^T C e^{A^t} dt \; .$$

$$\text{Using} \qquad \frac{dX_i(T)}{dT} = T^i e^{A_1 T} D e^{A_2 T}$$

$$\text{and } \frac{d^2 X_i(T)}{d^2 T} = iT^{i-1} e^{A_1 T} D e^{A_2 T} + T^i A_1 e^{A_1 T} D e^{A_2 T} + T^i e^{A_1 T} D e^{A_2 T} A_2 \ ,$$

$$\begin{aligned} \text{it is found :} \qquad \left.\frac{dX_i(T)}{dT}\right|_{\infty} - \left.\frac{dX_i(T)}{dT}\right|_{0} &= \int_0^\infty \frac{d^2 X_i(T)}{d^2 T} dT \\ &= iX_{i-1} + A_1 X_i + X_i A_2 \\ &= 0 \qquad \text{if} \qquad i > 0 \\ &= D \qquad \text{if} \qquad i = 0 \ . \end{aligned}$$

Therefore the quantities X_i may be found by solving recursively :

$$\begin{aligned} A_1 X_0 + X_0 A_2 + D &= 0 \\ A_1 X_i + X_i A_2 + iX_{i-1} &= 0 \qquad i > 0 \ . \end{aligned}$$

It is also easily found (see also [103]) that

$$\begin{aligned} X_i(\tau_1, \tau_2) &\triangleq \int_{\tau_1}^{\tau_2} t^i e^{A_1 t} D e^{A_2 t} dt \\ &= e^{A_1 \tau_1} \left(\int_0^\infty (t+\tau_1)^i e^{A_1 t} D e^{A_2 t} dt \right) e^{A_2 \tau_1} \\ &\quad - e^{A_1 \tau_2} \left(\int_0^\infty (t+\tau_2)^i e^{A_1 t} D e^{A_2 t} dt \right) e^{A_2 \tau_2} \\ &= e^{A_1 \tau_1} \left(\sum_{k=0}^{i} \frac{i!}{k!(i-k)!} \tau_1^{i-k} X_k \right) e^{A_2 \tau_1} \\ &\quad - e^{A_1 \tau_2} \left(\sum_{k=0}^{i} \frac{i!}{k!(i-k)!} \tau_2^{i-k} X_k \right) e^{A_2 \tau_2} \ , \end{aligned}$$

which enables us to compute the gramians weighted by a piecewise polynomial in the time domain; in this case, $A_1 = A = A_2^T$.

With the techniques provided above, one can now compute P_f and Q_f for every piecewise polynomial function[1] $f(t)$ (with finitely many discontinuities),
i.e. $\exists\, t_i,\ i = 1, \ldots, n,$ s.t. $f(t) = p_i(t),\ t_i < t < t_{i+1},\ i = 0, \ldots, n,$
where $t_0 \triangleq 0$, $t_{n+1} \triangleq \infty$ and $p_i(t)$ is a polynomial in t.

[1] $f(t)$ may in fact be replaced by a piecewise polynomial symmetric matrix-valued function $F(t)$ without considerable complications.

If n_f is the order of the piecewise polynomial (defined as the highest order of its polynomial sections, i.e. $n_f = \deg f(t) \triangleq \max_i \deg p_i(t))$, then $n_f + 1$ Lyapunov equations of the form $A^T X + XA + (\cdot) = 0$ and $n_f + 1$ of the form $AX + XA^T + (\cdot) = 0$ have to be solved. Solving such a Lyapunov equation classically involves finding a Schur decomposition of A, requiring about $25n^3$ flops (see [116]); after that, the solution of the Lyapunov equation can be found by back-substitution after approximately n^3 flops. Of course the Schur decomposition needs only be computed one time, such that all the Lyapunov equations can be solved in $(27 + 2n_f)n^3$[2]. If n_f is large, one may also consider (block) diagonalizing A; this requires little extra work, and then each Lyapunov equation may be solved with approximately $O(n^2)$ flops. However, as this (block) diagonalization cannot be performed using orthogonal transformations only, this may be numerically unreliable, so caution is advisable.

The gramians can also be computed if $f(t)$ is a rational function in t without poles on the positive real axis (not including infinity), at the cost of much more complicated computations. The expressions can be reduced to integrals of the type $\int_\alpha^\infty \frac{e^{\lambda t}}{t} dt$, $\int_\alpha^\infty \frac{e^{\lambda t}\cos(\gamma t)}{t} dt$ and $\int_\alpha^\infty \frac{e^{\lambda t}\sin(\gamma t)}{t} dt$; these cannot be solved analytically, but are tabulated in [2].

7.4.2 Time weighted gramians : discrete-time

For discrete-time systems, the controllability and observability gramians are defined as $P \triangleq \sum_0^\infty A^k BB^T A^{kT}$ and $Q \triangleq \sum_0^\infty A^{kT} C^T C A^k$, respectively. They can be computed by solving the Lyapunov equations $APA^T - P + BB^T = 0$ and $A^T QA - Q + C^T C = 0$.

Again we generalize these definitions to expressions of the type :

$$P_f \triangleq \sum_{k=0}^{\infty} f(k) A^k BB^T A^{kT} \qquad \text{and} \qquad Q_f \triangleq \sum_{k=0}^{\infty} f(k) A^{kT} C^T C A^k \ ,$$

where $f(k)$ is a piecewise polynomial function.

Here we do not define the matrices X_i as $\sum_{k=0}^\infty k^i A_1^k D A_2^k$, as one could expect from analogy with the continuous-time case, but as

$$X_i \triangleq \sum_{k=0}^{\infty} \frac{(k+i)!}{k!} A_1^k D A_2^k \ .$$

[2]Some $6n^3$ or so extra flops may be necessary for reordering the Schur decomposition in order to get better numerical reliability in the subsequent back-substitution.

Accordingly, we define

$$P_i \triangleq \sum_{k=0}^{\infty} \frac{(k+i)!}{k!} A^k B B^T A^{kT} \qquad \text{and} \qquad Q_i \triangleq \sum_{k=0}^{\infty} \frac{(k+i)!}{k!} A^{kT} C^T C A^k \; .$$

This definition of the matrices X_i results in simpler recurrence equations.
It is easily seen that $A_1 X_0 A_2 - X_0 + D = 0$, and for $i > 0$ one has :

$$\begin{aligned} A_1 X_i A_2 + i X_{i-1} &= \sum_{k=0}^{\infty} \frac{(k+i)!}{k!} A_1^{k+1} D A_2^{k+1} + \sum_{k=0}^{\infty} \frac{(k+i-1)! i}{k!} A_1^k D A_2^k \\ &= \sum_{k=1}^{\infty} \frac{(k+i-1)! k}{k!} A_1^k D A_2^k \\ &\qquad + i! D + \sum_{k=1}^{\infty} \frac{(k+i-1)! i}{k!} A_1^k D A_2^k \\ &= \sum_{k=0}^{\infty} \frac{(k+i-1)!(k+i)}{k!} A_1^k D A_2^k \; = \; X_i \; , \end{aligned}$$

hence
$$A_1 X_i A_2 - X_i + i X_{i-1} \; = \; 0 \; . \tag{7.7}$$

Note that $\frac{(k+i)!}{k!} = \sum_{j=0}^{i} f_{ij} k^j$, where the coefficients f_{ij} are defined recursively with

$$f_{i0} \triangleq i! \; , f_{ii} \triangleq 1 \; , \quad \text{and the recursion} \quad f_{ij} \triangleq f_{i-1,j-1} + i f_{i-1,j} \; .$$

Also, $k^i = \sum_{j=0}^{i} g_{ij} \frac{(k+j)!}{k!}$, and hence $\sum_{k=0}^{\infty} k^i A_1^k D A_2^k = \sum_{j=0}^{i} g_{ij} X_j$, where again the coefficients g_{ij} are defined recursively using

$$g_{i0} \triangleq (-1)^i \; , g_{ii} \triangleq 1 \; , \quad \text{and} \quad g_{ij} \triangleq g_{i-1,j-1} - (j+1) g_{i-1,j} \; .$$

Some of these coefficients are tabulated in Table 7.1.

Thus, a segment of a gramian weighted with a piecewise polynomial is computed using the following formula :

$$\begin{aligned} X_i(k_1,k_2) &\triangleq \sum_{k=k_1}^{k_2} k^i A_1^k D A_2^k \\ &= A_1^{k_1} \left(\sum_{k=0}^{\infty} (k+k_1)^i A_1^k D A_2^k \right) A_2^{k_1} \\ &\qquad - A_1^{k_2+1} \left(\sum_{k=0}^{\infty} (k+k_2+1)^i A_1^k D A_2^k \right) A_2^{k_2+1} \end{aligned}$$

i \ j	0	1	2	3	4	5	6
0	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0
2	2	3	1	0	0	0	0
3	6	11	6	1	0	0	0
4	24	50	35	10	1	0	0
5	120	274	225	85	15	1	0
6	720	1764	1624	735	175	21	1

i \ j	0	1	2	3	4	5	6
0	1	0	0	0	0	0	0
1	-1	1	0	0	0	0	0
2	1	-3	1	0	0	0	0
3	-1	7	-6	1	0	0	0
4	1	-15	25	-10	1	0	0
5	-1	31	-90	65	-15	1	0
6	1	-63	301	-350	140	-21	1

Table 7.1: Coefficients f_{ij} (top) and g_{ij} (bottom)

$$= A_1^{k_1} \sum_{l=0}^{i} \left(\sum_{j=l}^{i} C_i^j k_1^{i-j} g_{jl} \right) X_l A_2^{k_1}$$

$$- A_1^{k_2+1} \sum_{l=0}^{i} \left(\sum_{j=l}^{i} C_i^j (k_2+1)^{i-j} g_{jl} X_l \right) A_2^{k_2+1} \,,$$

$$\text{where } C_i^j \triangleq \frac{i!}{j!(i-j)!} \,.$$

Hence, if $f(t)$ (or $f(k)$ for a discrete-time system) is a piecewise polynomial function, P_f and Q_f are readily computed using the techniques given above.

7.4.3 Time weighted norms

The gramians P_f and Q_f can be used for computing the time weighted H_2 norm $\|H\|_f$, where $H \triangleq [A, B, C, 0]$. Indeed,

$$\begin{aligned} \|H(s)\|_f^2 &= \text{Tr} \int_0^\infty f(t) B^T e^{A^T t} C^T C e^{At} B dt \\ &= \text{Tr } B^T (\int_0^\infty f(t) e^{A^T t} C^T C e^{At} dt) B \\ &= \text{Tr } (B^T Q_f B) \,. \end{aligned}$$

Analogously, one shows that $\|H(s)\|_f^2 = \text{Tr } (C P_f C^T)$.

The proof that $\|H(z)\|_f^2 = \text{Tr } (C P_f C^T) = \text{Tr } (B^T Q_f B)$ holds for discrete-time systems too is quite as trivial.

7.4.4 Model reduction

It is easily verified that if a similarity transformation mapping $[A, B, C, D]$ into $[TAT^{-1}, TB, CT^{-1}, D]$ is applied to the realization of a system, the gramians P_i and Q_i transform into TP_iT^T and $T^{-T}Q_iT^{-1}$, respectively. From there it is quite as easily seen that this can be said of P_f and Q_f as well.

The standard balancing transformation, partitioning and truncating of the transformed state space matrices according to the largest diagonal values, yields a state space model for the reduced order system. Note that this scheme is quite heuristic, and does not provide much more than a candidate approximant, aiming at making the time weighted reduction error small.

We certainly make no claims with respect to optimality or error bounds. The key thing is that we provide a tool with which one may shape the error function in the time domain; but a lot of tuning and tweaking of $f(t)$ to obtain a desired effect may be necessary.

Of course the reduction may also be accomplished using Safonov's balancing-free method [234] (see p. 52).

7.4.5 Relation to other model reduction methods

The method and techniques proposed in this section bear some affinity to the weighted impulse response gramian model reduction of Sreeram and Agathoklis [254]. Indeed, in [254], Lyapunov equations of the same type as (7.7) appear; however, there these equations are solved for the system in controllability canonical form, and the observability gramians Q_i are then system invariants. The model reduction proposed by Sreeram and Agathoklis is achieved by retaining the states corresponding to the largest eigenvalues of Q_i, which is equivalent to balanced truncation using Q_i and the identity matrix I for gramians. This identity matrix is in general not equal to the time weighted controllability gramian of the system in controllability canonical form, and therefore the method is not equivalent with the one proposed here.

7.4.6 Stability of the reduced-order models

We shall show in this subsection that if, for a continuous-time system, $f(t) = \sum_{i=1}^{m} f_i t^i$, with $f(t) > 0$ and $f'(t) > 0 \; \forall \, t > 0$, is used as a weighting function, and the original system is controllable and asymptotically stable, then the reduced-order model is asymptotically stable as well. Indeed, for $f(t) = 1$, this is a standard result on balanced truncation [216], hence also for $f(t) = f_0$. It is also well known that P_0 is positive definite under the assumptions above.
Since $AP_i + P_i A^T + iP_{i-1} = 0$, by the Lyapunov criterion [153], $P_i > 0$ for all $i > 0$.
Also, we have $P_f = \sum_{i=0}^{m} f_i P_i$, and

$$-AP_f - P_f A^T = f_0 BB^T + \sum_{i=1}^{m} f_i i P_{i-1} = f_0 BB^T + P_{f'} > 0 \; .$$

Therefore, again by the Lyapunov criterion, $P_f > 0$. By applying a balancing similarity transformation, P_f is transformed into a diagonal matrix Σ, and $P_{f'}$ into a matrix $TP_{f'}T^T \triangleq P'_f$ which is of course still symmetric positive definite.

Hence

$$-\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}-\begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^T = \begin{bmatrix} P'_{f11} & P'_{f12} \\ P'^{T}_{f12} & P'_{f22} \end{bmatrix}$$

and

$$-A_{11}\Sigma_1 - \Sigma_1 A_{11}^T = P'_{f11} > 0,$$

so the Lyapunov criterion once more assures that A_{11} has only eigenvalues with negative real part.

7.5 Example

To illustrate the effectiveness of these methods with respect to time weighted error norms, consider the following academic example, where we wish to obtain a reduced order system with a small error in the Hilbert-Schmidt-Hankel norm. Consider the following continuous-time system :

$$G(s) = \frac{s^4 + 2s^3 + 6s^2 + 18s + 31}{s^5 + 7s^4 + 22s^3 + 36s^2 + 24s + 3}$$

which is to be reduced to order $r = 2$.

Unweighted balanced truncation renders the reduced order model :

$$G_b(s) = \frac{0.0651s + 1.1008}{s^2 + 0.7606s + 0.1092}, \qquad \text{with } \|G(s) - G_b(s)\|_{HSH} = 0.5142.$$

Time domain weighted balanced truncation with weighting $f(t) = t$ yields :

$$G_{hb}(s) = \frac{-0.8478s + 2.0563}{s^2 + 1.4221s + 0.2015}, \qquad \text{with } \|G(s) - G_{hb}(s)\|_{HSH} = 0.2618.$$

The optimal approximant with respect to the HSH norm, found using Alg. 7.2, is :

$$G_{ho}(s) = \frac{0.1954s^2 - 0.4907s + 1.8053}{s^2 + 1.2463s + 0.1749}, \qquad \text{with } \|G(s) - G_{ho}(s)\|_{HSH} = 0.2207.$$

Fig. 7.1 (a) shows the magnitude Bode plots of $G(s)$, $G_b(s)$, $G_{hb}(s)$ and $G_{ho}(s)$ and the errors $|G(j\omega) - G_b(j\omega)|$, $|G(j\omega) - G_{hb}(j\omega)|$, and $|G(j\omega) - G_{ho}(j\omega)|$, are plotted in Fig. 7.1 (b).

Fig. 7.2 ($r = 2$) shows the convergence of $\|\hat{a}^{[k]} - \hat{a}\|_2$; the full line is obtained when $S_i^{[k]}$ is computed, the dashed line when this computation is omitted, i.e.

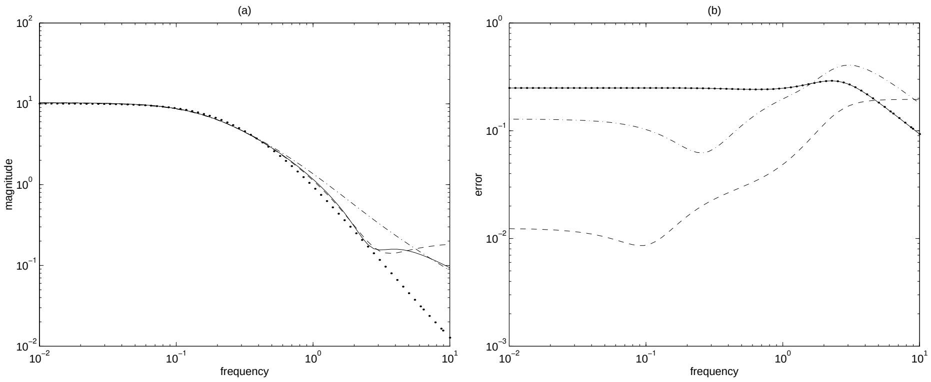


Figure 7.1: (a) Magnitude Bode plots.
Full line : original system, dashed line : optimal HSH reduction,
dash-dotted line: time weighted BT , dotted line : BT .
(b) Error magnitude Bode plots.
Full line : BT, dashed line: optimal HSH reduction,
dash-dotted line : time weighted BT .

when $S_i^{[k]}$ is put to zero. The fact that the algorithm converges much sooner when $S_i^{[k]} = 0$ is not typical; see e.g. Fig. 7.2 ($r = 3$), where the same plot is given for $r = 3$.

Note that there is no guarantee that the HSH error resulting from this time weighted balanced truncation will be smaller than the error with ordinary balanced truncation; not surprisingly, it is possible to find examples where the opposite is true. Yet intuitively it is clear that both optimal HSH model reduction and time weighted balanced truncation with weighting function $f(t) = t$ will put greater emphasis on the steady-state response and less on the immediate response of the system.
To compare time weighted balanced truncation to the H_2-optimal time weighted procedure Alg. 7.1, consider the discrete-time 5th order system

$$G(z) \triangleq \left[\begin{array}{ccccc|c} 1 & 0.1 & 0 & 0 & 0 & 0.3 \\ -0.1 & 0.8 & 0.1 & -0.1 & 0 & 0.4 \\ 0 & 0.1 & 0.8 & 0.4 & 0 & -0.1 \\ 0 & 0.1 & -0.4 & 0.6 & -0.1 & -0.1 \\ 0 & 0 & 0 & 0.1 & 0.4 & 0 \\ \hline 0.3 & -0.4 & 0.1 & -0.1 & 0 & -0.1 \end{array}\right].$$

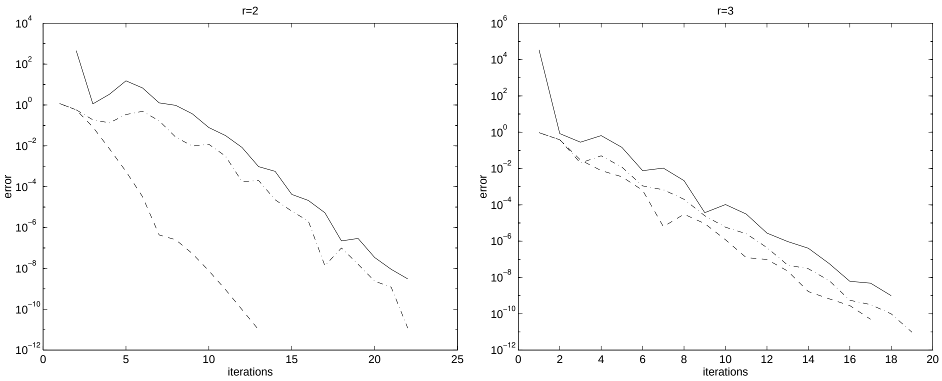


Figure 7.2: Convergence of $\|\hat{a}^{[k]} - \hat{a}\|_2$. Left : $r = 2$, right : $r = 3$.
Dash-dotted line : with $S_i^{[k]}$ computed, dashed line : with $S_i^{[k]} = 0$.
The full line shows $\frac{S_i^{[k]}}{T_i^{[k]}}$.

The impulse and step response of $G(z)$ are plotted in Fig. 7.3.

Suppose one wants an approximant of order $r = 2$ which captures the undershoot behavior of the system well; this can be achieved by trying to keep $\|G - G_r\|_f$ small, with $f_k \triangleq 100$ for $1 \leq k \leq 10$ and $f_k \triangleq 1$ for $k > 10$.

Unweighted balanced truncation gives :

$$G_b(z) = \frac{-0.1z^2 + 0.1216z + 0.0156}{z^2 - 1.8307z + 0.8413}\,, \qquad \text{with } \|G(z) - G_b(z)\|_f = 0.0317\,.$$

Time domain weighted balanced truncation results in :

$$G_{tb}(z) = \frac{-0.1z^2 + 0.0967z + 0.048}{z^2 - 1.7582z + 0.7659}\,, \qquad \text{with } \|G(z) - G_{tb}(z)\|_f = 0.0346\,.$$

Application of Alg. 7.1 with $p = 100$ yields :

$$G_{to}(z) = \frac{-0.1z^2 + 0.1056z + 0.0366}{z^2 - 1.7942z + 0.8064}\,, \qquad \text{with } \|G(z) - G_{to}(z)\|_f = 0.0189\,.$$

Fig. 7.4 shows the absolute error of the impulse response $|g(t) - g_r(t)|$ and of the step response $|s(t) - s_r(t)|$ for $G_b(z)$, $G_{tb}(z)$ and $G_{to}(z)$.

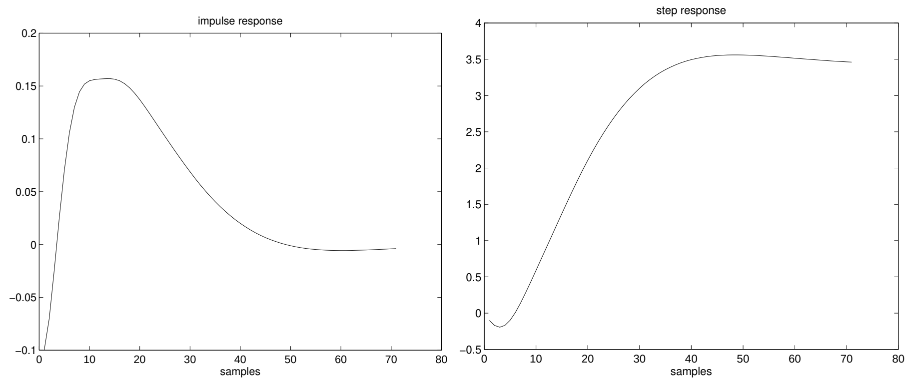


Figure 7.3: Impulse response and step response of $G(z)$.

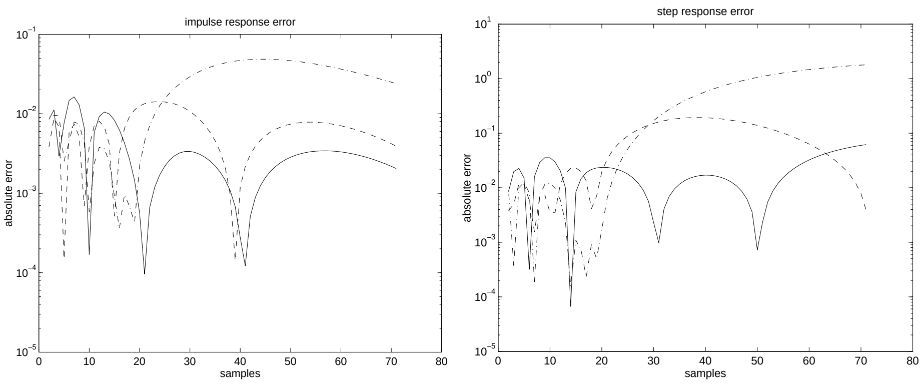


Figure 7.4: Absolute error for impulse and step response.
Full line : unweighted balanced truncation, dash-dotted line : time weighted balanced truncation, dashed line : time weighted H_2-optimal approximant.

7.6 Conclusions

The STLS framework presented in Chapter 5 is shown to be useful for SISO time weighted H_2 model reduction. A quite obvious generalization of the iterative Algorithm 5.3 is put forward for the finite time horizon problem; a similar generalization of the z-domain Algorithm 5.8 is less trivial, but for the special case of Hilbert-Schmidt-Hankel model reduction, a z-domain algorithm similar to Alg. 6.1 is presented.
For the latter problem, necessary conditions and an expression for the gradient of the object function are also derived.

We have introduced time weighted gramians, useful for computing time weighted norms. Furthermore, we advance a quite different, heuristic technique, based on balancing of these time weighted gramians, which may be used to yield an initial model for the these iterative methods. For certain cases, this technique is proved to preserve the stability of the model.

Chapter 8

Conclusions

8.1 Contributions of the thesis

This dissertation gives a number of contributions to the field of controller reduction, by providing some new frequency weighted or time-domain weighted model reduction procedures, and showing equivalences of certain reduction algorithms.

1. It is proven that Enns's Frequency Weighted Balanced Truncation (FWBT) [93] with certain performance-oriented frequency weights is equivalent to Closed Loop Balanced Truncation (CLBT) [302]; furthermore, it is shown that the computations for CLBT with observer-based controllers are structured, so that they can be performed in a faster way. A similar thing is done for FWBT with a stability-oriented weight. Also, for H_2-optimal controllers, CLBT is proven to be equivalent to BCRAM [312].

2. FWBT with the inverse of the outer factor of the system to be reduced as a weight is shown to be a competitive relative error approximation method; the algebraic Riccati equation that needs to be solved for this is also useful when computing a Balanced Stochastic Truncation (BST).

3. A novel iterative algorithm is given for H_2-optimal model reduction. The algorithms given in this thesis are mathematically equivalent to the one put forward in [73], but are simpler, and lead to better conditioned linear systems to be solved each iteration.

4. An extension for frequency weighted H_2 approximation is presented.

5. The same type of frequency weighted algorithm is developed for the Hilbert-Schmidt-Hankel norm.

Also, necessary conditions for optimal Hilbert-Schmidt-Hankel norm approximation are derived.

6. Two time-domain weighted approximation procedures are proposed, one which is optimal with respect to a finite-horizon H_2-norm, and a suboptimal, heuristic one based on balancing.

8.2 Open questions

1. In Chapters 3 and 4 and in Section 6.4 we demonstrated that for some general-purpose frequency weighted approximation methods, structure may be exploited when using certain frequency weights, which are precisely the ones that are most useful for controller reduction, such as the inverse, the stability-oriented weight W_s of Subsection 2.4.1, or the performance-oriented weights $W_{p,l}$ and $W_{p,r}$ of Subsection 2.4.2. It may be a fruitful idea to look for other frequency weighted algorithms where savings of this type could be realized.

2. The reduced controller C_r produced by CLBT performs very well in many cases, but especially when the reduced order is chosen so small that C_r deviates drastically from the original controller C even in a critical frequency range, the method seems to be inferior to other reduction schemes. The conditions under which CLBT is a good option deserve more attention. Clearly, an easy pre-computable bound on $\|H_{zw}(C_r) - H_{zw}(C)\|_\infty$ or $\|H_{zw}(C_r) - H_{zw}(C)\|_2$ would be a useful tool to determine those conditions; however, we fear that such a bound will prove elusive.
   For instance, a bound on $\|H_{zw}(C_r) - H_{zw}(C)\|_\infty$, applied to the standard plant $\begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix} \triangleq \begin{bmatrix} W_o G W_i & -W_o \\ W_i & 0 \end{bmatrix}$ of p. 33, would yield an H_∞ error bound for Enns's FWBT, which eluded a lot of research effort already in the past decade.

3. Closely connected to the previous question is the issue of the meaning of the 'generalized' Hankel singular values in more intuitive balancing-type algorithms such as FWBT, CLBT, IOWBT, etc. One application where this is of special concern is the Closed-Loop Balanced Truncation of decentralized controllers as in Section 3.8, where we proposed using a set of generalized Hankel singular values for each sub-controller to determine the order to which each sub-controller should be reduced.

4. As IOWBT is just a special instance of FWBT, one may suspect that an H_∞ error bound is just as hard to obtain. However, the fact that the approximation error of IOWBT and of BST are generally very close together

indicates that this is not necessarily the case. The connection between IOWBT and BST may be deeper than just the algebraic Riccati equation without a constant term; for instance, both amount to a Enns's method with (different) spectral factors of the inverse system. Our feeling is that this matter is not fully explored, yet no conclusive statements concerning this relationship could be made in this work.

5. The convergence of the STLS algorithms of Chapters 5, 6 and 7, is of course one major unresolved question. Both local and global convergence are open questions. Globally convergent continuous-time algorithms may be constructed [71], yet this result cannot be used to prove convergence of the algorithms presented here. In order to do so for one of our algorithms, one would have to find a continuous-time algorithm of which our algorithm is a discretization, prove that the continuous-time algorithm is globally convergent, and prove that the convergence is not lost in the discretization step. The chances that this is possible are dim.
   Of course we do not mean to imply by this that continuous-time algorithms cannot lead to useful discrete counterparts. One could hope that one day a globally convergent algorithm would be found, which lends itself to a translation to the z-domain in the way the algorithms in this thesis do; but at present, no such results have been obtained.

6. A separate question is of course whether the global minimum is found. This is certainly not always so for the STLS algorithm presented here : examples have been found, for which a suboptimal local minimum is reached starting from certain initial conditions. Yet the overall behavior of the algorithm seems to be superior to gradient-type descent algorithms; when and why this is the case, has to be examined in more detail.

7. The most salient shortcoming of the STLS-type approximation algorithms is the fact that they cannot deal with 'true' MIMO systems, i.e. MIMO but not SIMO or MISO. As we have stated in Section 5.3, a simple-minded extension of the STLS algorithms would result in approximants whose partial fraction representation has full rank residues $\hat{H}_i$, i.e. whose r poles have multiplicity greater than one. How to adapt the STLS framework to handle this as well is worth a closer look, but this will certainly require the algorithms to be changed in a nontrivial way.

8. The time weighted model reduction methods of Sections 7.2 and 7.4 have been merely offered as tools, but with very few application notes. When time weighting can be useful to obtain the desired closed-loop behavior after reduction, and how to design the time weights, is a matter that is still unclear.

Appendix A

Proofs and derivations

A.1 Taking the derivative of the Lagrangian

(See p. 174.)

Taking the derivative w.r.t. l, y and λ is straightforward.

It's also easily seen that $\quad \dfrac{\partial}{\partial \hat{h}_i}\left(l^T \hat{H} y\right) = [l(z)y(z)]_i \ .$

What remains to be computed is

$$
\begin{aligned}
&\frac{\partial}{\partial \hat{h}_i}\left(\frac{1}{2}\sum_{i=1}^{\infty}\left([w \star h]_i - \left[w \star \hat{h}\right]_i\right)^2\right) \\
&\quad = \frac{\partial}{\partial \hat{h}_i}\left(\frac{1}{4j\pi}\oint_T z^{-1} w(z^{-1}) w(z)\left(h(z^{-1}) - \hat{h}(z^{-1})\right)\left(h(z) - \hat{h}(z)\right) dz\right) \\
&\quad = -\frac{1}{4j\pi}\oint_T z^{i-1} w(z^{-1}) w(z)\left(h(z) - \hat{h}(z)\right) dz \\
&\qquad\qquad + \frac{1}{4j\pi}\oint_T z^{-i-1} w(z^{-1}) w(z)\left(h(z^{-1}) - \hat{h}(z^{-1})\right) dz \\
&\quad = -\frac{1}{2j\pi}\oint_T z^{i-1} w(z^{-1}) w(z)\left(h(z) - \hat{h}(z)\right) dz \\
&\quad = -\left[w(z^{-1}) w(z)\left(h(z) - \hat{h}(z)\right)\right]_i \ ,
\end{aligned}
$$

hence : $\quad -\left[w(z^{-1}) w(z)\left(h(z) - \hat{h}(z)\right)\right]_+ + l(z)y(z) = 0 \ .$

From this we can derive a generalized orthogonality condition as in [73, De Moor 1993a] :

$$\begin{aligned}
&\frac{1}{2j\pi}\oint_T z^{-1}\hat{h}(z^{-1})w(z^{-1})w(z)\left(h(z)-\hat{h}(z)\right)dz \\
&\quad = \frac{1}{2j\pi}\sum_{i=1}^{\infty}\hat{h}_i\oint_T z^{i-1}w(z^{-1})w(z)\left(h(z)-\hat{h}(z)\right)dz \\
&\quad = \sum_{i=1}^{\infty}\hat{h}_i\left[l(z)y(z)\right]_i = \left[\hat{h}(z)l(z^{-1})y(z^{-1})\right]_0 \\
&\quad = \left[\left[\hat{h}(z)y(z^{-1})\right]_+ l(z^{-1})\right]_0 = 0
\end{aligned}$$

$$\Downarrow$$

$$\oint_T z^{-1}\hat{h}(z^{-1})w(z^{-1})w(z)\left(h(z)-\hat{h}(z)\right)dz = 0$$

$$\Updownarrow$$

$$\sum_{i=1}^{\infty}\left(w\star\hat{h}\right)_i\left(w\star(h-\hat{h})\right)_i = 0$$

A.2 Lemmas

Lemma 6.1

$$\text{If} \qquad x(z)\in L_2^+,\ w(z),\ w^{-1}(z)\in RL_2^{0+}$$

$$\text{Then} \quad \left[w^{-1}(z^{-1})\left[w(z^{-1})x(z)\right]_+\right]_+ = x(z)\ ,\forall z\ s.t.\ |z|>1$$

Proof :
Since $x(z)\in L_2^+$, it converges in $\{z|\ 1<|z|\}$; since $w(z), w^{-1}(z)\in RL_2^{0+}$, $\exists\beta>1$ such that $w(z^{-1})$, $w^{-1}(z^{-1})$ converge in $\{z|\ |z|<\beta\}$.
$w(z^{-1})$ is bounded on T, hence $w(z^{-1})x(z)\in L_2$ converges in $\{z|\ 1<|z|<\beta\}$. Also $w(z^{-1})x(z) = q(z)+l(z^{-1}) \quad (1<|z|<\beta)$, where $q(z)\in L_2^+$, and $l(z)\in L_2^{0+}$. Therefore $w^{-1}(z^{-1})q(z) = x(z) - w^{-1}(z^{-1})l(z^{-1}) \qquad (1<|z|<\beta)$, and $\left[w^{-1}(z^{-1})q(z)\right]_+ = \left[w^{-1}(z^{-1})\left[w(z^{-1})x(z)\right]_+\right]_+ = x(z)$, whenever $|z|>1$. $\square$

Lemma 6.2

$$\left.\begin{array}{l} x(z) \in L_2^+ \\ n_s(z),\;\; d_s(z) \text{ `stable' polynomials} \\ n_a(z),\;\; d_a(z) \text{ `anti-stable' polynomials} \\ \left[\frac{n_s(z)n_a(z)}{d_s(z)d_a(z)}x(z)\right]_+ = \frac{y(z)}{a(z)} \in L_2^+ \\ \text{no common roots in } n_s(z)a(z) \text{ and } d_s(z) \end{array}\right\} \;\Rightarrow\;\; x(z) = \frac{d_s(z)f(z)}{n_s(z)a(z)}$$

where $f(z)$ *is a polynomial with* $\deg f(z) < \deg n_s(z) + \deg a(z) - \deg d_s(z)$

Proof :

- There is a $e(z) \in L_2^{0+}$ s.t.

$$\frac{n_s(z)n_a(z)}{d_s(z)d_a(z)}x(z) = \frac{y(z)}{a(z)} + e(z^{-1}) \tag{A.1}$$

  Multiplying this expression with $(z-\sigma_i)$ (where σ_i is a root of $d_s(z)$, with multiplicity m_i), and letting $z \to \sigma_i$, yields :

$$\frac{n_s(\sigma_i)n_a(\sigma_i)}{d_a(\sigma_i)} \lim_{z\to\sigma_i} \frac{x(z)}{(z-\sigma_i)^{m_i-1}} \lim_{z\to\sigma_i} \frac{(z-\sigma_i)^{m_i}}{d_s(z)} = 0$$

  Here $\lim_{z\to\sigma_i} \frac{x(z)}{(z-\sigma_i)^{m_i-1}}$ is the only quantity which can be zero, therefore σ_i is an m_i-multiple zero of $x(z)$.

- Suppose $x(z)$ has a stable pole p_i with a multiplicity m_i which is greater than its multiplicity as a root of $n_s(z)a(z)$, then multiplication of (A.1) by $(z-p_i)^{m_i}$ and letting $z \to p_i$ yields a nonzero left hand side, whereas the right hand side vanishes, therefore $x(z)$ has no poles except the roots of $n_s(z)a(z)$, and the multiplicity of its poles is maximally their multiplicity as a root of $n_s(z)a(z)$. □

Lemma 7.1

$$\left[w(z)l(z^{-1})\log|1-az^{-1}|\right]_s = \left[\int z^{-1}\left[z^{-s}w(z^{-1})l(z)\right]_+ dz\right]_{z=a^{-1}}$$

Proof :

- $\left[w(z)l(z^{-1})\log|1-az^{-1}|\right]_s$
  Define $f(z) \triangleq w(z)\log|1-az^{-1}| = \sum_i f_i z^{-i}$

$$\Rightarrow f_i = -\sum_{j=1}^{i} w_{i-j}\frac{a^j}{j}$$

$$\Rightarrow \left[f(z)l(z^{-1})\right]_s = \left[\sum_{i=1}^{\infty} f_i z^{-i} \sum_{j=1}^{\infty} l_j z^j\right]_s = \sum_{i=s+1}^{\infty} f_i l_{i-s}$$

$$= -\sum_{i=s+1}^{\infty}\sum_{j=1}^{i} w_{i-j}\frac{a^j}{j} l_{i-s}$$

$$= -\sum_{j=1}^{\infty}\sum_{i=\max(j,s+1)}^{\infty} w_{i-j}\frac{a^j}{j} l_{i-s}$$

- $\left[\int z^{-1}\left[z^{-s}w(z^{-1})l(z)\right]_+ dz\right]_{z=a^{-1}}$

$$\left[z^{-s}w(z^{-1})l(z)\right]_+ = \sum_{r=0}^{\infty} w_r z^r \sum_{i=\max(1,r+1-s)}^{\infty} l_i z^{-i-s}$$

$$= \sum_{j=1}^{\infty}\sum_{i=\max(j-s,1)}^{\infty} w_{i+s-j} l_i z^{-j} = \sum_{j=1}^{\infty}\sum_{i=\max(j,s+1)}^{\infty} w_{i-j} l_{i-s} z^{-j}$$

$$\Rightarrow \int z^{-1}\left[z^{-s}w(z^{-1})l(z)\right]_+ dz|_{z=a^{-1}} = -\sum_{j=1}^{\infty}\sum_{i=\max(j,s+1)}^{\infty} w_{i-j} l_{i-s}\frac{a^j}{j} \quad \square$$

Lemma A.1

Let $|\alpha| < 1$, $x(z), y(z) \in L_2^+$, *where* $y(z) = \sum_{i=1}^{l} \frac{Y_i}{(z-\alpha)^i} \in \{z \in \mathbb{C} \mid |z| > |\alpha|\}$

$$\text{Then} \quad \left[x(z^{-1})y(z)\right]_+ = \sum_{i=1}^{l}\frac{C_i}{(z-\alpha)^i} \; \textit{for} \; |z| > |\alpha| \; ,$$

$$C_i \triangleq \sum_{r=0}^{l-i}\frac{Y_{i+r}}{r!}\left[\frac{d^r}{dz^r}(x(z^{-1}))\right]_{z=\alpha}$$

Proof :
Expanding $x(z^{-1})$ (conv. $|z|<1$) as a Taylor series of $(z-\alpha)$, we find:

$$\begin{aligned}
x(z^{-1}) &= \sum_{i=0}^{\infty} \frac{1}{i!}\frac{d^i}{dz^i}\left(x(z^{-1})\right)_{z=\alpha}(z-\alpha)^i \qquad (|z-\alpha|<1-|\alpha|)\\
\frac{x(z^{-1})}{(z-\alpha)^s} &= \sum_{i=0}^{\infty} \frac{1}{i!}\frac{d^i}{dz^i}\left(x(z^{-1})\right)_{z=\alpha}(z-\alpha)^{i-s} \qquad (0<|z-\alpha|<1-|\alpha|)\\
\left[\frac{x(z^{-1})}{(z-\alpha)^s}\right]_+ &= \sum_{i=1}^{s} \frac{1}{(s-i)!}\frac{d^{s-i}}{dz^{s-i}}\left(x(z^{-1})\right)_{z=\alpha}\frac{1}{(z-\alpha)^i} \qquad (|z|>|\alpha|)\\
&\Downarrow\\
\left[x(z^{-1})y(z)\right]_+ &= \sum_{s=1}^{l} Y_s \sum_{i=1}^{s} \frac{1}{(s-i)!}\frac{d^{s-i}}{dz^{s-i}}\left(x(z^{-1})\right)_{z=\alpha}\frac{1}{(z-\alpha)^i}\\
&= \sum_{i=1}^{l}\sum_{r=0}^{l-i} \frac{Y_{r+i}}{r!}\frac{d^r}{dz^r}\left(x(z^{-1})\right)_{z=\alpha}\frac{1}{(z-\alpha)^i} \qquad \Box
\end{aligned}$$

Lemma A.2

$$\left[\frac{1}{(z-\alpha)^s}\right]_i = \alpha^{i-s}C_{s-1}^{i-1}$$

Proof :

$$\text{Let's first prove : } \frac{1}{(1-x)^s} = \sum_{i=0}^{\infty} C_{i+s-1}^{s-1}x^i$$

Indeed, it holds for $s=1$, and if it holds for $s-1$, then it holds for s :

$$\frac{1}{(1-x)^s} = \frac{1}{s-1}\frac{d}{dx}\left(\frac{1}{(1-x)^{s-1}}\right) = \sum_{i=1}^{\infty}\frac{i}{s-1}C_{i+s-2}^{s-2}x^{i-1} = \sum_{i=0}^{\infty} C_{i+s-1}^{s-1}x^i$$

$$\Rightarrow \quad \frac{1}{(z-\alpha)^s} = \frac{z^{-s}}{(1-\alpha z^{-1})^s} = \sum_{i=0}^{\infty} C_{i+s-1}^{s-1}\alpha^i z^{-i-s} = \sum_{i=s}^{\infty} C_{i-1}^{s-1}\alpha^{i-s}z^{-i} \qquad \Box$$

Lemma A.3

$$\text{Let} \qquad h(z) \triangleq \sum_{i=1}^{n}\frac{H_i}{z-\alpha_i} \in RL_2^+$$

$$\text{Then} \qquad \|h(z)\|_2^2 = \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{H_iH_j}{1-\alpha_i\alpha_j}$$

$$\textit{and} \qquad \|h(z)\|_{HSH}^2 = \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{H_iH_j}{(1-\alpha_i\alpha_j)^2}$$

Proof :

$$\begin{aligned}
\|h(z)\|_2^2 &= \frac{1}{2j\pi}\oint_T z^{-1}h(z^{-1})h(z)dz \\
&= \sum_i\sum_j H_iH_j\frac{1}{2j\pi}\oint_T z^{-1}\frac{1}{(z^{-1}-\alpha_i)(z-\alpha_j)}dz \\
&= \sum_i\sum_j \frac{H_iH_j}{1-\alpha_i\alpha_j}
\end{aligned}$$

$$\begin{aligned}
\|h(z)\|_{HSH}^2 &= \frac{1}{2j\pi}\oint_T h(z^{-1})\frac{d}{dz}h(z)dz \\
&= \sum_i\sum_j H_iH_j\frac{1}{2j\pi}\oint_T \frac{1}{z^{-1}-\alpha_i}\frac{d}{dz}\left(\frac{1}{z-\alpha_j}\right)dz \\
&= \sum_i\sum_j \frac{H_iH_j}{(1-\alpha_i\alpha_j)^2} \qquad \square
\end{aligned}$$

Bibliography

[1] T. J. S. Abrahamsson, "Constrained Minimization of Vibrational Magnitudes Using a Modal Reduction Method.," *Int. J. for Num. Methods in Eng.*, 1991, vol. 31, no. 8, pp. 1585.

[2] M. Abramowitz and I. Stegun, "Handbook of Mathematical Functions," *Dover Publ.*, 1968, pp. 227-237.

[3] N. I. Achieser, "Theory of Approximation," *Frederick Ungar Publishing Co., New York*, 1956, also *Dover Publications, Inc.*, 1992.

[4] V. M. Adamjan, D. V. Arov, and M. G. Krein, "Analytic properties of Schmidt pairs for a Hankel operator and the generalized Schmidt-Takagi problem," *Math. USSR Sbornik*, 1971, vol. 15, pp. 31-73.

[5] D. Aeyels and J. L. Willems, "Output Feedback: Periodic and Memoryless as an Alternative for Time-Invariant and Dynamic," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992.

[6] P. Agathoklis and V. Sreeram, "System identification and model reduction using weighted impulse response gramians," *Int. J. Systems Science*, 1990, vol. 21, pp. 1541-1552.

[7] P. Agathoklis and V. Sreeram, "Truncation criteria for model reduction using balanced realizations," *Electronic Letters*, 1988, vol. 24, pp. 837-838.

[8] P. Agathoklis and V. Sreeram, "Lower bounds for the stability margin of linear continuous systems based on the weighted performance criteria," *Proc. IEE*, 1989, vol. 136, pp. 63-67.

[9] P. Agathoklis and V. Sreeram, "System identification and model reduction of linear continuous systems from impulse response data," *Proc. 26th Allerton Conf., Monticello, Illinois*, 1988, pp. 199-208.

[10] L. A. Aguirre, "The least squares Padé method for model reduction," *Int. J. Systems Science*, 1992, vol. 23, no. 10, pp. 1559-1570.

[11] H. Akaike, "Markovian Representation of Stochastic Processes by Canonical Variables," *SIAM J. Control*, 1975, vol. 13, no. 1, pp. 162-173.

[12] R. W. Aldaheri, "Model order reduction via real Schur-form decomposition," *Int. J. Control*, 1991, vol. 53, no. 3, pp. 709-716.

[13] U. M. Al-Saggaf and G. F. Franklin, "An error bound for a discrete reduced order model of a linear multivariable system," *IEEE Trans. Aut. Control*, 1987, vol. 32, no. 9, pp. 815-819.

[14] U. M. Al-Saggaf and G. F. Franklin, "Model reduction via balanced realizations: An extension and frequency weighting techniques," *IEEE Trans. Aut. Control*, 1988, vol. 33, no. 7, pp. 687-692.

[15] U. M. Al-Saggaf and G. F. Franklin, "On model reduction," *Proc. 25th Conf. Decision Control, Athens, Greece*, 1986, pp. 1064-1069.

[16] B. D. O. Anderson, "An algebraic solution to the spectral factorization problem," *IEEE Trans. Aut. Control*, 1967, vol. 12, no. 4, pp. 410-414.

[17] B. D. O. Anderson, "Weighted Hankel-norm approximation: calculation of bounds," *Syst. & Contr. Lett.*, 1986, vol. 7, pp. 247-255.

[18] B. D. O. Anderson and J. B. Moore, "Linear Optimal Control," (Chapter 10), *Prentice-Hall*, 1971.

[19] B. D. O. Anderson and S. Vongpanitlerd, "Network Analysis and Synthesis: A Modern Systems Theory Approach," *Prentice-Hall*, 1973.

[20] B. D. O. Anderson and J. B. Moore, "Optimal Control - Linear Quadratic Methods," (Chapter 10), *Prentice-Hall*, 1989.

[21] B. D. O. Anderson and P. C. Parks, "Lumped approximation of distributed systems and controllability questions," *IEE Proc.-D*, 1985, vol. 132, no. 3, pp. 89-94.

[22] B. D. O. Anderson and R. E. Skelton, "The generation of all q-Markov Covers," *IEEE Trans. Circuits and Systems*, 1988, vol. 35, no. 4, pp. 375-384.

[23] B. D. O. Anderson and Y. Liu, "Controller Reduction : Concepts and Approaches," *IEEE Trans. Aut. Control*, 1989, vol. 34, no. 8, pp. 802-812.

[24] B. D. O. Anderson, U.-L. Ly, and Y. Liu, "Lower Order Controller Design for a High Order Plant," *Benchmark Problems for Control System Design, Report of the IFAC Theory Committee, Davison E.J., ed.*, 1990.

[25] M. Aoki, "Control of large-scale dynamic systems by aggregation," *IEEE Trans. Aut. Control*, 1968, vol. 14, no. 3, pp. 246-253.

[26] P. Apkarian, J. P. Chretien, P. Gahinet, and J. M. Biannic, "μ Synthesis by $\mathcal{D}-\mathcal{K}$ Iterations with Constant Scaling," Proc. Amer. Control Conf., San Francisco, 1993, pp. 3192-3196.

[27] J. D. Aplevich, "Approximation of discrete linear systems," *Int. J. Control*, 1973, vol. 17, no. 3, pp. 565-575.

[28] J. D. Aplevich, "Gradient methods for optimal linear systems reduction," *Int. J. Control*, 1973, vol. 18, no. 4, pp. 767-772.

[29] M. J. Balas, "Trends in large space structure control theory: Fondest hopes, wildest dreams," *IEEE Trans. Aut. Control*, 1982, vol. 27, no. 3, pp. 522-535.

[30] M. J. Balas, J. C. Doyle, K. Glover, A. Packard and R. Smith, "μ-Analysis and Synthesis Toolbox," *The Math Works Inc.*, 1993.

[31] L. Baratchart and M. Olivi, "Index of critical points in rational L_2 approximation," *Syst. & Contr. Lett.*, 1988, vol. 10, pp. 167-174.

[32] L. Baratchart and M. Olivi, "New tools in rational L_2 approximation," *Proc. IFAC Symp. Identification and System Parameter Estimation, Beijing*, 1988, vol. 2, pp. 1014-1019.

[33] L. Baratchart and M. Olivi, "Recent and new results in rational L_2 approximation," *Modeling, Robustness and Sensitivity Reduction in Control Systems, R.F. Curtain Ed.*, 1987, pp. 119-126.

[34] L. Baratchart, M. Cardelli, and M. Olivi, "Identification and rational L^2 approximation: a gradient algorithm," *Automatica*, 1991, vol. 27, pp. 413-418.

[35] L. Baratchart, M. Olivi, and F. Wielonski, "Asymptotic properties in rational L_2 approximation," *Analysis and Optimization of Systems, A. Bensoussan and J.L. Lions, eds., Springer Verlag*, 1990, vol. 144, pp. 477-486.

[36] L. Baratchart and F. Wielonski, "Rational Approximation in the Real Hardy Space H_2 and Stieltjes Integrals: A Uniqueness Theorem," *Constr. Appr.*, 1993, vol. 9, no. 1, pp. 1-21.

[37] L. Baratchart, "On the existence and generic properties of L_2 approximants for linear systems.," *IMA J. of Math. Control & Information*, 1986, vol. 3, pp. 89-101.

[38] R. H. Bartels and Stewart, G. W., "Solution of the matrix equation $AX + XB = C$," *Comm. ACM*, 1972, vol. 15, no. 9.

[39] M. S. Bazaara, H. D. Sherali, and C. M. Shetty, "Nonlinear Programming: Theory and Algorithms," *John Wiley & Sons*, 1993.

[40] J. B. Bednar, "On the approximation of FIR by IIR digital filters," *IEEE Trans. Acoustics Speech & Sign. Proc.*, 1983, vol. 31, no. 1, pp. 28-34.

[41] B. Beliczynski, I. Kale, and G. D. Cain, "Approximation of FIR by IIR digital filters: an algorithm based on balanced model reduction," *IEEE Trans. Sign. Proc.*, 1992, vol. 40, no. 3, pp. 532-541.

[42] B. Beliczynski, J. Gryka, G. D. Cain, and I. Kale, "IIR filter design via Hankel-norm approximation of FIR prototype filters: A streamlined approach," *Electr. Lett.*, 1994, vol. 30, no. 4, pp. 292-294.

[43] J. Benede, G. Leitmann, and E. Ryan, "Reduced Order Output Feedback Control of a Class of Uncertain Systems," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991.

[44] D. S. Bernstein, L. D. Davis, S. W. Greeley, and D. C. Hyland, "Numerical solution of the optimal solution/maximum entropy design equations for low-order, robust controller design," *Proc. 24th IEEE Conf. Decision and Control, Ft. Lauderdale, USA*, 1985, pp. 1795-1798.

[45] D. S. Bernstein, "Robust Static and Dynamic Output-Feedback Stabilization: Deterministic and Stochastic Perspectives," *IEEE Trans. Aut. Control*, 1987, vol. 32, no. 12, pp. 1076-1084.

[46] D. S. Bernstein, "The optimal projection equations for static and dynamic output feedback: The singular case," *IEEE Trans. Aut. Control*, 1987, vol. 32, no. 12, pp. 1139-1143.

[47] D. S. Bernstein, W. M. Haddad, "The optimal projection equations with Petersen-Hollot bounds: Robust stability and performance via fixed-order dynamic compensation for systems with structured real-value parameters uncertainty," *IEEE Trans. Aut. Control*, 1988, vol. 33, no. 6, pp. 578-582.

[48] D. S. Bernstein, W. M. Haddad, "Optimal reduced order state estimation for unstable plants," *Int. J. Control*, 1989, vol. 50, no. 4, pp. 1259-1266.

[49] D. S. Bernstein and W. M. Haddad, "Robust stability and performance via fixed-order dynamic compensation with guaranteed cost bounds," *Math. Control Signals Systems*, 1990, vol. 3, pp. 139-163.

[50] M. Bettayeb, "New interpretation of balancing state space representations as an input-output energy minimization problem.," *Int. J. Systems Sci.*, 1991, vol. 22, no. 2, pp. 325-331.

[51] D. L. Boley, e. a. , ed, "Practical Iterative Methods for Large Scale Computations," *Elsevier*, 1989.

[52] D. L. Boley, "Krylov space methods on state-space control models," *University of Minnesota, Supercomputing Institute*, 1992, no. UMSI 92/69.

[53] M. J. Bosley and F. P. Lees, "A survey of simple transfer-function derivations from high-order state-variable models," *Automatica*, 1972, vol. 8, pp. 765-775.

[54] C. P. Bottura and C. J. Munaro, "Discrete multivariable systems order reduction via Schur decomposition," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 2081-2085.

[55] S. Boyd, V. Balakrishnan, and P. Kabamba, "A bisection method for computing the H_∞ norm of a transfer matrix and related problems," *Math. Control Signals Systems*, 1989, vol. 2, pp. 207-219.

[56] S. Boyd and C. Barratt, "Linear controller design: Limits of performance," *Prentice-Hall*, 1990.

[57] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, "Linear Matrix Inequalities in System and Control Theory," *SIAM Studies in Applied Mathematics*, 1994, vol. 15.

[58] U. A. Brunner and S. Balemi, "A plausible approximation method for linear systems and its application for the design of low-order controllers," *Amer. Cont. Conf.*, 1992, pp. 800-801.

[59] A. Bultheel and P. Dewilde, "On the Adamjan-Arov-Krein approximation, identification and balanced realization of a system," *Proc. Europ. Conf. on Circuit Theory an Design*, 1980, vol. 2, pp. 186-191.

[60] A. Bultheel and M. Van Barel, "Padé techniques for model reduction in linear system theory: a survey," *J. Comp. and Applied Math.*, 1986, vol. 14, pp. 401-438.

[61] A. Bultheel and M. Van Barel, "Vector orthogonal polynomials and least squares approximation," *SIAM J. Matrix Anal. Appl.*, 1995, vol. 16, no. 3, pp. 863-885.

[62] A. Bultheel and M. Van Barel, "Euclid, Padé and Lanczos, another golden braid," *Report TW 188, Dept. Comp. Sc., K.U. Leuven*, 1993.

[63] E. Byrns, G. Sweriduk, and A. Calise, "Optimal H_2 and H_∞ Fixed Order Dynamic Compensation Using Canonical Forms," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991.

[64] R. E. Carmola and E. H. Chimowitz, "Analysis of modal reduction techniques for the dynamics of general tridiagonal systems," *Computers & chemical engineering*, 1990, vol. 14, no. 2, pp. 227.

[65] C. Ceton, P. M. R. Wortelboer, and O. H. Bosgra, "Frequency-weighted closed-loop balanced reduction," in *Proc. 2nd Eur. Control Conf.*, Groningen, The Netherlands, 1993, pp. 697-701.

[66] B.-C. Chang and J. C. Chen, "H_2 controller reduction by structured truncation," *Amer. Cont. Conf., June 2-4, San Francisco, USA*, 1993, pp. 1296.

[67] J. C. Chen, K. Zhou, and B.-C. Chang, "Closed-Loop Controller Reduction by a Structured Truncation Approach," *Proc. 33rd Conf. Decision Control, Dec. 14-16, Lake Buena Vista, USA*, 1994, pp. 2726-2731.

[68] J. A. Davis and R. E. Skelton, "Another balanced controller reduction algorithm," *Syst. & Contr. Lett.*, 1984, vol. 4, pp. 79-83.

[69] E. J. Davison, "A Method for Simplifying Linear Dynamic Systems," *IEEE Trans. Aut. Control*, 1966, vol. 11, no. 1, pp. 93.

[70] J. Dehaene, C. Yi, and B. De Moor, "Calculation of the structured singular value with gradient based optimization algorithms on a Lie group of structured unitary matrices," *ESAT-SISTA TR 94-46, Subm. to IEEE Trans. Aut. Control*, 1994, pp. 15.

[71] J. Dehaene, "Continuous-time matrix algorithms, systolic algorithms and adaptive neural networks," *Doctoral dissertation, ESAT-SISTA, Kath. Univ. Leuven*, October 1995.

[72] B. L. R. De Moor, Van Overschee P. , Schelfhout G., "H_2 model reduction for SISO systems," *IMA Preprint Series # 1035, IMA, Univ. Minnesota*, 1992.

[73] B. L. R. De Moor, Van Overschee P. , Schelfhout G., "H_2 model reduction for SISO systems," *Proc. of IFAC '93, Sydney, Australia*, 1993, vol. 2, pp. 227-230.

[74] B. L. R. De Moor, "Structured total least squares and L_2 approximation problems.," *Lin. Alg. Appl., Spec. issue on Numerical Linear Algebra in Systems & Control, (eds : Van Dooren, Ammar, Nichols, Mehrmann)*, 1993, vol. 188-189, pp. 163-207.

[75] B. De Moor, "Dynamic total linear least squares," *Proc. 10th IFAC Symposium on System Identification, July 4-6, 1994, Copenhagen, Denmark*, 1994.

[76] B. De Moor, "Total least squares for affinely structured matrices and the noisy realization problem," *IEEE Trans. Signal Proc.*, 1994, vol. 42, no. 11, pp. 3104-3113.

[77] B. De Moor and B. Roorda, "L_2-optimal linear system identification – Structured Total Least Squares for SISO Systems," *33rd IEEE Conf. Dec. Control, Lake Buena Vista, Florida*, 1994, pp. 2874.

[78] U. B. Desai and D. Pal, "A transformation approach to stochastic model reduction," *IEEE Trans. Aut. Control*, 1984, vol. 29, pp. 1097-1100.

[79] C. A. Desoer, R. W. Liu, J. Murray, and R. Saeks, "Feedback system design: The fractional representation approach to analysis and synthesis," *IEEE Trans. Aut. Control*, 1980, vol. 25, no. 3, pp. 399-412.

[80] C. De Villemagne and R. E. Skelton, "Controller Reduction Using Canonical Interactions," *IEEE Trans. Aut. Control*, 1988, vol. 33, no. 8, pp. 740-750.

[81] C. De Villemagne and R. E. Skelton, "Model reductions using a projection formulation," *Int. J. Control*, 1987, vol. 46, no. 6, pp. 2141-2169.

[82] J. C. Doyle, "Guaranteed margins for LQG regulators," *IEEE Trans. Aut. Control*, 1978, vol. 23, no. 4, pp. 756-757.

[83] J. C. Doyle and G. Stein, "Multivariable feedback design: Concepts for a classical/modern synthesis," *IEEE Trans. Aut. Control*, 1981, vol. 26, no. 1, pp. 4-16.

[84] J. C. Doyle, "Analysis of feedback systems with structured uncertainties," *IEE Proc.-D*, 1982, vol. 129, no. 6, pp. 242-250.

[85] J. C. Doyle, "Advances in Multivariable Control," *Lecture Notes at ONR/Honeywell Workshop. Minneapolis, Oct. 1984*, 1984.

[86] J. C. Doyle, K. Glover, P. P. Khargonekar and B. A. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Trans. Automat. Control*, vol. 34, no. 8, 1989, pp. 831-847.

[87] J. C. Doyle, Francis, B. A. , and Tannenbaum, "Feedback control theory," *MacMillan*, 1991.

[88] J. C. Doyle, A. Packard, and K. Zhou, "Review of LFTs, LMIs, and μ," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 1227-1232.

[89] W. Draijer, M. Steinbuch, and O. H. Bosgra, "Adaptive control of the radial servo system of a compact disc player," *Automatica*, 1992, vol. 28, no. 3, pp. 455-462.

[90] G. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrica*, 1936, vol. 1, pp. 211-218.

[91] A. Edelman, "The first annual large dense linear system survey," *Dept. Math., U.C. Berkeley*, 1991, vol. ftp math.berkeley.edu.

[92] A. Edelman, "Large dense numerical linear algebra in 1993: the parallel computing influence," *Dept. Math., U.C. Berkeley*, 1992, vol. ftp math.berkeley.edu.

[93] D. Enns, "Model reduction for control system design," *Ph. D. dissertation, Dept. AA, Stanford Univ.*, 1984.

[94] D. Enns, "Model reduction with balanced realization: an error bound and a frequency weighted generalization," in *Proc. 23rd Conf. Decision & Control*, Las Vegas, USA, 1984, pp. 127-132.

[95] K. V. Fernando and H. Nicholson, "Singular perturbational model reduction of balanced systems," *IEEE Trans. Aut. Control*, 1982, vol. 27, no. 2, pp. 466-468.

[96] K. V. Fernando and H. Nicholson, "Singular perturbational model reduction in the frequency domain," *IEEE Trans. Aut. Control*, 1982, vol. 27, no. 4, pp. 969-970.

[97] K. V. Fernando and H. Nicholson, "Singular perturbational approximations for discrete-time balanced systems," *IEEE Trans. Aut. Control*, 1983, vol. 28, no. 2, pp. 240-242.

[98] K. V. Fernando and H. Nicholson, "Reciprocal transformations in balanced model-order reduction," *IEE Proc.-D*, 1983, vol. 130, no. 6, pp. 359-362.

[99] F. D. Galiana, "On the Approximation of Multiple-Input Multiple-Output Constant Linear Systems," *Int. J. Control*, 1973, vol. 17, no. 6, pp. 1313-1324.

[100] D. Gangsaas, K. R. Bruce, J. D. Blight and U.-L. Ly, "Application of Modern Synthesis to Aircraft Control: Three Case Studies," *IEEE Trans. Aut. Control*, 1986, vol. 31, no. 11, pp. 995-1014.

[101] C. F. Gauss, "Theoria combinationis observationum erroribus minimis obnoxiae," *Commentationes Societas Regiae Scientarium Gottengensis Recentiores*, 1823, vol. 5, pp. 33-90.

[102] W. Gawronski and J.-N. Juang, "Near-Optimal Model Reduction in Balanced or Modal Coordinates," *Proc. Allerton Conf.*, 1988, pp. 209-218.

[103] W. Gawronski and J.-N. Juang, "Model reduction for flexible structures," *Control and Dynamic Systems, C.T. Leondes, ed., Academic Press*, 1990, vol. 36, pp. 143-222.

[104] Y. Ge, E. G. Collins, Jr. , L. T. Watson, and L. D. Davis, "An Input Normal Form Homotopy for the L_2 Optimal Model Order Reduction Problem," *IEEE Trans. Aut. Control*, 1994, vol. 39, no. 6, pp. 1302-1305.

[105] P. Genesio and M. Milanese, "A note on the derivation and use of reduced-order models," *IEEE Trans. Aut. Control*, 1976, pp. 118-122.

[106] Y. V. Genin and S. Y. Kung, "A two-variable approach to model reduction with Hankel norm criterion," *IEEE Trans. Circuits and Systems*, 1981, vol. 28, pp. 912-924.

[107] K. Glover, "All optimal Hankel-norm approximations of linear multivariable systems and their L_∞-error bounds," *Int. J. Control*, 1984, vol. 39, pp. 1115-1193.

[108] K. Glover, "Robust stabilization of linear multivariable systems: relations to approximation," *Int. J. Control*, 1986, vol. 43, no. 3, pp. 741-766.

[109] K. Glover, "Multiplicative approximation of linear multivariable systems with L^∞ error bound," *American Control Conf., Seattle, USA*, 1986, pp. 1705-1709.

[110] K. Glover and E. A. Jonckheere, "A comparison of two Hankel-norm methods for approximating spectra," *Modeling, Identification and Robust Control, C.I. Byrnes and A. Lindquist, eds., Elsevier Science Publishers / North-Holland, Amsterdam*, 1986, pp. 297-306.

[111] K. Glover and J. C. Doyle, "State-space formulae for all stabilizing controllers that satisfy an H_∞ norm bound and relations to risk sensitivity," *Syst. & Contr. Lett.*, 1988, vol. 11, pp. 167-172.

[112] K. Glover and D. McFarlane, "Robust Stabilization of normalized coprime factor plant description with H_∞-bounded uncertainty," *IEEE Trans. Aut. Control*, 1989, vol. 34, no. 8, pp. 821-830.

[113] P. J. Goddard and K. Glover, "Controller Reduction: Weights for Stability and Performance Preservation," *Proc. 32nd Conf. Decision Control, San Antonio, USA*, 1993, pp. 2903-2908.

[114] P. J. Goddard and K. Glover, "Performance-Preserving Frequency Weighted Controller Approximation: A Coprime Factorization Approach," *Proc. 33rd Conf. Decision Control, Dec. 14-16, Lake Buena Vista, USA*, 1994, pp. 2720-2725.

[115] C.-H. Goh, G. Grubel, and R. A. Hyde, "Robust Stabilization of a Large Flexible Structure via Normalized Coprime Factor Plant Description," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992, no. WA-10.

[116] G. H. Golub and C. Van Loan, "Matrix Computations," *John Hopkins University Press*, 1989.

[117] M. Gopal and S. I. Mehta, "On the selection of eigenvalues to be retained in the reduced order model," *IEEE Trans. Aut. Control*, 1982, vol. 27, no. 3, pp. 688-690.

[118] S. W. Greeley, D. C. Hyland, and S. Richter, "Reduced-order compensation: LQG reduction versus optimal projection using a homotopic continuation method," *Proc. 26th Conf. Decision Control, Los Angeles, USA*, 1987, pp. 742-747.

[119] M. Green, "The approximation of power spectra by phase matching," *Proc. 25th IEEE Conf. Decision Control, Athens, Greece*, 1986, pp. 1085-1090.

[120] M. Green, "Balanced Stochastic Realizations," *J. Lin. Algebra Appl.*, 1988, vol. 98, pp. 211-247.

[121] M. Green, "A relative error bound for balanced stochastic truncation," *IEEE Trans. Aut. Control*, 1988, vol. 33, no. 10, pp. 961-965.

[122] M. Green and B. D. O. Anderson, "Model reduction by phase matching," *Math. Control Signals Systems*, 1989, vol. 2, pp. 221-263.

[123] M. Green and B. D. O. Anderson, "Generalized balanced stochastic truncation," *Proc. 29th Conf. Decision Control, Honolulu, Hawaii*, 1990, pp. 476-481.

[124] G. Gu, "Model reduction with Relative/Multiplicative Error Bounds and Relations to LQG Controller Reduction," *Proc. 33rd Conf. Decision Control, Dec. 14-16, Lake Buena Vista, USA*, 1994, p. 3193.

[125] Y. Halevi, "Frequency weighted model reduction via optimal projection," *Proc. 29th Conf. Decision Control, Honolulu, Hawaii*, 1990, pp. 2906-2911.

[126] Y. Halevi, "Discrete-time frequency weighted model order reduction," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 1972-1975.

[127] G. D. Halikias, Limebeer, D. J. N. , and K. Glover, "A state-space algorithm for the super-optimal Hankel-norm approximation algorithm," *SIAM J. Control and Optimization*, 1993, vol. 31, no. 4, pp. 960-982.

[128] P. Halmos, "Finite Dimensional Vector Spaces," *Van Nostrand, New York*, 1958.

[129] S. J. Hammarling, "Numerical solution of the stable, non-negative definite Lyapunov equation," *IMA J. Numer. Anal.*, 1982, pp. 303-323.

[130] B. Hanzon, "The area enclosed by the oriented Nyquist diagram and the Hilbert-Schmidt-Hankel norm of a linear system," *IEEE Trans. Aut. Control*, 1992, vol. 37, no. 6, pp. 835-839.

[131] B. Hanzon and J. M. Maciejowski, "Constructive Algebra Methods for the L_2 Problem for Stable Linear Systems," *Proc. 10th IFAC Symp. on System Identification*, 1994, vol. 1, pp. 191-196.

[132] Y.-P. Harn and R. L. Kosut, "Optimal Low-order Controller Design via 'LQG-like' Parameterization," *Automatica*, 1993, vol. 29, no. 6, pp. 1377-1394.

[133] P. Harshavardhana, E. A. Jonckheere, and L. M. Silverman, "Stochastic Balancing and Approximation - Stability and Minimality," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 8, pp. 744-746.

[134] P. Harshavardhana and E. A. Jonckheere, "Spectral factor reduction by phase matching – the continuous time SISO case," *Int. J. Control*, 1985, vol. 42.

[135] G. Hirzinger and G. Kreisselmeier, "On Optimal Approximation of High-Order Linear Systems by Low-Order Models," *Int. J. Control*, 1975, vol. 22, no. 3, pp. 339-408.

[136] D. Y. Hu and L. Reichel, "Krylov-Subspace Methods for the Sylvester Equation," *Lin. Alg. Appl.*, 1992, vol. 172, pp. 283.

[137] Y.-S. Hung, K. Glover, "Optimal Hankel-norm approximation of stable systems with first-order stable weighting functions," *Syst. & Contr. Lett.*, 1986, vol. 7, pp. 165-172.

[138] M. F. Hutton and B. Friedland, "Routh Approximations for Reducing Order of Linear Time-Invariant Systems," *IEEE Trans. Aut. Control*, 1975, vol. 20, no. 3, pp. 329-337.

[139] D. C. Hyland, "The optimal projection approach to fixed-order compensation: Numerical methods and illustrative result," *AIAA 21st Aerospace Sciences Meeting, Reno, NV*, 1983, no. 83-0303, pp. 1-10.

[140] D. C. Hyland, "Comparison of various controller-reduction methods: Suboptimal versus optimal projection," *Proc. AIAA Dynamics Specialists Conf., Palm Springs, CA*, 1984, pp. 381-389.

[141] D. C. Hyland and Bernstein D. S., "The optimal projection equations for fixed-order dynamic compensation," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 11, pp. 1034-1037.

[142] D. C. Hyland and D. S. Bernstein, "The Optimal Projection Equations for Model Reduction and the Relationships Among the Methods of Wilson, Skelton, and Moore," *IEEE Trans. Aut. Control*, 1985, vol. 30, no. 12, pp. 1201-1211.

[143] D. C. Hyland and S. Richter, "On direct versus indirect methods for reduced-order controller design," *IEEE Trans. Aut. Control*, 1990, vol. 35, no. 3, pp. 377-379.

[144] H. Inaba and N. Okutsu, "Reduced Order Stabilizing Controllers via Normalized Coprime Factorization," *MTNS '91, Kobe, Japan*, 1991, pp. 205-206.

[145] U. Jaakso and U. Nurges, "Comments on "Approximate linear realization of given dimension via Ho's algorithm"," *IEEE Trans. Autom. Control*, 1975, vol. 20, pp. 302-303.

[146] E. A. Jonckheere and L. M. Silverman, "A new set of invariants for linear systems - application to approximation," *MTNS 1981, Santa Monica, CA*, 1981, vol. 4, pp. 129-133.

[147] E. A. Jonckheere and L. M. Silverman, "A new set of invariants for linear systems - application to reduced order compensator design," *IEEE Trans. Aut. Control*, 1983, vol. 28, pp. 953-964.

[148] E. A. Jonckheere, "Principal component analysis of flexible systems – Open-loop case," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 12, pp. 1095-1097.

[149] E. A. Jonckheere and J. W. Helton, "Power spectrum reduction by optimal Hankel norm approximation of the phase of the outer spectral factor," *IEEE Trans. Aut. Control*, 1985, vol. 30, pp. 1192-1200.

[150] E. A. Jonckheere and R. Li, "L_∞ error bound for Phase Matching Approximation - The One-Step-At-A-Time Hankel Norm Model Reduction," *Int. J. Control*, 1987, vol. 46, no. 4, pp. 1343-1354.

[151] P. T. Kabamba, "Balanced Gains and Their Significance for L_2 Model Reduction," *IEEE Trans. Aut. Control*, 1985, vol. 30, no. 7, pp. 690-693.

[152] P. T. Kabamba, "Balanced forms: canonicity and parameterization," *IEEE Trans. Aut. Control*, 1985, vol. 30, no. 11, pp. 1106-1109.

[153] T. Kailath, "Linear Systems," *Prentice-Hall*, 1980.

[154] D. Kavranoglu and M. Bettayeb, "Characterization of the solution to the optimal H_∞ model reduction problem," *Syst. Control Lett.*, 1993, vol. 20, pp. 99-107.

[155] S. W. Kim, B. D. O. Anderson and A. G. Madievski, "Error Bound for Transfer Function Order Reduction Using Frequency Weighted Balanced Truncation," *Systems & Control Letters*, 1995, vol. 24, pp. 183-192.

[156] H. Kimura, "Pole Assignment by Gain Output Feedback," *IEEE Trans. Aut. Control*, 1975, pp. 509-516.

[157] H. Kimura and Y. Honoki, "Balanced approximation of digital FIR filter with linear phase characteristics," *Proc. Int. Symp. Circuits Syst., Kyoto, Japan*, 1985, vol. 1, pp. 283-286.

[158] K. Knopp, "Infinite sequences and series," *Dover Publications*, 1956.

[159] P. V. Kokotovich, R. E. O'Malley, and P. Sannuti, "Singular perturbations and order reduction in control theory - an overview," *Automatica*, 1976, vol. 12, pp. 123-132.

[160] S. Y. Kung, "A new identification and model reduction algorithm via singular value decomposition," *Proc. 12th Asilomar Conf. Circ. Sys. Comp., Pacific Grove, CA*, 1978.

[161] S. Y. Kung and D. W. Lin, "Optimal Hankel-norm model reductions: Multivariable systems," *IEEE Trans. Aut. Control*, 1981, vol. 26, no. 4, pp. 832-852.

[162] S.-Y. Kung and D. W. Lin, "A state-space formulation for optimal Hankel-norm approximations," *IEEE Trans. Aut. Control*, 1981, vol. 26, no. 4, pp. 942-946.

[163] H. Kwakernaak and R. Sivan, "Linear optimal control systems," *John Wiley & Sons*, 1972.

[164] H. Kwakernaak, "Robust Control and H_∞ Optimization," *Automatica*, 1993, vol. 29, no. 2, pp. 255-273.

[165] J. Lam and B. D. O. Anderson, "L_1 impulse response error bound for balanced truncation," *Syst. & Contr. Lett.*, 1992, vol. 18, pp. 129-137.

[166] G. J. Lastman and N. K. Sinha, "A comparison of the balanced matrix method and the aggregation method of model reduction," *IEEE Trans. Aut. Control*, 1985, vol. 30, no. 3, pp. 301-304.

[167] G. A. Latham and B. D. O. Anderson, "Frequency-weighted optimal Hankel-norm approximation of stable transfer functions," *Syst. & Contr. Lett.*, 1985, vol. 5, pp. 229-236.

[168] A. J. Laub, "A Schur Method for Solving Algebraic Riccati Equations," *IEEE Trans. Aut. Control*, 1979, vol. 24, pp. 913-914.

[169] A. J. Laub, M. T. Heath, C. C. Paige, and R. C. Ward, "Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms," *IEEE Trans. Aut. Control*, 1987, vol. 32, no. 2, pp. 115-121.

[170] A. Lepschy, S. Loro, G. A. Mian, and U. Viaro, "A numerical procedure for rational L_2 approximation," *MTNS 91, Kobe, Japan*, 1991, pp. 271-272.

[171] A. Lepschy, G. A. Mian, G. Pinato, and U. Viaro, "Rational L_2 approximation: A non-gradient algorithm," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 2321-2323.

[172] W. S. Levine and M. Athans, "On the Determination of the Optimal Constant Output-Feedback Gains for Linear Multivariable Systems," *IEEE Trans. Aut. Control*, 1970, vol. 15, no. 1, pp. 44-48.

[173] N. Levinson, "The Wiener RMS Error Criterion in Filter Design and Prediction," *J. Math. Phys.*, 1947, vol. 25, pp. 261-278.

[174] K. Liu and R. E. Skelton, "Q-Markov Covariance Equivalent Realization and its Application to Flexible Structure Identification," *J. Guid. Control Dyn.*, 1993, vol. 16, no. 2, pp. 308-319.

[175] Y. Liu and B. D. O. Anderson, "Controller Reduction via Stable Factorization and Balancing," *Int. J. Control*, 1986, vol. 46, no. 2, pp. 507-531.

[176] Y. Liu and B. D. O. Anderson, "Singular perturbation approximation of balanced systems," *Int. J. Control*, 1989, vol. 50, no. 4, pp. 1379-1405.

[177] Y. Liu, B. D. O. Anderson, and U.-L. Ly, "Coprime Factorization Controller Reduction with Bezout Identity Induced Frequency Weighting," *Automatica*, 1990, vol. 26, no. 2, pp. 233-249.

[178] Y. Liu and B. D. O. Anderson, "Frequency weighted controller reduction methods and loop transfer recovery," *Automatica*, 1990, vol. 26, no. 3, pp. 487-497.

[179] R. Luus, "Optimization in Model Reduction," *Int. J. Control*, 1980, vol. 32, pp. 741-747.

[180] U.-L. Ly, Gangsaas, D., "Application of Modified Linear Quadratic Gaussian Design to Active Control of a Transport Airplane," *AIAA, Guidance and Control Conference, Boulder, Colorado*, 1979.

[181] U.-L. Ly, Gangsaas, D. , Norman, D. C., "Practical gust load alleviation and flutter suppression control laws based on an LQG methodology," *AIAA, Aerospace Sciences Meeting*, 1981.

[182] U.-L. Ly, A. E. Bryson, and R. H. Cannon, "Design of Low-Order Compensators Using Parameter Optimization," *Automatica*, 1985, vol. 21, no. 3, pp. 315-318.

[183] J.-F. Magni and C. Champetier, "On eigenstructure assignment by gain output feedback," *SIAM J. Control and Optimization*, 1991, vol. 29, no. 4, pp. 848-865.

[184] P. M. Mäkilä, "On the Anderson-Moore method for solving the optimal output feedback problem," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 9, pp. 834-836.

[185] P. M. Mäkilä, "Parametric LQ control," *Int. J. Control*, 1985, vol. 41, no. 6, pp. 1413-1428.

[186] P. M. Mäkilä and H. T. Toivonen, "Computational methods for parametric LQ problems - a survey," *IEEE Trans. Aut. Control*, 1987, vol. 32, no. 8, pp. 658-671.

[187] P. M. Mäkilä, "On Identification of Stable Systems and Optimal Approximation," *Automatica*, 1991, vol. 27, no. 4, pp. 663-676.

[188] D. McFarlane and K. Glover, "Robust controller design using normalized coprime factor plant descriptions," *Lecture Notes in Control and Information Sciences, Springer Verlag*, 1989, vol. 138.

[189] D. McFarlane, K. Glover, and M. Vidyasagar, "Reduced-order controller design using coprime factor model reduction," *IEEE Trans. Aut. Control*, 1990, vol. 35, no. 3, pp. 369-373.

[190] L. Meier and D. G. Luenberger, "Approximation of linear constant systems," *IEEE Trans. Aut. Control*, 1967, vol. 12, pp. 585-587.

[191] M. Mercadal, "Study of the constrained LQG problem using homotopy," *Proc. 29th Conf. Decision Control, Honolulu, Hawaii*, 1990, pp. 966-970.

[192] D. G. Meyer, "A fractional approach to model reduction," *Proc. Amer. Control Conf.*, 1988, pp. 1041-1047.

[193] R. N. Mishra and D. A. Wilson, "A new algorithm for optimal reduction of multivariable systems," *Int. J. Control*, 1980, vol. 31, no. 3, pp. 443-466.

[194] D. D. Moerder and A. J. Calise, "Convergence of a numerical algorithm for calculating optimal output feedback gains," *IEEE Trans. Aut. Control*, 1985, vol. 30, no. 9, pp. 900-903.

[195] B. C. Moore, "Principal component analysis in linear systems: controllability, observability, and model reduction," *IEEE Trans. Aut. Control*, 1981, vol. 26, no. 1, pp. 17-31.

[196] A. P. Morgan, A. J. Sommese, and L. T. Watson, "Finding all isolated solutions to polynomial systems using HOMPACK," *ACM Trans. Math. Software*, 1989, vol. 15, no. 2, pp. 93-122.

[197] C. T. Mullis and R. A. Roberts, "Roundoff noise in digital filters: Frequency transformations and invariants," *IEEE Trans. Acoustics Speech & Sign. Proc.*, 1976, vol. 24, no. 6, pp. 538-550.

[198] D. Mustafa, "Reduced-Order Robust Controllers: H_∞-Balanced Truncation and Optimal Projection," *Proc. 29th Conf. Decision Control, Honolulu, Hawaii*, 1990, pp. 488-493.

[199] D. Mustafa and K. Glover, "Controller Reduction by H_∞-Balanced Truncation," *IEEE Trans. Aut. Control*, 1991, vol. 36, no. 6, pp. 668-682.

[200] D. Mustafa, "A class of systems for which balanced truncation is Hankel-norm optimal," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 1943-1948.

[201] D. Mustafa, "Balanced Truncation is Hankel-Norm Optimal when used to Approximate a Finite-Difference Model of a Parabolic PDE," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992, no. FA-9.

[202] D. Mustafa and T. N. Davidson, "Balanced Truncation Can Be Optimal," *Syst. & Contr. Lett.*, 1993, vol. 20, no. 6, pp. 397-400.

[203] K. M. Nagpal and K. Zhou, "Weighted $\mathcal{L}_\infty$ norm approximation.," *Amer. Cont. Conf., June, Seattle, USA*, 1995, pp. 3994.

[204] C. N. Nett, C. A. Jacobson, and M. J. Balas, "A connection between state-space and doubly coprime fractional representations," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 9, pp. 831-832.

[205] C. N. Nett, D. S. Bernstein, W. M. Haddad, "Minimal complexity control law synthesis, part 1: Problem formulation and reduction to optimal static output feedback," *Amer. Cont. Conf., Pittsburgh, USA*, 1989, pp. 2056-2065.

[206] C. N. Nett, D. S. Bernstein, W. M. Haddad, "Minimal complexity control law synthesis, part 2: Problem solution via H_2/H_∞ optimal static output feedback," *Amer. Cont. Conf., Pittsburgh, USA*, 1989, pp. 2506-2511.

[207] E. Noldus and M. Decoster, "Reduction of linear systems by weighted frequency response matching," *Regelungstechnik*, 1976, no. 4, pp. 122-128.

[208] R. Ober, "Balanced parameterization of classes of linear systems," *SIAM J. Control and Optimization*, 1991, vol. 29, no. 6, pp. 1251-1287.

[209] Ph. C. Opdenacker and E. A. Jonckheere, "A state space approach to approximation by phase matching," *Modeling, Identification and Robust Control, C.I. Byrnes and A. Lindquist, eds., Elsevier Science Publishers / North-Holland, Amsterdam*, 1986, pp. 349-362.

[210] A. Packard, M. K. H. Fan, and J. C. Doyle, "A power method for the structured singular value," *Proc. 27th Conf. Decision Control, Austin, Texas*, 1988, vol. 3, pp. 2132-2137.

[211] A. Packard and J. C. Doyle, "The complex structured singular value," *Automatica*, 1993, vol. 29, no. 1, pp. 71-110.

[212] H. Padé, "Sur la réprésentation approchée d'une fonction par des fractions rationelles," *Annales Scientifiques de l'Ecole Normale Supérieure, ser. 3*, 1892, vol. 9, pp. 1-93.

[213] P. J. Parker and B. D. O. Anderson, "Unstable rational function approximation," *Int. J. Control*, pp. 1783-1801.

[214] J. R. Partington, "Approximation of unstable infinite-dimensional systems using coprime factors," *Syst. & Contr. Lett.*, 1991, vol. 16, pp. 89-96.

[215] J. R. Partington, "An Introduction to Hankel Operators," *Cambridge Univ. Press, London Math. Soc. Student Texts*, 1988, vol. 13.

[216] L. Pernebo and L. M. Silverman, "Model reduction via balanced state space realization," *IEEE Trans. Aut. Control*, 1982, vol. 27, no. 2, pp. 382-287.

[217] M. J. D. Powell, "Approximation theory and methods," *Cambridge University Press*, 1981.

[218] R. Prakash, "Properties of a Low-Frequency Approximation Balancing Method of Model Reduction," *IEEE Trans. Aut. Control*, 1994, vol. 39, no. 5, pp. 1135-1141.

[219] Ph. A. Regalia, "A revised look at the stationary points of some adaptive IIR filtering algorithms," *'Challenges of a generalized system theory', eds. Dewilde, Kaashoek and Verhaeghe, Essay of the Dutch Academy of Sciences, Sect. Physics, 1st series, Amsterdam, The Netherlands*, 1993, vol. 40, pp. 9-13.

[220] P. A. Regalia and M. Mboup, "Duality Aspects Between Rational H_2 and Hankel Norm Approximations," *Proc. Intl. Symp. MTNS, Regensburg, Germany*, 1993, pp. 839-842.

[221] P. A. Regalia, "An Unbiased Equation Error Identifier and Reduced-Order Approximations," *IEEE Trans. Sign. Proc.*, 1994, vol. 42, no. 6, pp. 1397-1412.

[222] Ph. A. Regalia, "On the approximation character of some recursive identification schemes," *Proc. of Mathematics in Signal Processing, Warwick, December 1992, J. McWhirter, ed., Clarendon Press, Oxford*, 1994.

[223] Ph. A. Regalia, "Adaptive IIR filtering in signal processing and control," *Marcel Dekker, New York*, 1995.

[224] S. Richter and E. G. Collins, "A homotopy algorithm for reduced order control design using the optimal projection equations," *Proc. 28th Conf. Decision Control, Tampa, FL, USA*, 1989, pp. 506-511.

[225] J. B. Riggs and T. F. Edgar, "Least Squares Reduction of Linear Systems Using Impulse Response," *Int. J. Control*, 1974, vol. 20, no. 2, pp. 213-223.

[226] D. E. Rivera and M. Morari, "Control-relevant model reduction problems for SISO H_2, H_∞, and μ-controller synthesis," *Int. J. Control*, 1987, vol. 46, pp. 505-527.

[227] D. E. Rivera and M. Morari, "Low-order SISO controller tuning methods for the H_2, H_∞ and μ objective functions," *Automatica*, 1990, vol. 26, no. 2, pp. 361-369.

[228] T. J. Rivlin, "An Introduction to the Approximation of Functions," *Blaisdell Publ. Co., Waltham, Mass.*, 1969, also *Dover Publ.*, 1981.

[229] J. D. Roberts, "Linear model reduction and the solution of the algebraic Riccati equation by use of the sign function," *Int. J. Control*, 1980, vol. 32, no. 4, pp. 677-687.

[230] W. Rudin, "Real and Complex Analysis," *McGraw-Hill, New York*, 1966.

[231] Y. Saad, "Numerical solution of large Lyapunov equations," *MTNS 89, Amsterdam, The Netherlands*, 1989, vol. 3, pp. 503-511.

[232] Y. Saad, "Overview of Krylov subspace methods with applications to control problems," *MTNS 89, Amsterdam, The Netherlands*, 1989, vol. 3, pp. 401-410.

[233] M. G. Safonov and R. Y. Chiang, "Model reduction for robust control: A Schur relative-error method," *Int. J. Adapt. Contr. & Signal Proc.*, 1988, vol. 2, pp. 259-272.

[234] M. G. Safonov and R. Y. Chiang, "A Schur method for balanced model reduction," *Amer. Cont. Conf.*, 1988, pp. 1036-1040.

[235] M. G. Safonov, R. Y. Chiang, and D. J. N. Limebeer, "Optimal Hankel model reduction for non-minimal systems," *IEEE Trans. Aut. Control*, 1990, vol. 35, no. 4, pp. 496-502.

[236] V. R. Saksena, J. O'Reilly, and P. V. Kokotovich, "Singular perturbations and time-scale methods in control theory: survey 1976-1983," *Automatica*, 1984, vol. 20, no. 3, pp. 273-293.

[237] G. Schelfhout, P. Van Overschee, and B. De Moor, "Frequency weighted H_2 model reduction," *Nonlinear Numerical Methods and Rational Approximation II, A. Cuyt, ed., Kluwer Academic Publishers*, 1994, pp. 187-199.

[238] G. Schelfhout and B. De Moor, "A relative error model reduction method using balancing," *Proc. 33rd Conf. Decision Control, Dec. 14-16, Lake Buena Vista, USA*, 1994, pp. 2734-2735.

[239] G. Schelfhout and B. De Moor, "Frequency weighted H_2 and Hilbert-Schmidt-Hankel norm model reduction," *Proc. 33rd Conf. Decision Control, Dec. 14-16, Lake Buena Vista, USA*, 1994, pp. 3215-3216.

[240] G. Schelfhout and B. De Moor, "Inverse Outer Factor Weighted Balanced Truncation," *ESAT-SISTA Internal Report No. 94-43.*

[241] G. Schelfhout and B. De Moor, "Optimal Model Reduction in the Frequency Weighted H_2 and Hilbert-Schmidt-Hankel Norms," *ESAT-SISTA Internal Report No. 94-44.*

[242] G. Schelfhout and B. De Moor, "Time-Domain Weighted Balanced Truncation," *Proc. 3rd European Control Conference, TM-3, September 1995, Rome, Italy*, pp. 2187-2190.

[243] G. Schelfhout and B. De Moor, "Closed-Loop Balanced Truncation for Observer-Based Controllers," *Accepted for publication in IEEE Trans. on Autom. Control*.

[244] G. Schelfhout and B. De Moor, "A Note on Closed-Loop Balanced Truncation," *Accepted for 34th IEEE Conf. on Decision and Control, New Orleans, USA*, 1995.

[245] E. Schömig and U.-L. Ly, "Mixed H_2/H_∞ Control with an Output Feedback Compensator Using Parameter Optimization," *Amer. Cont. Conf.*, 1992, pp. 733-737.

[246] J. A. Sefton and R. Ober, "Hankel Norm Approximation and Control Systems," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992, no. WP-6.

[247] Y. Shamash, "Stable reduced-order models using Padé-type approximations," *IEEE Trans. Autom. Control*, 1974, vol. 19, pp. 615-616.

[248] S. Shokoohi, L. M. Silverman, and P. M. Van Dooren, "Linear time-variable systems: Balancing and model reduction," *IEEE Trans. Aut. Control*, 1983, vol. 28, no. 8, pp. 810-822.

[249] L. M. Silverman, M. Bettayeb, "Optimal approximation of linear systems," *Proc. Joint Autom. Control Conf.*, 1980, no. FA8-A.

[250] R. E. Skelton and E. G. Collins, "Set of q-Markov covariance equivalent models of discrete systems," *Int. J. Control*, 1987, vol. 46, pp. 1-12.

[251] M. C. Smith, "Stabilization by reduced-order controllers," *IEEE Trans. Aut. Control*, 1991, vol. 36, no. 1, pp. 120.

[252] J. T. Spanos, M. H. Milman, and D. L. Mingori, "A New Algorithm for L_2 Optimal Model Reduction," *Automatica*, 1992, vol. 28, no. 5, pp. 897-909.

[253] V. Sreeram and P. Agathoklis, "Model reduction using balanced realizations with improved low frequency behavior," *Syst. & Contr. Lett.*, 1989, vol. 12, pp. 33-38.

[254] V. Sreeram and P. Agathoklis, "Model reduction of linear discrete systems via weighted impulse response gramians," *Int. J. Control*, 1991, vol. 53, no. 1, pp. 129-144.

[255] V. Sreeram and P. Agathoklis, "Discrete-system reduction via impulse-response Gramians and its relation to q-Markov covers," *IEEE Trans. Aut. Control*, 1992, vol. 37, pp. 653-658.

[256] V. Sreeram and P. Goddard, "Model reduction using impulse-response gramians: a frequency domain approach," *Int. J. Systems Science*, 1992, vol. 23, no. 10, pp. 1745-1760.

[257] K. Steiglitz and L. E. McBride, "A technique for identification of linear systems," *IEEE Trans. Aut. Control*, 1965, vol. 10, pp. 461-464.

[258] M. Steinbuch and O. Bosgra, "Necessary conditions for static and fixed order dynamic mixed H_2/H_∞ optimal control," *Selected Topics in Identification, Modeling and Control, Delft University Press*, 1990, vol. 2, pp. 17-23.

[259] G. W. Stewart, "Algorithm 406: HQR3 and EXCHNG: Fortran Subroutines for Calculating and Ordering the Eigenvalues of a Real Upper Hessenberg Matrix," *ACM Trans. Math. Soft.*, 1976, vol. 2, pp. 275-280.

[260] P. Stoica and T. Söderström, "The Steiglitz-McBride algorithm revisited - Convergence analysis and accuracy aspects," *IEEE Trans. Aut. Control*, 1981, vol. 26, pp. 712-717.

[261] L. Thiele, "Balanced model reduction in time- and frequency domain," *Int. Symposium on Circuits and Systems*, 1985, pp. 345-348.

[262] L. Thiele, "On the sensitivity of linear state-space systems," *IEEE Trans. Circuits and Systems*, 1986, vol. 33, pp. 502-510.

[263] H. T. Toivonen, "A globally convergent algorithm for the optimal constant output feedback problem," *Int. J. Control*, 1985, vol. 41, no. 6, pp. 1589-1599.

[264] H. T. Toivonen and P. M. Mäkilä, "Newton's method for solving parametric linear quadratic control problems," *Int. J. Control*, 1987, vol. 46, no. 3, pp. 897-911.

[265] H. T. Toivonen and P. M. Mäkilä, "A descent Anderson-Moore algorithm for optimal decentralized control," *Automatica*, 1988, vol. 21, no. 6, pp. 743-744.

[266] W. F. Trench, "An Algorithm for the Inversion of Finite Toeplitz Matrices," *J. SIAM*, 1964, vol. 12, pp. 515-522.

[267] W. F. Trench, "Inversion of Toeplitz band matrices," *Math. Comp.*, 1974, vol. 28, pp. 1089-1095.

[268] A. Trofino and V. Kucera, "Stabilization via static output feedback," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 910-913.

[269] G. Troullinos and J. F. Dorsey, "Application of Balanced Realizations to Power System Equivalents," *IEEE Trans. Aut. Control*, 1985, vol. 30, no. 4, pp. 414-416.

[270] M. Van Barel and A. Bultheel, "A parallel algorithm for discrete least squares rational approximation," *Numer. Math.*, 1992, vol. 63, pp. 99-121.

[271] M. Van Barel and A. Bultheel, "Discrete linearized least squares rational approximation on the unit circle," *J. Comput. Appl. Math.*, 1994, vol. 50, pp. 545-563.

[272] M. Van Barel and A. Bultheel, "Orthogonal polynomial vectors and least squares approximation for a discrete inner product," *Electr. Trans. Numer. Anal.*, 1995, vol. 3, pp. 1-23.

[273] A.-J. Van der Veen and P. M. Dewilde, "AAK model reduction for time-varying systems," , pp. 125-130.

[274] P. Van Dooren, "Numerical Linear Algebra Techniques for Large Scale Matrix Problems in Systems and Control," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992, no. TM-6.

[275] S. Van Huffel, B. De Moor, and H. Chen, "Relationships between Structured and Constrained TLS, with Applications to Signal Enhancement," *Proc. of the Int. Symposium on the Mathematical Theory of Networks and Systems (MTNS), Regensburg, Germany, August 2-6*, 1993.

[276] P. T. L. M. Van Woerkom, "Mathematical models of flexible aircraft dynamics: a survey of order approaches," *IFAC Automatic Control in Space, Toulouse, France*, 1985, pp. 127-135.

[277] A. Varga, "A note on Hammarling's algorithm for the discrete Lyapunov equation," *Syst. & Contr. Lett.*, 1990, vol. 15, pp. 273-275.

[278] A. Varga, "Balancing free square-root algorithm for computing singular perturbation approximations," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 1062-1065.

[279] E. I. Verriest, "Low Sensitivity Design and Optimal Order Reduction for the LQG Problem," *Proc. 24th Midwest Symp. Circ. Syst.*, Albuquerque, NM, USA, 1981, pp. 365-369.

[280] E. I. Verriest, "Suboptimal LQG-Design via Balanced Realizations," *Proc. 20th Conf. Decision Control, San Diego, USA*, 1981, pp. 686-687.

[281] E. I. Verriest and T. Kailath, "On generalized balanced realizations," *IEEE Trans. Aut. Control*, 1983, vol. 28, no. 8, pp. 504-505.

[282] E. I. Verriest, "Reduced Order LQG Design: Conditions for Feasibility," *Proc. 25th Conf. Decision Control, Athens, Greece*, 1986, pp. 1765-1769.

[283] E. I. Verriest and G. Pajunen, "Robust Control of Systems with Input and State Constraints via LQG-Balancing," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992, no. FA-4.

[284] M. Vidyasagar, H. Schneider, and B. A. Francis, "Algebraic and topological aspects of feedback stabilization," *IEEE Trans. Aut. Control*, 1982, vol. 27, no. 4, pp. 880-894.

[285] M. Vidyasagar, "The graph metric for unstable plants and robustness estimates for feedback stability," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 5, pp. 403-417.

[286] M. Vidyasagar, "Control system synthesis, a factorization approach," *MIT press*, 1985.

[287] M. Vidyasagar and H. Kimura, "Robust Controllers for Uncertain Linear Multivariable Systems," *Automatica*, 1986, vol. 22, no. 1, pp. 85-94.

[288] M. Vidyasagar, "Normalized Coprime Factorizations for Non-strictly Proper Systems," *IEEE Trans. Aut. Control*, 1988, vol. 33, pp. 300-301.

[289] D. A. Wagie and R. E. Skelton, "Model Reduction and Controller Synthesis in the Presence of Parameter Uncertainty," *Automatica*, 1986, vol. 22, no. 3, pp. 295-308.

[290] W. Wang and M. Safonov, "A tighter relative-error bound for balanced stochastic truncation," *Syst. Contr. Lett.*, 1990, vol. 14, pp. 307-317.

[291] W. Wang, "Relative-error model reduction, identification and control.," *Ph.D. thesis, Dept. Electrical Engineering-Systems, Univ. of Southern California*, 1990.

[292] W. Wang, J. C. Doyle, C. Beck, and K. Glover, "Model reduction of LFT systems," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 1233-1238.

[293] J. C. Willems, "Paradigms and puzzles in the theory of dynamical systems," *IEEE Trans. Aut. Control*, 1991, vol. 36, no. 3, pp. 259-294.

[294] D. A. Wilson, "Optimum solution of model-reduction problem," *Proc. IEE-D*, 1970, vol. 117, pp. 1161-1165.

[295] D. A. Wilson, "Model Reduction of Multivariable Systems," *Int. J. Control*, 1974, vol. 20, no. 1, pp. 57-64.

[296] D. A. Wilson and R. N. Mishra, "Optimal Reduction of Multivariable Systems," *Int. J. Control*, 1979, vol. 29, no. 2, pp. 267-278.

[297] D. A. Wilson and R. N. Mishra, "Design of Low-Order Estimation Using Reduced Models," *Int. J. Control*, 1979, vol. 29, no. 3, pp. 447-456.

[298] W. M. Wonham, "Linear multivariable control: A geometric approach," *Springer Verlag, New York*, 1974.

[299] P. M. R. Wortelboer and O. H. Bosgra, "Generalized Frequency Weighted Balanced Reduction," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992, FA-9.

[300] P. M. R. Wortelboer and O. H. Bosgra, "Mixed forms of modal and balanced reduction," *Selected Topics in Identification, Modeling and Control, Delft University Press*, 1993, vol. 6, pp. 143-150.

[301] P. M. R. Wortelboer, "Frequency-weighted balanced reduction in closed-loop systems," *Selected Topics in Identification, Modeling and Control, Delft University Press*, 1993, vol. 6, pp. 133-142.

[302] Wortelboer, P. M. R., "Frequency-weighted Balanced Reduction of Closed-loop Mechanical Servo-systems: Theory and Tools," *Ph.D. dissertation*, Techn. Univ. Delft, 1994.

[303] Wortelboer, P. M. R., "Wor-Toolbox," Matlab Toolbox for Weighted Order Reduction, ftp ftp-mr.wbmt.tudelft.nl, cd /pub/wortelboer

[304] P. M. R. Wortelboer and O. H. Bosgra, "An iterative algorithm for frequency-weighted H_2-norm optimal reduction and centering," *Selected Topics in Identification, Modeling and Control, Delft University Press*, 1994, vol. 7, pp. 23-34.

[305] P. M. R. Wortelboer and O. H. Bosgra, "Frequency Weighted Closed-Loop Order Reduction in the Control Design Configuration," *Proc. 33rd IEEE Conf. Dec. Control, Lake Buena Vista, USA*, 1994, pp. 2714-2719.

[306] J.-S. Yang, "On output frequency-weighted controller reduction," *Proc. 30th Conf. Decision Control, Brighton, UK*, 1991, pp. 245-247.

[307] F.-B. Yeh and L.-F. Wei, "Super-Optimal Hankel-Norm Approximations," *Proc. 31st Conf. Decision Control, Tucson, USA*, 1992, no. FA-9.

[308] H.-H. Yeh, C. S. Hsu and S. S. Banda, "Fixed-order H_∞ Compensator Design," *Int. J. Robust and Nonlinear Control*, 1994, vol. 4, no. 3, pp. 363.

[309] Yi Ch., "Robustness Analysis and Controller Design for Systems with Structured Uncertainties," *Doctoral dissertation, ESAT-SISTA, Kath. Univ. Leuven*, May 1995.

[310] D. C. Youla, J. J. Bongiorno, and H. A. Jabr, "Modern Wiener-Hopf design of optimal controllers – Part I: The single-input single-output case," *IEEE Trans. Aut. Control*, 1976, vol. 21, pp. 3-13.

[311] D. C. Youla, H. A. Jabr, and J. J. Bongiorno, "Modern Wiener-Hopf design of optimal controllers – Part II: The multivariable case," *IEEE Trans. Aut. Control*, 1976, vol. 21, pp. 319-338.

[312] A. Yousuff and R. E. Skelton, "A note on balanced controller reduction," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 3, pp. 254-257.

[313] A. Yousuff and R. E. Skelton, "Controller Reduction by Component Cost Analysis," *IEEE Trans. Aut. Control*, 1984, vol. 29, no. 6, pp. 520-530.

[314] A. Yousuff and R. E. Skelton, "An optimal controller reduction by covariance equivalent realizations," *IEEE Trans. Aut. Control*, 1986, vol. 31, no. 1, pp. 56-60.

[315] A. Yousuff, D. A. Wagie, and R. E. Skelton, "Linear system approximation via covariance equivalent realizations," *J. Math. Anal. Appl.*, 1985, vol. 106, pp. 91-115.

[316] G. Zames, "Feedback and optimal sensitivity: Model reference transformations, multiplicative semi-norms, and approximate inverses," *IEEE Trans. Aut. Control*, 1981, vol. 26(2), no. 2, pp. 301-320.

[317] H. P. Zeiger and A. J. McEwen, "Approximate linear realization of given dimension via Ho's algorithm," *IEEE Trans. Autom. Control*, 1974, vol. 19, pp. 153.

[318] K. Zhou, "Remarks on Stochastic Model Reduction," *IEEE Trans. Aut. Control*, 1987, vol. 32, no. 4, pp. 356-358.

[319] K. Zhou, "Frequency-weighted model reduction with L_∞ error bounds," *Syst. & Contr. Lett.*, 1993, vol. 21, no. 2, pp. 115-126.

[320] K. Zhou, Y. Zheng, and L. Ting, "Stability and error bounds for discrete time frequency weighted balanced truncation.," *Proc. 32nd Conf. Decision & Control, San Antonio, Texas*, 1993, pp. 2927-2928.

[321] K. Zhou, "Error bounds for frequency weighted balanced truncation and relative error model reduction.," *Proc. 32nd Conf. Decision & Control, San Antonio, Texas*, 1993, pp. 3347-3352.

[322] K. Zhou, "Weighted Optimal Hankel Norm Model Reduction," *Proc. 32nd Conf. Decision & Control, San Antonio, Texas*, 1993, pp. 3353-3354.

[323] K. Zhou, "A Comparative Study of H_∞ Controller Reduction Methods," *Amer. Cont. Conf., June, Seattle, USA*, 1995, pp. 4015.

[324] K. Zhou, "Frequency-Weighted $\mathcal{L}_\infty$ Norm and Optimal Hankel Norm Model Reduction," *IEEE Trans. Aut. Control*, 1995, vol. 40, no. 10, pp. 1687-1699.