Lehrstuhl für Regelungstechnik
Fakultät für Maschinenwesen
Technische Universität München

Preserving Stability in Model and Controller Reduction

with application to embedded systems

Amirhossein Yousefi

Vollständiger Abdruck der von der Fakultät für Maschinenwesen
der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr.-Ing. Wolfgang A. Wall
Prüfer der Dissertation:	
	1. Univ.-Prof. Dr.-Ing. habil. Boris Lohmann
	2. Univ.-Prof. Dr. rer. nat. Oliver Junge

Die Dissertation wurde am 19.10.2006 bei der Technischen Universität München eingereicht und durch die Fakultät für Maschinenwesen
am 12.12.2006 angenommen.

A ship in port is safe,
but that is not what ships are for.
Sail out to sea and do new things.
– Admiral Grace Hopper, Computer Pioneer

DEDICATION

To my parents, for making it possible to embark on this journey. To my wife, for making it possible to conclude it.

ACKNOWLEDGMENTS

I would like to express my gratitude to my adviser, Boris Lohmann, for his support, patience, and encouragement throughout my graduate studies. It is not often that one finds an adviser and colleague that always finds the time for listening to the little problems and roadblocks that unavoidably crop up in the course of performing research. His technical and editorial advice was essential to the completion of this dissertation and has taught me innumerable lessons and insights on the workings of academic research in general.

The friendship of Behnam Salimbahrami and Michael Buhl is much appreciated and has led to many interesting and good-spirited discussions relating to this research. I am also grateful to my students Paul De Monter, Jürgen Haag and Alexander Dietrich for helping considerably with realizing the code for embedded applications.

My thanks also go to the members of the institute of automation at the university of Bremen where this work has partly done there and the institute of automatic control at the technical university of Munich for their assistance and the friendly environment they provided.

Last, but not least, I would like to thank my wife Shadi for her understanding and love during the past few years. Her support and encouragement was in the end what made this dissertation possible. My parents, Mehdi and Eftekhar, receive my deepest gratitude and love for their dedication and the many years of support during my undergraduate studies that provided the foundation for this work.

Amirhossein Yousefi — Munich, August 2006

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Part I

Preliminaries

Chapter 1

INTRODUCTION

1.1 Motivating Trends

Since starting the industrial revolution, the leading industry has been requiring more and more precise tools for delivering high quality production while making the production procedure as efficient as possible and the production cost as low as possible. In this context, automatic control has found its place among the leading technologies. In the early days a control engineer because of hardware implementation limits was restricted essentially to use her knowledge of the process and her understanding of the underlying physics to design very simple controllers. As rigorous standards were imposed on the performance specifications, the need of advanced automatic tools from which complex controllers could be designed became a major issue. On the other hand recent advances in high speed processors with the capability of carrying out high numerical computations in real time have let complicated control algorithm to be realized. Therefore quickly, the theorist of feedback control theory oriented their research toward the computation of high performance (*optimal*) controllers. For instance a product of these attempts is Robust H_∞ control design which uses bounds on the modeling error to design controller with guaranteed stability and/or performance.

Regarding the fact that practicing engineers need controllers which are precise and have low complexity, the new developed design tools came short to satisfy the latter requirement. The complexity of new controllers restricted their operation in embedded systems and increased their implementation costs. As the control theorist did not make much effort to reduce the complexity of controllers a huge gap between theory and practice appeared. During the last two decades many efforts have been made to bridge this gap and a new field of research emerged, which has been called *controller reduction*. This orientation lies basically on the ideas from model reduction approaches. But it later became clear that a major issue that makes controller reduction distinct from model reduction is the interplay between controller reduction and control design. The recent developments in this field has made the reduction for control to be seen as a complementary design step.

1.2 State of The Art

The early works on controller reduction only aimed at expanding the ideas from model reduction field, without considering the closed loop stability and performance. Modern methods for model reduction started in the sixties. The key feature of all techniques was to retain the dominant modes of the original system and to discard the nondominant modes. Then in 1981, B. C. Moore, introduced the concept of balanced realization and its application in model reduction [47]. This was a breakthrough in the field of model reduction and provided the balanced truncation method for reducing a stable linear system with associated a priori error bound. Some different aspects of this method are reviewed in Chapter 3. As mentioned the advanced controller design methods tend to supply controller with order comparable to the plant order. The available methods of model reduction did not prove of themselves an effective way for controller reduction, since the do not provide a systematic way for handling the problem of retaining closed loop stability and performance. During the eighties, controller reduction became a field of tremendous research activity. Some notable results were achieved, all trying to capture the plant into the controller reduction problem. Perhaps the first result was published in 1984 by D. F. Enns and his method relied on introducing frequency weighting (as a function of plant) into the balanced truncation procedure [21]. Another direction of development that occurred in the nineties was the application of coprime factorization. This line of research was initiated by the publication of K. Glover and D. Mustafa considering reduction of H_∞ optimal controllers [49]. A summary of major achievements is presented in Chapter

4.

Although different controller order reduction approaches serve different objectives, stability and transfer matrix matching are common main goals in all of them. Normally, stability is an essential property that cannot be ignored and it is a challenge to transfer matrix matching. In other words, the approaches that focus on preserving special characteristics of the original transfer matrix might easily lose the stability of the reduced system. For instance Enns controller reduction method (Balanced truncation with two sided weighting) suffers the lack of stability, although it is successful in transfer matrix matching of the closed loop system in the sense of infinity norm. Similar to model reduction, there has been some efforts to modify controller reduction approaches to preserve the stability of the controller itself. For instance the publication of Lin and Chiu proposed a minor adjustment to the schemes of Enns which guarantees just the stability of the reduced order controller but not the closed loop system [38]. This idea was further developed in [59, 65] and [58]. An open problem which still remains is how we can guarantee the stability of the closed loop system. This dissertation is dedicated to give an answer to this question.

1.3 Thesis Contribution

The contributions of this thesis address both researchers in field of model and controller reduction and they are essentially focused on the problem of instability in both cases.

The first contribution concerns model reduction methods based on similarity transformations. For a given model a general framework is proposed that parameterizes a large set of reduced models that preserve the stability of the original model. As an application of this result, it is shown how different model reduction methods can be modified, if they fail to maintain stability. The basic idea in this part is also exploited as a guideline to deal with the main objective of this dissertation which is preserving stability of the closed loop system in case of controller reduction. This result is detailed in Chapter 5 and has been partially published in

Yousefi, A. and Lohmann, B. "A note on Stability in Model Reduction". *submitted to the International Journal of Systems Science*, 2006.

The second contribution of this thesis is isolating the problem of guaranteeing the internal stability of the reduced closed loop system from the problem of transfer matrix matching. Our treatment of stability follows that of [29, 60], where stability is guaranteed by generalized Gramians. The approach here, utilizes the theoretic framework based on strict Lyapunov inequalities. Then for a given controller a framework is proposed that parameterizes a set of reduced controllers that preserve the stability of the closed loop system. In addition, a sufficient condition for the existence of such a framework is derived. An interesting application of these results is an algorithm to preserve stability in currently available controller reduction approaches that suffer from the lack of stability, by slightly modifying the corresponding similarity transformations. The details of these results are available in Chapter 6 and have been in published in

Yousefi, A. and Lohmann, B. "A Parameterization of Reduced Stable Models and Controllers". *submitted to the International Journal of Control*, 2006.

A special application of this result to frequency weighted controller reduction approach is studied in an example in Chapter 8 and more comprehensively published in

Yousefi, A. and Lohmann, B. "Stabilizing Frequency Weighted Controller Reduction Approach". *submitted to ISA (The Instrumentation, Systems and Automation Society) Transactions*, 2006.

1.4 Synopsis

This thesis is organized as follows.

Chapter 2: Feedback Control Theory. This chapter reviews the trend of feedback control theory in the last decades and it summarizes the achievements on mathematical analysis of stability and robustness. It also includes a description of the closed loop setup that is considered in this thesis, as well as the definitions and tools for analysing closed loop stability and performance. Finally the basic motivations for performing controller reduction are explained.

Chapter 3: Order Reduction. This chapter contains a summary of model reduction based on balancing and its different interpretations, namely from energy, linear operators and optimality point of view.

Chapter 4: Low Order Controller. This chapter explains the concept of stabilization using feedback and the minimal order required for stabilization. Then a summary of well-known controller reduction approaches based on minimizing multiplicative error, relative error, frequency weighted absolute error, coprime factors and H_∞ error are presented and pros and cons of each approach is shortly expressed.

Chapter 5: Preserving Stability in Model Reduction. It sometimes happens that model reduction methods based on similarity transformation result in unstable models although the original model is stable. Given a stable model, based on Lyapunov criterion, this chapter proposes a general framework that parameterizes a large set of stable reduced models that each can be obtained by truncating a realization of the original model. As an application of this result it is shown how Krylov subspace based model reduction methods and frequency weighted reduction methods can be modified, if they fail to maintain stability.

Chapter 6: Preserving Stability in Controller Reduction. This chapter addresses the problem of instability in controller reduction methods based on similarity transformations. For a given controller a framework is proposed that parameterizes a set of reduced controllers that preserve the stability of the closed loop system. Then based on this result a complementary algorithm to obtain a stabilizing stable reduced controller based on similarity transformations is presented. As an application of this result it is shown how the two sided frequency weighted reduction method can be modified to guarantee the closed loop stability. In addition, a sufficient condition for the existence of such a framework is derived.

Chapter 7: Effect of Controller Reduction on Embedded Controllers. Our results in Chapter 6 combined with methods in Chapter 4 give a reduced controller that guarantees the stability and performance of the closed loop system. This chapter aims at giving some insight into the effect of controller reduction on the size and processing speed of embedded controllers.

Chapter 8: Illustrative Examples. The efficiency of the methods presented in Chapter 5 and 6 are illustrated with two examples from literatures. By means these examples a comparison study among different approaches is carried out and the advantages of the new methods are highlighted.

Chapter 9: Conclusions and Discussions. This chapter concludes this thesis and proposes some possible further research topics.

Chapter 2

FEEDBACK CONTROL THEORY

2.1 Introduction

Automatic feedback control systems have been known and used for more than 2000 years. The word *feedback* is a 20_{th} century neologism introduced by radio engineers to describe parasitic, positive feedback of the signal from the output of an amplifier to the input circuit. Afterward it has entered gradually into common usage in the English-speaking world during the latter half of the century. The history of automatic control divides conveniently into four main eras as follows:

- Early Control: to 1900
- The Pre-Classical Period: 1900-1940
- The Classical Period: 1935-1955
- Modern Control: Post-1955

Table 2.1 shows briefly the trend of modern control since 1955 for linear systems, concerning advantages and disadvantages of the provided tools in this period. For an interesting overview of control history see [12, 13]. As it is clear form Table 2.1, the modern control theory can address robust stability and performance in a systematic design procedure, but getting high order controller is unavoidable. One of the successes of modern linear control is introducing the concept of robust stability

Time	method	advantages	disadvantages
30's -50's	Classical Control (PID, Lead-Lag, Root Locus)	graphics-oriented	restricted to SISO systems
50-60's	Linear Quadratic Regulator(LQR)	guaranteed stability margin pure gain controller	full state feedback accurate model many iterations
60-70's	Linear Quadratic Gaussian(LQG)	use available data on noise and disturbance	no stability margin accurate model many iterations
80-90's	Loop Transfer Recovery(LQG/LTR)	guaranteed stability margin systematic design procedure	possible high gain focus on one point
	$\mathcal{H}_2$ Optimal Control $\mathcal{H}_\infty$ Optimal Control μ Synthesis	systematic design address stability and sensitivity transparent performance definition	many iterations high order controller

Table 2.1: A brief history of modern automatic control for linear systems

and its mathematical tools. Sections 2.2 and 2.3 are concerned with this definition and related mathematical preliminaries, which are used as essential tools in proceeding chapters. Another outcome of modern linear control is formulating performance in a transparent manner. Section 2.4 reviews concisely the definition of performance and restrictions on it in presence of constraints.

2.2 *Stability*

In this section we consider the stability of a general closed loop system which is shown schematically in Figure 2.1. Assume that the plant G (which is a model of a real system), the controller K and sensor F (which are the models of a hardware realization of a controller and sensors) in Figure 2.1 are fixed real rational proper transfer matrices[1]. We should always keep in mind that for linearized model the

[1] We will relax this assumption in the next section (robust stability) by taking into account the effect of wide range of uncertainties.

whole argument in this section is valid only in an area around the operating point and we have to bury in mind that it's better (at least as a point of view of stability analysis) to design the plant and implement the controller as linear as possible. We can roughly say "stability in the sense of physics means more robust stability, because the reality is not as ideal as what we do in mathematics".

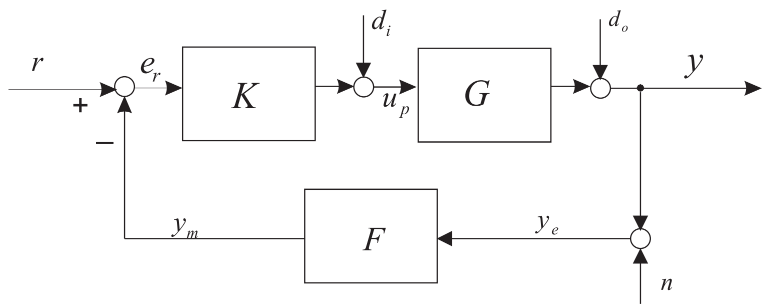

Figure 2.1: One degree of freedom control configuration

The first question one would ask is whether the feedback interconnection makes sense or is physically realizable. Therefore before we step in the definition of stability we define the notation of well-posedness.

Definition 2.1: A feedback system is called well-posed when it is physically realizable [74].

Well-posedness can be checked Mathematically using the following theorems.

Theorem 2.1: A feedback system is well-posed iff $I+L(\infty)$ is invertible, where $L(s) = F(s)G(s)K(s)$ is the loop transfer function.

Theorem 2.2: In terms of state-space realization, well-posedness is equivalent to the invertibility of $\begin{bmatrix} I & 0 & D_f \\ -D_k & I & 0 \\ 0 & -D_g & I \end{bmatrix}$, where D_i matrices denote the feed-through matrices of the part systems, see (2.1).

Fortunately, in most practical cases we have $D_g = 0$ and hence well-posedness for most practical systems is guaranteed. It is also clear that well-posedness is the necessary condition for the stability.

Definition 2.2: A system is (internally) **stable** if all signals, containing all measurable and immeasurable signals in all components including even hidden modes, remain bounded provided that the all injected signals (at any possible location) and initial conditions are bounded.

It should be noted that stability analysis in contrast to its initial definitions (in other references) is independent of the way we have chosen the inputs-outputs pair and is completely an internal property of a system [2]. Internal stability is a basic requirement for a practical feedback system. This is because all interconnected systems maybe unavoidable subject to some nonzero initial conditions and some errors, and it cannot be tolerated in practice that such errors at some location will lead to unbounded signals at some other locations in the closed-loop system.

Another approach to define stability is Lyapunov's criterion which is equivalent to Definition 2.2 (at least for LTI systems). The basic idea of this approach is as follows:

Definition 2.3: If the total energy of a dynamic system is continuously dissipated, then the system is called **stable** and it will finally settle down to an equilibrium point.

In this method by finding an appropriate[3] functional V, which (in a generalized point of view) represents the energy of the whole system, useful stability criteria can be established not only for LTI systems but also for several class of nonlinear systems. Based on similar ideas the other notations like passive systems or dissipative systems are also discussed in other references which have more values in theoretical discussions

[2] In some special cases the internal stability is equivalent to external stability of some pair of inputs and outputs, for instance see Theorem 4.5 in [55].

[3] $V(0) = 0,\ \forall x(\neq 0)\ V(x) > 0,\ \dot{V}(x) < 0$

rather that practical ones. More details on Lyapunov's method can be found in [35] and more information on passive and dissipative systems are available in [67, 68].

2.2.1 Mathematical analysis of stability

State space

The state space approach represents a very clear and usually understandable view of different modes (modal approach) and their relationships. Consider the following state space representation in continuous and discrete form:

$$\mathcal{S} : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} \tag{2.1}$$

$$\mathcal{S}_d : \begin{cases} x(k+1) = A_d x(k) + B_d u(k) \\ y(k) \qquad = C_d x(k) + D_d u(t) \end{cases} \tag{2.2}$$

Analyzing stability in the state space matches completely with the above definition. Because it is done by analyzing the matrix A (or A_d) which reflects comprehensively all internal modes of the system without assuming the interactions of inputs and outputs. Loosely speaking every route that connects any possible pair of input-output will pass through some parts (or all) of these internal modes. Therefore if all the internal modes are stable, the stability of the whole system is guaranteed.

Theorem 2.3: The system represented in (2.1) is (internally) stable iff $Re\{\lambda_i(A)\} < 0$. (Discrete time: (2.2) is stable iff $\|\lambda_i(A_d)\| < 1$.)

Remark 2.1: Assuming A is an $n \times n$ matrix, its eigenvalues ($\lambda_i(A),\ i = 1, \cdots, n$) are the roots of the n'th order characteristic equation $\Phi(\lambda) = det(A - \lambda I)$.

Corollary 2.1: **Lyapunov Criterion**: Matrix A is a stability matrix ($Re\{\lambda_i(A)\} < 0$) iff for any positive definite (symmetric) matrix Q, exists a positive definite (symmetric) matrix P such that $A^T P + PA = -Q$. (Discrete time: iff $P - A^T PA = -Q$)

Corollary 2.2: The system represented in (2.1) is (internally) stable iff the matrix A is a stability matrix.

This result is also consistent with Lyapunov's criterion (Defenition 2.3), noting that the quantity $V(x(t)) = x^T(t)Px(t)$ can be regarded as a generalized energy associated with the realization (2.1). In a stable system the energy should decay with time, consistent with the calculation

$$\frac{d}{dt}V(x(t)) = x^T(t)(PA + A^T P)x(t) = -x^T(t)Qx(t). \tag{2.3}$$

From this we can conclude that $x(t) \to 0$ as $t \to \infty$, provided Q is positive definite. It can also be shown that if A is a stability matrix, then the Lyapunov equation ($A^T P + PA = -Q$) has a *unique* solution for every Q which equals to $P = \int_0^\infty e^{A^T t} Q e^{At} dt$[4].

Frequency domain

In the frequency domain we suppose that all transfer matrices in Figure 2.1 have minimal realizations and none of them contains any unobservable or uncontrollable modes. Mathematically it means that in constructing each transfer matrix has not any zero-pole cancellation occurred. The stability of the close loop system can be analyzed using the following theorems:

Theorem 2.4: The system described by Figure 2.1 is (internally) stable iff the transfer matrix $\begin{bmatrix} I & 0 & F \\ -K & I & 0 \\ 0 & -G & I \end{bmatrix}^{-1}$ belongs to $\mathcal{RH}_\infty$.

With the assumption that all transfer matrices have minimal realizations, the internal stability of the closed loop system is equivalent to external stability of the following

[4]The matrix equation $FX+XG+C = 0$ has a unique solution iff $\lambda_i(F) \neq \lambda_j(-G)$. if $Re(\lambda_i(F)) + Re(\lambda_j(G)) < 0$ this unique solutions is given by $X = \int_0^\infty e^{F\tau} C e^{G\tau} d\tau$.

nine (in fact four distinct) transfer matrices [5]

$$
\begin{aligned}
\begin{pmatrix} r \\ d_i \\ n \end{pmatrix} &= \begin{bmatrix} I & 0 & F \\ -K & I & 0 \\ 0 & -G & I \end{bmatrix} \begin{pmatrix} e_r \\ u_p \\ y_e \end{pmatrix} \Leftrightarrow \\
\begin{pmatrix} e_r \\ u_p \\ y_e \end{pmatrix} &= \begin{bmatrix} I & 0 & F \\ -K & I & 0 \\ 0 & -G & I \end{bmatrix}^{-1} \begin{pmatrix} r \\ d_i \\ n \end{pmatrix} \\
&= \begin{bmatrix} (I+FGK)^{-1} & -FG(I+KFG)^{-1} & -F(I+GKF)^{-1} \\ K(I+FGK)^{-1} & (I+KFG)^{-1} & -KF(I+GKF)^{-1} \\ GK(I+FGK)^{-1} & G(I+KFG)^{-1} & (I+GKF)^{-1} \end{bmatrix}
\end{aligned}
\tag{2.4}
$$

Remark 2.2: Theorem 2.4 also guarantees well-posedness of the closed loop system.

Remark 2.3: Theorem 2.4 includes other special cases where at least one of the transfer matrices is stable (page 69) [74]! One special case is Corollary 2.3.

Corollary 2.3: Suppose $K, G, F \in \mathcal{RH}_\infty$. Then the system in Figure 2.1 is (internally) stable iff $(I + FGK)^{-1} \in \mathcal{RH}_\infty$, or equivalently $det(I + FGK)$ has no zeros in the closed right-half plane.

It is also interesting to note that it is possible that all transfer matrices K, G, F are unstable but the closed loop system is stable! Thus, the condition $K, G, F \in \mathcal{RH}_\infty$ in the Corollary 2.3 just simplifies the calculations and it's neither a necessary nor a sufficient condition for the stability. Although unstable controllers (due to instability problems in sensor failure) are usually undesirable but it can be shown that in some cases are unavoidable.

[5] BIBO (bounded input, bounded output) Stability: All transfer functions defined from any independent input to any definable output be externally stable[61]. Another equivalent definition for externally stable system is a system with bounded impulse response ($\int_0^\infty \|h(t)\|_2 dt < \infty$).

Concluding remarks

Many linear systems arise from nonlinear systems by linearization about an operating point. The relationship between the stability of the linearized model and that of the original model is therefore of great interest. The question that arise here is "How far from the operating point the stability statement for the linearized model is valid for the nonlinear system?". One of the notable achievements of Lyapunov theory is a contribution to this question. It is proved that if A is a stability matrix, then the original nonlinear equation is also stable under *small* perturbations from the equilibrium position and the acceptable range of allowed perturbation can be calculated invoking the following theorem.

Theorem 2.5: Let $\dot{z}(t) = f(z(t))$ and z_e be an equilibrium point ($f(z_e) = 0$), then if $z(t) = x(t) + z_e$ from Taylor expansion we have

$$f(x(t) + z_e) = f(z_e) + Ax(t) + g(x(t))$$
$$\text{where } A \triangleq \left.\frac{\partial f}{\partial x}\right|_{z_e}, \quad \lim_{\|x\|_2 \to 0} \frac{\|g(x(t))\|_2}{\|x\|_2} = 0$$

and the linearized model obtained as follows

$$\dot{x}(t) = Ax(t)$$

Now, if A is a stability matrix, then the original nonlinear equation is also stable under perturbation x iff

$$\frac{2g^T(x(t))Px(t)}{x^T(t)x(t)} < 1$$

provided that P is a positive definite matrix.

Proof overview: Assuming the following Lyapunov function $V(x(t)) \triangleq x^T(t)Px(t)$

$$\begin{aligned} &\Rightarrow \dot{V}(x(t)) = x^T(t)(PA + A^TP)x(t) + 2g^T(x(t))Px(t) \\ (\text{assume } Q = I \text{ in } (2.3)) &= -x^T(t)x(t)\left(1 - \frac{2g^T(x(t))Px(t)}{x^T(t)x(t)}\right) \\ &\Rightarrow \dot{V}(x(t)) < 0 \text{ iff } \frac{2g^T(x(t))Px(t)}{x^T(t)x(t)} < 1 \end{aligned} \tag{2.5}$$

More information on this issue can be found in [11].

The common issue of all approaches for analyzing the stability of LTI systems boil down to analyzing the roots of a polynomial. For instance in the state space approach stability is determined by characteristic polynomial $det(A - sI)$ and in frequency domain (assuming no right-half plane zero pole cancellation when the product FGK is formed) by numerator of $det(I + FGK)$ which is also a polynomial. It is worth also to note that all linear differential equations have homogeneous answer in form of exponentials (including sines and cosines) and the sign of these exponents (modes) determine the stability of the corresponding system. Thus, several methods have been developed to find the sign of these modes (which are roots of characteristic polynomial) without direct calculation. For instance Nyquist criterion, Routh-Hurwitz test and Kharitonov theorem [22] for continuous case and Schur-Cohn test [62] for discrete case.

The idea behind the definition of internal stability is that it is not enough to look only at a specific input-output transfer matrix, for instance from r to y. As an example, we cannot always conclude the internal stability of the whole system from stability of the transfer matrix from r to y. Because still an internal signal (and even y) could become unbounded resulted from another input and probably causing internal damages to the physical system. This idea can also be generalized to performance definition and other analysis of the closed loop system. We should bury in mind that in a closed loop control system we are not only dealing with one transfer matrix, but also with several ones, where each one should fulfill different specs and stability is just one of them.

The most general block diagram of a control system is shown in Figure 2.2. The generalized plant consists of everything that is fixed like the plant, actuators, sensors and so on[6]. The controller consists of the designable part which might be an electric circuit, microcontroller or some other such devices. The signals w, z, y and u are,

[6] The generalized plant even might contain uncertain but bounded elements that in some references the uncertain part is shown with an extra block Δ.

in general, vector-valued functions of time. The components of w are all exogenous inputs like references, disturbances, sensor noises and so on. The components of z are all signals we wish to control: tracking errors between reference signals and plant outputs, actuator signals whose value must be kept between certain limits and so on. The vector y contains all measured and accessible signals. Finally u contains all controlled inputs to the generalized plant. The stability analysis of this configuration is very similar to what we discussed in this section, but we should note that deriving a minimal realization of the generalized plant (preventing zero pole cancellation) plays a crucial role.

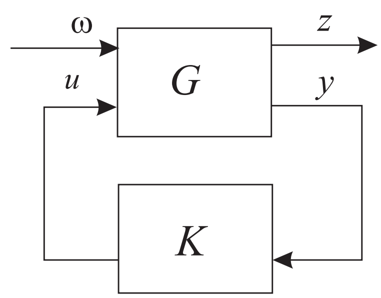

Figure 2.2: General control configuration

2.3 Robust Stability

Noting that the universe of mathematical models from which a model is chosen is distinct from the universe of physical systems, we can conclude that there is always a mismatch between the model and its corresponding physical system so-called *model uncertainty*. We can sometimes reduce this error by adding more complexity to the model, but complexity is not desirable. A good model should be simple enough to facilitate design, yet complex enough to give the engineer confidence that the design based on it will work on the true plant. In this section we would like to conclude the stability of the physical plant (system) in the face of model uncertainties.

Definition 2.4: A system is robust stable if it remains (internally) stable in presence of all allowable uncertainties. As a view of control theory, with a given controller K the system remains stable for all plants in the (prescribed) uncertainty set ([55] p. 303).

2.3.1 Mathematical analysis of robustness

Robustness in classic control

In Figure 2.1 consider $G(s)$ is a second order SISO systems as follows:

$$G(s) = \frac{\omega_n^2}{s(s+2\zeta\omega_n)}$$

and $K(s) = 1$, $F(s) = 1$, where ω_n is the frequency in $\frac{rad}{sec}$ and ζ is the damping. Based on the following five properties the performance of the transfer function $T = \frac{y}{r} = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$ can be evaluated[7].

1. Phase margin: shows the robust stability against phase deviation and time delays.

$$\phi = 2\sin^{-1}(\frac{1}{2\|S_{min}\|}) = \tan^{-1}(2\zeta\omega_n/\omega_c) \approx 100\zeta \quad (degree)$$

   where $\omega_c = \omega_n\sqrt{(4\zeta^4+1)^{1/2} - 2\zeta^2}$ is the open loop bandwidth.

2. Gain margin: shows robust stability against gain deviation (it might have two directions!).

3. The maximum peak of the complementary sensitivity function $\|T\|_\infty = M_{p\omega}$.

$$M_{p\omega} = 1/(2\zeta\sqrt{1-\zeta^2}) \approx \frac{1}{2\sin\phi/2} \quad \zeta < \sqrt{2}/2 \quad (M_{p\omega} = 1 \quad \zeta \geq \sqrt{2}/2)$$

   Note that $M_{p\omega}$ is the absolute magnitude and is not in dB scale.

---

[7] These properties can usually be generalized to higher order systems specially the case of two dominant poles.

4. M_{min} The minimum distance of Nyquist plot from the critical point (M circles).

$$M_{min} = 1/\|S\|_\infty$$

5. D_{min}: The minimum additive model uncertainty ($\min \|\Delta G(s)\|_\infty$) that makes the closed loop unstable.

   Remark 2.4: In [8] it is shown that M_{min} and D_{min} are not sufficient and reliable conditions for robust stability!

The constraints on the phase and gain margin can be presented in graphical format in polar coordinate (Nyquist) as shown in Figure 2.3 so-called Θ region [22].

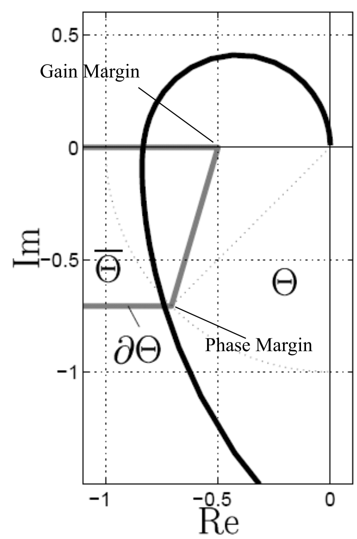

Figure 2.3: Specification of Θ-stability: Nyquist plot must avoid forbidden $\overline{\Theta}$ region

Frequency domain (modern approach)

A powerful tool for analyzing the robust stability is the small gain theorem and its generalized version based on structured singular value. In order to apply this theorem the whole uncertainties (including all structured and unstructured) must be pulled out in form of Δ transfer matrix as shown in Figure 2.4 and the remaining parts form another transfer matrix M[8]. It is clear that the transfer matrix M must be stable to preserve stability in absence of uncertainties and we assume that Δ is also stable! The elements of Δ could represent different uncertainties such as unstructured, structured (independent channels), lumped nonlinearities, time delays and unmodelled dynamics, parameters variation but the best choice of uncertainty representation for a specific model depends on the errors that the model makes.

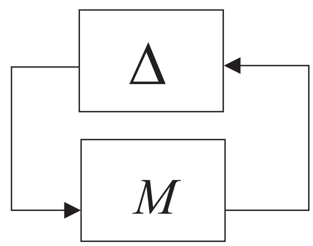

Figure 2.4: M Δ-structure

Theorem 2.6: Assume that the nominal system $M(s)$ and the perturbation $\Delta(s)$ are stable. Then the M Δ system in Figure 2.4 is **robust stable** iff Nyquist plot of $det(I - M(j\omega)\Delta(j\omega))$ does not encircle the origin, for all Δ in the allowed perturbation set $\mathbf{\Delta}$:

$$
\begin{array}{ll}
\Leftrightarrow (det(I - M\Delta))^{-1} \in \mathcal{RH}_\infty & , \mathbf{\Delta} = \{\Delta_1, \Delta_2, \cdots\},\ \Delta_i \text{ stable and norm-bounded} \\
\Leftrightarrow \lambda_i(M(j\omega)\Delta(j\omega)) \neq 1,\ \forall i,\ \forall\omega & , \forall c \in \mathbb{R}, |c| < 1 \text{ if } \Delta_i \in \mathbf{\Delta} \Rightarrow c\Delta_i \in \mathbf{\Delta} \\
\Leftarrow \rho(M(j\omega)\Delta(j\omega)) < 1,\ \forall\omega & , \forall c \in \mathbb{R}, |c| < 1 \text{ if } \Delta_i \in \mathbf{\Delta} \Rightarrow c\Delta_i \in \mathbf{\Delta} \\
\Leftrightarrow \rho(M(j\omega)\Delta(j\omega)) < 1,\ \forall\omega & , \forall c \in \mathbb{C}, |c| < 1 \text{ if } \Delta_i \in \mathbf{\Delta} \Rightarrow c\Delta_i \in \mathbf{\Delta} \\
\Leftrightarrow \rho(M\Delta) \leq \overline{\sigma}(M\Delta) \leq & \text{(small gain theorem)} \\
\quad \leq \overline{\sigma}(M(j\omega))\overline{\sigma}(\Delta(j\omega)) < 1,\ \forall\omega & , \mathbf{\Delta} = \{\Delta |\ \Delta \text{ stable and norm-bounded }\}
\end{array}
$$

[8]Determination and extraction of uncertainties in M Δ-structure is not a trivial procedure and needs experience and knowledge on different class of uncertainties.

$$\Leftarrow \min_{D \in \overline{D}} \overline{\sigma}(DMD(j\omega)^{-1})\overline{\sigma}(\Delta(j\omega)) < 1,\ \forall\omega$$
$$,\boldsymbol{\Delta} = \{diag[\delta_1 I_{r1}, \cdots, \delta_s I_{r_S}, \Delta_1, \cdots, \Delta_F] : \delta_i(s), \Delta_i(s) \in \mathcal{RH}_\infty\}$$
$$\Leftrightarrow \mu_\Delta(M(j\omega))\overline{\sigma}(\Delta(j\omega)) < 1,\ \forall\omega$$
$$,\boldsymbol{\Delta} = \{diag[\delta_1 I_{r1}, \cdots, \delta_s I_{r_S}, \Delta_1, \cdots, \Delta_F] : \delta_i(s), \Delta_i(s) \in \mathcal{RH}_\infty\}$$

where $\overline{D}$ is the set of block diagonal matrices whose structure is compatible to that of Δ, i.e. $\Delta D = D\Delta$.

Remark 2.5: The relationship between loop transfer matrix, close loop characteristic polynomial (ϕ_{cl}), open loop characteristic polynomial (ϕ_{ol}) is as follows:

$$det(I - M\Delta(j\omega)) = \prod_i \lambda_i(I - M\Delta(j\omega)) = \prod_i (1 + \lambda_i(M\Delta(j\omega))) = \frac{\phi_{cl}(s)}{\phi_{ol}(s)}.c$$

where is c is just a constant ([55] p. 145).

Remark 2.6: Δ should be convex[9] but it can be a nonlinear and time varying stable operator ([74] p. 139).

Remark 2.7: Small gain theorem is applicable to any norm with submultiplicative property ($\|AB\| \leq \|A\|\|B\|$).

Remark 2.8: The $\mathcal{H}_\infty$ version of small gain theorem is very conservative (because it ignores the structure of uncertainties). A remedy proposed for this problem is gathering all uncertainties in one point which is very sensitive when the condition number of the plant transfer matrix is big ($\gamma(G) = \overline{\sigma}(G)/\underline{\sigma}(G)$). (See [55] p. 239)

Remark 2.9: The structured singular value is bounded as follows:

$$\rho(M) \leq \max_{U \in \overline{U}} \rho(UM) \leq \mu_\Delta(M) \leq \inf_{D \in \overline{D}} \overline{\sigma}(DMD^{-1}) \leq \overline{\sigma}(M)$$

where $\overline{U}$ is the set of all unitary matrices U with the same block-diagonal structure as Δ. The lower bound can have multiple local maxima that are not global. Thus

[9] The set $\mathcal{A}$ is convex if $a_i, a_j \in \mathcal{A} \Rightarrow \forall k,\ 0 \leq k \leq 1,\ (ka_i + (1-k)a_j) \in \mathcal{A}$.

local search cannot be guaranteed to obtain μ. The upper bound can be formulated as a convex optimization problem (D-iteration), so the global minimum can be found. But unfortunately the (minimum of) upper bound is equal to μ iff $2S + F \leq 3$ (in Δ structure). The DK-iteration which is used for controller design in μ synthesis is not a convex optimization problem!

Time domain (modern approach)

In the time domain, robust stability of the closed loop system is analyzed by considering the effect of (structured) uncertainties on the closed loop characteristic polynomial. The main results are based on Hurwitz and Kharitanov[10] stability analysis. One of the famous approaches in this category is the parameter space. This approach is a powerful tool (but graphics-oriented) for structured uncertainties (e.g. parameter uncertainties in the transfer function) but practically its efficiency is restricted to few number of parameters (up to three)[22].

Sensitivity in the context of robustness

Sensitivity can be generally defined as follows:

$$S_x^f = \frac{\partial f/f}{\partial x/x}$$

where f is a function of x. As an example using this formula we can evaluate the sensitivity of closed loop transfer function to variation of different loop elements (controller, plant, feedback elements and etc.). For instance it can be verified that the closed loop transfer function is very sensitive to the feedback elements (specially at loop high gain frequency we have $S_x^f \approx 1$) and therefore high accuracy elements must be chosen to realize the feedback path.

From robust stability point of view the sensitivity of internal stability to the variation

[10] More information on pages 19-21 and 341-344 in [22].

of different loop elements can be verified by exploiting the small gain theorem. In this way we can check the tolerable uncertainty in each loop element in view of stability. For instance any high gain element (at any frequency) increases the sensitivity of the loop, specially when it is followed with small gain elements.

2.4 Performance

The performance for a feedback system has three major items which are *reference tracking*, *disturbance rejection* and *noise attenuation*.

Definition 2.5: The performance of a feedback system is defined as the capability to keeping the specified output signals (y) close to the reference signals (r) in presence of any disturbances (d_i, d_o) and noises (n).

Remark 2.10: If the defined set of reference signals has only the element zero, the performance task is called *regulating* which is very common in industrial applications.

2.4.1 Mathematical definition and analysis of performance

Frequency domain

Fig.2.1 shows the possible exogenous input signals and possible measurable points of a feedback system. Thus the list of exogenous input signals is as follows:

1. Reference input: r
2. Disturbance in the plant input: d_i
3. Disturbance in the plant output: d_o
4. Noise in the measurement signals: n

Definition 2.6: The difference between the reference and the output signals is called the error ($e \triangleq r - y$).

Remark 2.11: Sometimes the control error is defined as $e_r = r - y_m$, which only considers the measured value not the actual value!

The transfer matrices that connect the exogenous inputs to the error (e) are shown in Table 2.2 (Reminder: $S_o = (I + FGK)^{-1}$ and $T_o = GK(I + FGK)^{-1}$).

exogenous signals	$e = r - y$	$e_r = r - y_m$
reference r	$I - T_o$	S_o
disturbance in input d_i	$-GS_o$	$-GS_o$
disturbance in output d_o	$-S_o$	$-S_o$
measurement noise n	T_oF	T_oF

Table 2.2: Effect of exogenous signals on the error (e) and measured error (e_r)

Definition 2.7: The performance can be mathematically represented as the rate of attenuation of four exogenous signals (r, d_i, d_o, n) on the error signal (e).

One way to describe the rate of attenuation in the performance of a control system is in terms of the size of the error and input signals (e.g. $\sup_u \frac{\|e\|}{\|u\|}$). There are several ways of defining signal's size (norm of signals) that result in different performance evaluation methods. Suppose u and y are respectively the input and output of the transfer matrix G as shown in Figure 2.5. Table 2.3[11] shows the system gain ($\sup_u \frac{\|y\|}{\|u\|}$)

Figure 2.5: G is the transfer function from u to y

with respect to different signal norms for u and y.

$\sup_u \frac{\|y\|}{\|u\|}$	$\|u\|_2$	$\|u\|_\infty$	$\|u\|_{pow}$	$u(t)=\delta(t)$	$u(t)=\sin(\omega t)$
$\|y\|_2$	$\|G\|_\infty$	∞	∞	$\|G\|_2$	∞
$\|y\|_\infty$	$\|G\|_2$	$\|G\|_1$	∞	$\|G\|_\infty$	$\|G(j\omega)\|$
$\|y\|_{pow}$	0	$\leq \|G\|_\infty$	$\|G\|_\infty$	0	$\frac{1}{\sqrt{2}}\|G(j\omega)\|$

Table 2.3: System gain (see Glossary for the norm definitions)

It can be seen from Figure 2.1 and Table 2.2 that the performance is related to the norm of transfer matrices: S_o, GS_o, T_o and T_oF. It can also be easily verified that it is not possible to shape each of these transfer matrices independently. It means improving one performance index aggravate the other one! The remedy to this conflict is to consider each transfer matrix in its active range of frequency, which it could be carried out by defining proper weights on the exogenous signals. These weights could be considered as frequency filters in front of each input as shown in Figure 2.6. Thus,

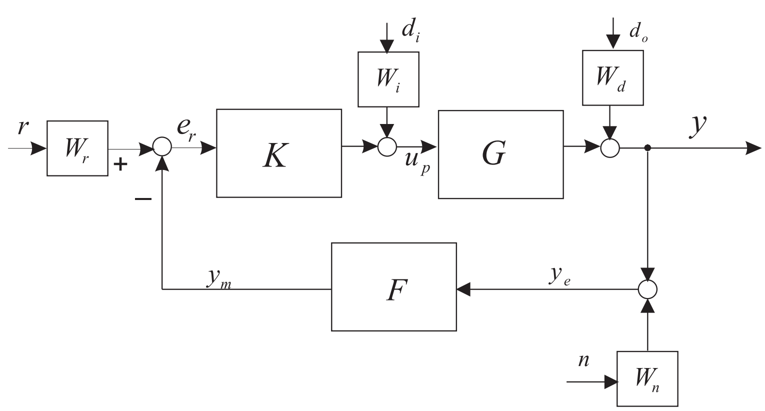

Figure 2.6: Weights on exogenous signals which reflects the frequency range of each one

to reduce the effect of exogenous input signals on the performance index e, the norm

[11]In this table the notations G, u and y are general (see Figure 2.5).

of the corresponding transfer matrices should be very small in the specified frequency range of each input signal. Equivalently, the norm of all weighted transfer matrices should be small in all frequencies. By introducing the weighing functions (easily from Figure 2.6) Table 2.2 will be converted to Table 2.4.

exogenous signals	$e = r - y$	$e_r = r - y_m$
reference r	$(I - T_o)W_r$	$S_o W_r$
disturbance in input d_i	$-GS_oW_i$	$-GS_oW_i$
disturbance in output d_o	$-S_oW_d$	$-S_oW_d$
measurement noise n	T_oFW_n	T_oFW_n

Table 2.4: Effect of exogenous signals on the error (e) and measured error (e_r) with weighting functions

Thus, based on Definiton 2.7 the rate of attenuation (γ) for each exogenous signal will be defined as follows:

1. For reference tracking γ_r must be small (ideal case $\gamma_r = 0$).

$$\sup_r \frac{\|e\|}{\|r\|} \leq \gamma_r \quad \text{(steady state behavior } \|(I - T_o(j\omega))W_r(j\omega)\| \leq \gamma_r,\ \forall\omega)$$

2. For input disturbance rejection γ_i must be small (ideal case $\gamma_i = 0$).

$$\sup_{d_i} \frac{\|e\|}{\|d_i\|} \leq \gamma_i \quad \text{(steady state behavior } \| - G(j\omega)S_o(j\omega)W_i(j\omega)\| \leq \gamma_i,\ \forall\omega)$$

3. For output disturbance rejection γ_o must be small (ideal case $\gamma_o = 0$).

$$\sup_{d_o} \frac{\|e\|}{\|d_o\|} \leq \gamma_o \quad \text{(steady state behavior } \| - S_o(j\omega)W_d(j\omega)\| \leq \gamma_o,\ \forall\omega)$$

4. For noise attenuation γ_n must be small (ideal case $\gamma_n = 0$).

$$\sup_n \frac{\|e\|}{\|n\|} \leq \gamma_r \quad \text{(steady state behavior } \|T_o(j\omega)F(j\omega)W_n(j\omega)\| \leq \gamma_n,\ \forall\omega)$$

where Ws are proper weight transfer matrices.

In some cases the requirements on the shape of (performance) transfer matrices are conflicting. This discord, roots in the configuration and characteristic of different transfer matrices. This issue is discussed comprehensively in [55] under constraints on performance targeting. The performance analysis always have to be done for a defined set of reference signals, disturbances and noises. It is worth noting that (using configuration shown in Figure 2.1), it is not possible to attenuate a noise at the same frequency that the reference signal has to be followed! Some rules of thumb: For disturbance rejection at plant output (y), in general, large $\underline{\sigma}(L_o)$ is required for (d_o) and large enough controller gain $\underline{\sigma}(K)$ for d_i. Hence in this case good multivariable feedback design boils down to achieving high loop (and possibly) controller gain in the necessary frequency range.

Remark 2.12: With feed-forward block in Table 2.2 the tracking transfer matrix changes to $\frac{e(s)}{r(s)} = I - T_o + GK_F S_o$, where K_F is the feedforward block.

$\mathcal{H}_2$ performance

The performance in the sense of $\mathcal{H}_2$ means minimizing the effect of output disturbance (d_o) on the error signal, exploiting second and infinity norm for signals d_o and e respectively. Using Table 2.2 and element (2,1) in Table 2.3, we can easily find out that this cost function which should be minimized is as follows:

$$\sup_{d_o} \frac{\|e\|_\infty^2}{\|d_o\|_2^2} = \|S_o\|_2^2 (\Rightarrow \|W_e S_o W_d\|_2^2 \text{ assuming some weights })$$

and to avoid saturation of actuators (constraints) and restricting the inputs' energy, we should add the following term to the cost function with proper compromise factor (ρ):

$$\sup_{d_o} \frac{\|u_p\|_\infty^2}{\|d_o\|_2^2} = \|KS_o\|_2^2 (\Rightarrow \|W_u K S_o W_d\|_2^2 \text{ assuming some weights })$$

therefore the final cost function is as follows:

$$\begin{aligned}
\sup_{d_o}(\frac{\|e\|_\infty^2}{\|d_o\|_2^2} + \rho^2\frac{\|u_p\|_\infty^2}{\|d_o\|_2^2}) &= \left\| \begin{bmatrix} S_o \\ \rho K S_o \end{bmatrix} \right\|_2^2 \\
(\text{ assuming some weights }) &\Rightarrow \left\| \begin{bmatrix} W_e S_o W_d \\ \rho W_u K S_o W_d \end{bmatrix} \right\|_2^2
\end{aligned}$$

An identical performance cost function will be derived if we try to minimize the effect of special type of disturbance (d_o) (impulse) on the error function in the stochastic framework. Assume that the disturbance d_o can be approximately modeled as an impulse with random input direction; that is $d_o(t) = \eta\delta(t),\ E(\eta\eta^*) = I$ where E denotes the expectation. We define the cost function as follows (Reminder: element (1,4) in Table 2.3 shows the effect of impulse input on the output) :

$$E\{\|e\|_2^2\} = \|S_o\|_2^2(\Rightarrow \|W_e S_o W_d\|_2^2 \text{ assuming some weights })$$

and to avoid saturation of actuators (constraints) we add the following term to the cost function with proper compromise factor (ρ):

$$E\{\|u_p\|_2^2\} = \|KS_o\|_2^2(\Rightarrow \min \|W_u K S_o W_d\|_2^2 \text{ assuming some weights })$$

therefore the performance is achieved by minimizing the following cost function:

$$\begin{aligned}
E\{\|e\|_2^2 + \rho^2\|u_p\|_2^2\} &= \left\| \begin{bmatrix} S_o \\ \rho K S_o \end{bmatrix} \right\|_2^2 \\
(\text{ assuming some weights }) &\Rightarrow \left\| \begin{bmatrix} W_e S_o W_d \\ \rho W_u K S_o W_d \end{bmatrix} \right\|_2^2
\end{aligned}$$

Remark 2.13: The relationship between the above definition and standard LQR is described in [74] on page 254.

Remark 2.14: Another aspect of LQR is increasing the fidelity of linearized model to the original nonlinear system, by proper choice of the weight matrices. This is done by minimizing the second order effects in the linearized model resulted form Taylor series [7].

$$\min \int_0^{t_f} \delta x Q \delta x + \delta u R \delta u dt$$

The $\mathcal{H}_2$ problem can be presented in a more general format. In Figure 2.7 w represents all exogenous signals such as reference, disturbance and sensor noises and z all signals that we wish to control or confine, such as error signal ($e = r - y$) and actuator signals (u_p). The performance can be verified by minimizing the $\mathcal{H}_2$ norm of the transfer matrix T_{zw}.

$\mathcal{H}_\infty$ performance

The performance in the sense of $\mathcal{H}_\infty$ means minimizing the effect of output disturbance (d_o) on the error signal, exploiting second (or *pow*) norm for signals d_o and e. Invoking Table 2.2 and element (1,1) (or (3,3)) in Table 2.3 (similar to $\mathcal{H}_2$) the performance in $\mathcal{H}_\infty$ sense will be achieved by solving the following optimization problem:

$$\begin{aligned} \min_u \sup_{d_o} \{ \frac{\|e\|_2^2}{\|d_o\|_2^2} + \rho^2 \frac{\|u\|_2^2}{\|d_o\|_2^2} \} \quad &= \quad \left\| \begin{bmatrix} S_o \\ \rho K S_o \end{bmatrix} \right\|_\infty^2 \\ (\text{ assuming some weights }) \quad &\Rightarrow \quad \left\| \begin{bmatrix} W_e S_o W_d \\ \rho W_u K S_o W_d \end{bmatrix} \right\|_\infty^2 \end{aligned}$$

The $\mathcal{H}_\infty$ problem can be presented in more general format using the configuration in Figure 2.7. The performance can be verified by minimizing the $\mathcal{H}_\infty$ norm of the transfer matrix T_{zw}.

Performance in classic control (frequency domain)

In Figure 2.1 consider $G(s)$ is a second order SISO system as follows:

$$G(s) = \frac{\omega_n^2}{s(s + 2\zeta\omega_n)}$$

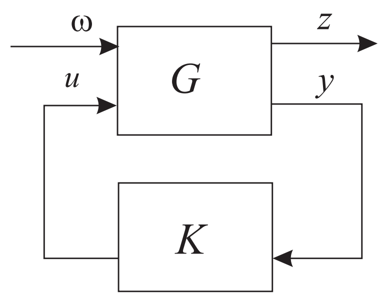

Figure 2.7: General configuration, $T_{z\omega} = \mathcal{F}_l(G, K)$

and $K(s) = 1$, $F(s) = 1$. Based on the following seven properties the performance of the transfer function $T(s) = \frac{y}{r} = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$ can be evaluated[12].

1. Bandwidth[13] in terms of $T(s)$ is defined as the highest frequency at which $|T(j\omega)|$ crosses $1/\sqrt{2}$ ($\approx -3dB$) from above. This frequency shows the range that the closed loop system is effective, which is usually required to be bigger than some specified frequency but it is restricted by physical constraints.

$$\omega_{BT} = \omega_n \sqrt{(2 - 4\zeta^2 + 4\zeta^4)^{1/2} + 1 - 2\zeta^2}$$

Remark 2.15: Crossover frequency is defined as the frequency at which $|L(j\omega)|$ first crosses $1/\sqrt{2}$ ($\approx -3dB$) from above. In practice ω_c is a substitute (an approximation) for ω_{BT} in the controller design, because it is directly affected by the controller frequency shape. Therefore it is more convenient to design a controller that achieve an specified open loop bandwidth ω_c and *hope* that ω_{BT} is in the acceptable range. Again for a second order SISO we have

$$\omega_c = \omega_n \sqrt{(4\zeta^4 + 1)^{1/2} - 2\zeta^2}, \quad (\omega_c < \omega_{BT} < 2\omega_c,\ \omega_{BT} \approx 1.5\omega_c,)$$

[12]These properties can usually be generalized to higher order systems specially the case of two dominant poles.

[13]The closed-loop bandwidth, ω_B, is the frequency at which $|S(j\omega)|$ first crosses $1/\sqrt{2}$ ($\approx -3dB$) from below.

2. The maximum peak of the complementary sensitivity function $M_{p\omega} = \|T\|_\infty$.

$$M_{p\omega} = \frac{1}{(2\zeta\sqrt{1-\zeta^2})} \approx \frac{1}{2\sin\phi/2} \quad \zeta < \frac{\sqrt{2}}{2} \quad (M_{p\omega} = 1 \quad \zeta \geq \frac{\sqrt{2}}{2})$$

   where ϕ is the phase margin. High $M_{p\omega}$ causes overshoot in the time response therefore it is usually required to be close to one. Note that $M_{p\omega}$ is the absolute magnitude and is not in dB scale.

3. Rising time (step response): The time it takes for the output to first reach 90%, from 10% of its final value,which is usually required to be small.

$$t_r \approx \frac{2.2}{\omega_{BT}} \quad sec$$

   Note that ω_{BT} is in $\frac{rad}{sec}$.

4. Settling time (step response): the time after which the output remains within ±2% of its final value, which is usually required to be small.

$$t_s \approx \frac{4.6}{\zeta\omega_n} \quad sec$$

   Note that ω_n is in $\frac{rad}{sec}$.

5. Overshoot (step response): the peak value divided by the final value, which should be typically less than 1.2 (20%).

$$M_{pt} = 1 + e^{-\pi\zeta/\sqrt{1-\zeta^2}}, \quad (\zeta < \frac{\sqrt{2}}{2})$$

6. Decay ratio (step response): ratio of second peak to the first one which normally less than 0.3 is acceptable.

7. Steady state offsets (step response): the difference between final and desired final value, which is usually required to be small.

The constraints on ζ and ω_n can be presented in graphical format in the complex plane as shown in Figure 2.8 so-called Γ region [22].

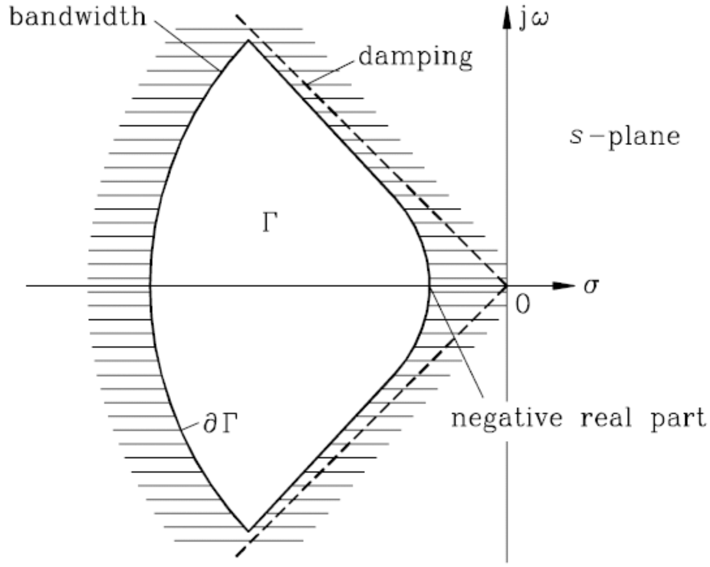

Figure 2.8: The pole region Γ that guarantees damping, negative real part and bandwidth limitation.

Remark 2.16: These specification are also useful in constructing proper second order weights in order to define performance. For instance it is possible to keep S_o of a feedback system beneath $\gamma W_d^{-1}(\zeta, \omega_n)$ to have similar step response to a second order system (without the coefficient $\gamma \geq 1$ is not possible!).

Performance in classic control (time domain)

One idea for defining the performance in the time domain is the quality of response to a specified input ($r(t)$). That is, for a given command $r(t)$ (which is zero for $t < 0$), the ideal controller is the one that generates the plant input $u(t)$ (zero for $t < 0$) which minimizes[14]

$$ISE = \int_0^\infty \|y(t) - r(t)\|^2 dt$$

[14] ISE stands for Integral Square Error.

This controller is ideal in the sense that it may not be realizable because the cost function includes no penalty on the input $u(t)$ [48]. Another disadvantage, is that it only contains tracking performance which is not adequate.

Another common approach is step response specifications such as rising time, settling time and overshoot for step response. These specifications cannot be directly used for design purposes therefore the usual way is to translate them in frequency domain and reaching the proper performance by reforming the poles (in SISO case root locus) in Γ region shown in Figure 2.8. Some of these relationships are given in the previous subsection.

Remark 2.17: The step response is not always a suitable reference to measure the performance! In frequency domain it can be shown that it is not possible to fulfill the performance specs for all frequencies. Therefore the desirable performance can be achieved (and must be defined) for a specific set of reference signals (with specified frequency range). The frequency spectrum of step signal is $\frac{1}{j\omega} + \pi\delta(\omega)$ which drops off with the slope of $-20dB/decade$ and has crossover frequency of 1 rad/sec which might be so wide for this purpose.

2.5 Design Steps

Modern control design packages often fall short of offering what is truly practical. A great many physical plants are modeled with high order equations. If we use a typical commercial software design package, we will typically produce, using robust control or $\mathcal{H}_2$ design methods, a controller which is also high order. In the following, typical control design steps are shown [19]:

1. Studying the system to be controlled and decide what types of sensors and actuators will be used and where they will be placed.

2. Modeling the resulting system to be controlled.

3. Simplifying the model if necessary so that it is tractable*.

4. Determining the properties of the model.

5. Deciding on performance specifications.

6. Deciding on type of controller to be used and design to meet the specs, if not modify specs or type of controller.

7. Simplifying the controller if necessary so that it is implementable*.

8. Simulating the resulting controller on a computer or in a pilot plant and repeat from step 1 if it's necessary.

9. Choosing the hardware and software to implement the controller.

10. Tuning the controller in presence of practical situation

The steps marked with an asterisk (*) show places where reduction approaches could be helpful tools. These are the places that we can tackle the problem of complexity. The focus of our research in this thesis is to suggest a complementary tool for the existing reduction methods to preserve the stability of the reduced model and the closed loop.

Chapter 3

ORDER REDUCTION

3.1 Introduction

Accurate modeling is a necessary part in all fields of engineering dealing with physical systems. Nowadays to achieve an accurate model, powerful computers and newly developed methods and algorithms are put into service. The task of modeling typically leads to approximate the behavior of a real system by a set of ordinary differential equations. To model a physical behavior with a good accuracy, a high order differential equation is unavoidable and this is in conflict with design engineer requirements. Because a good model should be simple enough to facilitate design, yet complex enough to give the engineer confidence that the design based on it work on the true plant. The main goal of model reduction is to find the best compromise between accuracy and simplicity. Several methods have been proposed for model reduction of linear systems in different fields like control engineering, microsystems and applied mathematics. These methods are mostly based on minimization of some predefined error functions, deleting the less important states or matching some of the parameters of the original and reduced systems. A well-accepted method among the existing methods is balancing and truncation which was first proposed by Moore [47]. A brief summary of balancing and its properties are given in the next sections.

3.2 Truncation, Singular Perturbation and Projection

One approach to obtain reduced order systems is to split the states into two sets so-called dominant and nondominant states. For designating the set of dominant state variables, there are different methods but among them balancing has been very successful approach that we discuss it in more details in the next section. After finding the dominant states, the next step is to trim the nondominant part from the equations. For removing the nondominant state variables, two approaches are common. The first one so-called truncation is done by trimming the corresponding rows and columns of the nondominant state variables as shown below($G \rightarrow G_{tr}$). The second approach so-called singular perturbation (also called residualization) does the similar thing but it also considers the steady state effect of the nondominant state variables on the rest as shown below ($G \rightarrow G_{re}$).

$$G(s) = \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & D \end{array}\right] \rightarrow \begin{array}{l} G_{tr}(s) = \left[\begin{array}{c|c} A_{11} & B_1 \\ \hline C_1 & D \end{array}\right] \\ G_{re}(s) = \left[\begin{array}{c|c} A_{11} - A_{12}A_{22}^{-1}A_{21} & B_1 - A_{12}A_{22}^{-1}B_2 \\ \hline C_1 - C_2A_{22}^{-1}A_{21} & D - C_2A_{22}^{-1}B_2 \end{array}\right] \end{array}$$

Some interesting properties of these two reduced systems are as follows:

$$G_{tr}(0) \neq G(0), G_{tr}(\infty) = G(\infty), \quad G_{re}(0) = G(0), G_{re}(\infty) \neq G(\infty)$$

Finding dominant states as a linear combination of original states and truncating the dominant states can be realized in one step using projection matrices as follows:

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \rightarrow G^*_{tr}(s) = \left[\begin{array}{c|c} LAR & LB \\ \hline CR & D \end{array}\right]$$

where $LR = I_r$. Noting that we can always find $\overline{L}$ and $\overline{R}$ such that $\begin{bmatrix} L \\ \overline{L} \end{bmatrix} [R \quad \overline{R}] = I_n$ therefore $\begin{bmatrix} L \\ \overline{L} \end{bmatrix}$ represents the similarity transformation T that appoints the dominant states.

Remark 3.1: The entire class of $k_{th}(< n)$[1] order system is larger than those which can be obtained by projection matrices (subsystem elimination). The properties of the projection relationship between two models of different orders have been investigated in [30]. In the case of square systems, any reduced model whose order is less than $n-m$ can be obtained by infinitely many projections, any reduced model whose order is greater than $n-m$ cannot be obtained by projection, and the case of exactly $n-m$ depends on the eigenvalue structure of a certain pencil.

3.3 Balancing and Its Virtues

A system is balanced iff the controllability and observability Gramians (P, Q) are equal and diagonal ($P = Q = diag(\sigma_1, \cdots, \sigma_n)$). The Gramians satisfy the following two Lyapunov equations:

$$AP + PA^T + BB^T = 0$$
$$A^TQ + QA + C^TC = 0$$

In a balanced system if σ_i slope critically downward, it is possible to split the system in two dominant and weak subsystems. Approximation in this basis could take place by truncation or singular perturbation. The common approach for evaluating balanced basis is Schur method. This method includes two Lyapunov equations (or one Sylvester), two Schur decompositions and one singular value decomposition to construct the reduced order system.

$$\begin{array}{lllll}
\text{1) Lyap. equ.} & AP + PA^T + BB^T = 0 & & & \\
\text{2) Lyap. equ.} & A^TQ + QA + C^TC = 0 & \Rightarrow & T & = S^{-1/2}U_L^TV_A^T, \\
\text{3) Schur dec.} & V_A^TPQV_A = S_{asc} & & T^{-1} & = V_DU_RS^{-1/2} \\
\text{4) Schur dec.} & V_D^TPQV_D = S_{dec} & & x_b & = Tx \\
\text{5) svd} & V_A^TV_D = U_LSU_R^T & & &
\end{array}$$

where x_b is the new (balanced) state vector and x is the original state vector. To obtain the balanced reduced system it is even sufficient to make Gramian matrices

[1] n and k are the order of the original and reduced model respectively and m is the input dimension.

block diagonal and equal ($P = Q = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}$)!

Remark 3.2: If σ_is don't repeat, the balancing transformation matrix is (up to multiplication by a diagonal matrix with ± 1 on the diagonal) unique [47]. For more details on uniqueness see [5] (p. 176).

Remark 3.3: This approach is also applicable to discrete systems, using bilinear transformation ($z = \frac{\alpha+s}{\alpha-s}$). For numerical issues and other algorithms see [5] (p. 184-185).

Remark 3.4: Balancing is only meaningful for stable transfer functions. In order to reduce an unstable system, one idea is to reduce its stable part and add the unstable part in the end. A better idea is to exploit coprime factorization and reduce each (apparently stable) factor separately.

Remark 3.5: In order to match at low frequencies we can substitute s with $1/s$ and then carry out balancing and truncation and in the end retrieve the reduced system by inverse substitution ($1/s \to s$). It can be shown that this approach is equivalent to balanced singular perturbation.

Remark 3.6: There are some other types of balancing, e.g. bounded real balancing and positive real balancing. For more details see [5] (p. 190).

Generalized Graminas

In [73] it is proved that if

$$A\Sigma + \Sigma A^T + BB^T \leq 0$$
$$A^T\Sigma + \Sigma A + C^T C \leq 0$$

where $\Sigma = diag(\sigma_1, \cdots, \sigma_n)$, then the truncated system (G_r) is a balanced truncated of the extended original system (with some extra virtual inputs and outputs) and still satisfies the error bound criterion, which means

$$\|G - G_r\|_\infty \leq 2 \sum_{i=k+1}^{n} \sigma_i$$

Based on this property the matrices $\hat{P}$ and $\hat{Q}$ which satisfy

$$A\hat{P} + \hat{P}A^T + BB^T \leq 0$$
$$A^T\hat{Q} + \hat{Q}A + C^TC \leq 0$$

are called generalized Gramians.

B&T from stability point of view

Assume that the original system is stable and balanced ($P = Q = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}$). If Σ_1 and Σ_2 have no diagonal entries in common, then the truncated system is stable, controllable and observable. The argument on this issue is based on the fact that the reduced model satisfies a similar Lyapunov equation ($A_{11}\Sigma_1 + \Sigma_1 A_{11}^T = B_1 B_1^T$) and therefore the stability will be preserved. More discussion and proves can be found in [52].

B&T from energy point of view

The smallest amount of energy needed to steer the following system

$$\mathcal{S} : \begin{cases} \dot{x} &= Ax + Bu \\ y &= Cx \end{cases} \tag{3.1}$$

from $x = 0$ to $x = x(t_f)$ is equal to $E_i^2 = x(t_f)^T P^{-1}(0, t_f)x(t_f)$ and the energy obtained by observing the output of the system with initial condition x_0 assuming $u = 0$ is given by $E_o^2 = x_0^T Q x_0$. Thus, one way to reduce the number of states is to eliminate those which require a large amount of energy (E_i) to be reached and yield small amounts of observations energy (E_o). However this concept is basis dependent, and therefore in order for such a scheme to work, one would have to look for a basis which two concepts are equivalent, which means finding a balanced basis such that:

$$P = Q = diag(\sigma_1, \cdots, \sigma_n)$$

where the σ_i are the Hankel singular values of the system. As it mentioned in a balanced system if σ_i slope critically downward, it is possible to split the system in two dominant and weak subsystems. In this case the dominant subsystem absorbs and releases more amount of energy in comparison to the weak one (see [47], p. 27-28).

B&T from optimality point of view

The following theorem shows how much balancing is close to optimal in the sense of $\mathcal{H}_2$ norm.

Theorem 3.1: Suppose $G_r = \left[\begin{array}{c|c} A_r & B_r \\ \hline C_r & D_r \end{array}\right]$ solves the following optimal model-reduction problem.

$$\min_{G_r} \|G - G_r\|_2$$

Then there exists nonnegative definite matrices $\hat{Q}, \hat{P} \in \mathbb{R}^{n \times n}$ such that

$$\begin{aligned} rank(\hat{Q}) = rank(\hat{P}) &= rank(\hat{Q}\hat{P}) = n_r < n \\ \tau(A\hat{Q} + \hat{Q}A^T + BB^T) &= 0 \\ (A^T\hat{P} + \hat{P}A + C^TC)\tau &= 0 \end{aligned}$$

where $\tau \in \mathbb{R}^{n \times n}$, is an oblique projection matrix constructed from special factorization of $\hat{Q}$ and $\hat{P}$ (for more details see [32]).

Note that the theorem consists of necessary conditions in the form of two modified Lyapunov equations (similar to controllability and observability Gramians in balancing) plus rank conditions (similar to small Hankel singular values in balancing).

The transformation for balancing ($x_b = Tx$) is also the solution to the following optimization problem:

$$\min_{T} trace[TPT^T + T^{-T}QT^{-1}]$$

and obviously the minimum is equal to $2\sum_i \sigma_i$.

B&T from principle components point of view

The set of differential equations in (3.1) is a linear operator from $\mathcal{L}_2[0, t_f]$ to $\mathbb{R}^n$ (considering the states at time t_f as the output)[9, 41], i.e.,

$$\begin{array}{l} \mathcal{S} : \mathcal{L}_2[0, t_f] \rightarrow \mathbb{R}^n \\ x(t_f) = \mathcal{S}(u(t)) = \int_0^{t_f} e^{A\tau} Bu(t_f - \tau) d\tau \end{array}$$

and $e^{At}B$ can be decomposed with $w_1, \cdots, w_n$ used as orthonormal basis vectors of $\mathbb{R}^n$ and $f_1(t), \cdots, f_n(t)$ as orthonormal basis (functions) of $\mathcal{L}_2[0, t_f]$, i.e.,

$$e^{At}B = \left[\begin{array}{ccc} w_1 & \cdots & w_n \end{array}\right] \left[\begin{array}{ccc} \sigma_{c_1} & & 0 \\ & \ddots & \\ 0 & & \sigma_{c_n} \end{array}\right] \left[\begin{array}{ccc} f_1(t) & \cdots & f_n(t) \end{array}\right]^T$$

This decomposition is called principal component decomposition, which is similar to singular value decomposition of matrices. Similarly, for Ce^{At} (which maps states to output) we have,

$$Ce^{At} = \left[\begin{array}{ccc} g_1(t) & \cdots & g_n(t) \end{array}\right] \left[\begin{array}{ccc} \sigma_{o_1} & & 0 \\ & \ddots & \\ 0 & & \sigma_{o_n} \end{array}\right] \left[\begin{array}{ccc} v_1 & \cdots & v_n \end{array}\right]^T$$

Principal components of $e^{At}B$ and Ce^{At} show the controllable and observable space of state variables with respect to impulse input and they depend on the internal coordinate system. By balancing the system, the directions of these two spaces will be aligned and the corresponding component magnitude become equal [47], i.e.,

$$\left[\begin{array}{ccc} w_1 & \cdots & w_n \end{array}\right] = \left[\begin{array}{ccc} v_1 & \cdots & v_n \end{array}\right] = I_n, \quad \sigma_{o_i} = \sigma_{c_i} = \sigma_i$$

Remark 3.7: Balancing can be carried out with time limited Gramians $(P(0,t_f), Q(0,t_f))$. These quantities are positive semi-definite and therefore qualify as Gramians. They also satisfy Lyapunov equations. Balancing in this case is obtained by simultaneous diagonalization of time limited Gramians. The reduced model obtained by truncation in this basis, and the impulse response of the reduced model is expected to match that of the full model in the specified time interval. However the reduced model is not guaranteed to be stable, this can be fixed as shown in [29].

B&T from error bound point of view

One of the advantages of balancing based on reduction is its a priori $\mathcal{H}_\infty$ error bound based on Hankel singular values, which is valid for both truncation and singular perturbation.

$$\sigma_{k+1} \leq \|G - G_r\|_\infty \leq 2 \sum_{i=k+1}^{n} \sigma_i$$

The lower bound of the infinity norm (σ_{k+1}) shows the best approximation of a k_{th} order system. Conservatism of the infinity norm is discussed in [50] (p. 10) and it is shown that approximation will be difficult when the system is all-pass or strictly proper part of an all-pass system. Some more error bounds for the balanced truncated system are ([5], p. 182):

$$\begin{aligned} \|G - G_r\|_1 &\leq 2 \sum_{i=k+1}^{n} (2i-1)\sigma_i \\ \|G - G_r\|_2 &= trace[(B_2 B_2^T + 2Y_2 A_{12})\Sigma_2] \end{aligned}$$

Chapter 4

LOW ORDER CONTROLLER

4.1 Introduction

A fundamental issue that was studied in both classical and modern control theory is the internal stabilization of systems by means of feedback. It is generally acknowledged that the paper of Youla et al. [69] which gives a characterization of all stabilizing controllers was a milestone in this study[1]. One of the major impact of Youla parameterization (also called YJBK parameterization) on control theory is isolating the problem of optimizing performance and robustness from the problem of guaranteeing the internal stability of the closed loop system. Section 4.2 shows the structure of all stabilizing controllers based on Youla parameterization. Although Youla parametrization is one of the most important breakthroughs in modern control theory, it has the drawback that the order of the stabilizing controller varies as the optimization search is performed. Therefore the first question that arises is the minimum order required to guarantee the closed loop stability (ignoring performance!). To answer this question a non-convex optimization problem must be solved. Section 4.3 summarizes the results on this issue. As it is mentioned, the modern approaches for control design carry out an objective search based on a cost function (optimization problem) among the stabilizing controllers. In fact, the order of the optimal controller, resulted this way, is almost always quite high, being comparable to that of the plant. The second question that arise here is the minimal order controller that guarantees stability and a specified performance criterion. The answer to this question is straight forward,

[1]Another formulation of Youla parameterization and conditions is given in [2].

normally if we look for the optimal answer it is unique and results in a high order controller. Therefore the task of controller reduction is to compromise between the optimality and complexity. Section 4.4 presents some of the existing methods for controller reduction and their impact on the closed loop characteristics.

4.2 Stabilizing Controllers

The very first problem that needs to be focused is to stabilize a given plant with a controller, with preferably as low order as possible. Given a plant G, this section presents a general framework that parameterizes all controllers K that (internally) stabilize G.

Definition 4.1: coprime factorization[2]: A useful way of representing systems is the coprime factorization which may be used both in state space and transfer function form. In the latter case a right coprime factorization (*rcf*) of G is

$$G = N_r M_r^{-1} \text{ over } \mathcal{RH}_\infty,$$

where N_r and M_r are stable coprime transfer matrices. Mathematically, coprimeness means that there exist stable X_r and Y_r such that the following Bezout identity is satisfied

$$X_r N_r + Y_r M_r = I$$

Similarly, a left coprime factorization (*lcf*) of G is

$$G = M_l^{-1} N_l \text{ over } \mathcal{RH}_\infty,$$

Here N_l and M_l are stable and coprime, that is, there exist stable transfer matrices X_l and Y_l such that the following Bezout identity is satisfied

$$N_l X_l + M_l Y_l = I$$

[2]To understand the reason behind this special representation see [19] (p. 59).

In *rcf* the stability implies that N_r should contain all right half plane poles of G, and M_r should contain as right half plane zeros all the right half plane poles of G. The comprimeness implies that there should be no common right half plane zeros in N_r and M_r which result in pole-zero cancellations when forming $N_r M_r^{-1}$. Similar properties can be proved for *lcf*. The coprime factorization is not unique and it can be derived more easier from state space realization by solving an algebraic Riccati Equation [61].

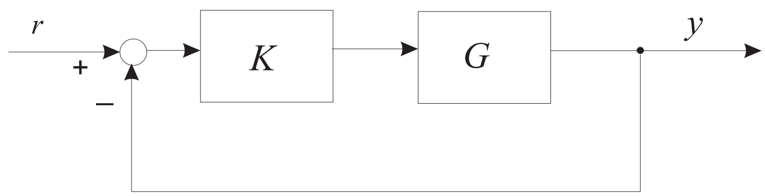

Figure 4.1: Negative feedback system

Theorem 4.1: All stabilizing controllers K with the configuration shown in Figure 4.1 can be parametrized using the following two approaches:

- Coprime factorization approach [71, 72] (Youla parameterization):

  $K = (Y_r - QN_l)^{-1}(X_r + QM_l)$
  where Q ranges over $\mathcal{RH}_\infty$ such that $det(I + Y_l^{-1}N_rQ)(\infty) \neq 0$
  or
  $K = (X_r + M_rQ)(Y_l - N_rQ)^{-1}$
  where Q ranges over $\mathcal{RH}_\infty$ such that $det(I + QN_lY_r^{-1})(\infty) \neq 0$

- State space approach [40]: Assume $G(s) = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right]$. Let F and L be such that $A + LC_2$ and $A + B_2F$ are stable, then all controllers that stabilize G have the configuration shown in Figure 4.2.

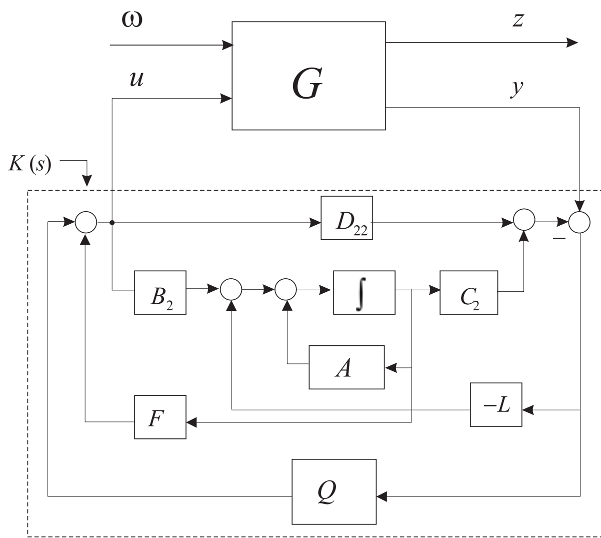

Figure 4.2: Structure of all stabilizing controllers

Based on state space approach in Theorem 4.1, every stabilizer controller amounts to adding stable dynamics to the plant and then stabilizing the extended plant by means of an observer and state feedback.

Remark 4.1: The set of all closed loop transfer matrices from exogenous inputs to outputs achievable by an stabilizing controller is an affine function of Q[3].

Corollary 4.1: The parameterization of all stabilizing controllers is easy when the plant itself is stable. Suppose $G \in \mathcal{RH}_\infty$ then the set of all stabilizing controllers is described as

$$K = Q(I + GQ)^{-1}$$

for any $Q \in \mathcal{RH}_\infty$ and $I + G(\infty)Q(\infty)$ nonsingular.

Remark 4.2: Given an stabilizer controller K, the free parameter in Youla factorization can be easily calculated. For more details see [74] (p. 230).

[3] T is an affine function of Q if $T = T_{11} + T_{12}QT_{21}$

The parameterization of all two-degree-of-freedom stabilizing controllers is also available in [70].

4.3 Minimal Order Stabilization

This section finds the necessary and sufficient conditions for the existence of a stabilizing controller of a prescribed order (k). the main result is summarized in the following theorem [20].

Theorem 4.2: There exists a stabilizing output feedback law of order k iff there exist symmetric matrices R, S and a scalar $\gamma > 0$ such that

$$AR + RA^T < BB^T, \quad A^T S + SA < C^T C \tag{4.1}$$

and

$$\begin{bmatrix} \gamma R & I \\ I & \gamma S \end{bmatrix} \geq 0 \text{ with } Rank \begin{bmatrix} \gamma R & I \\ I & \gamma S \end{bmatrix} \leq n + k \tag{4.2}$$

Based on this theorem, by iteration on the parameter k we can find the minimum achievable order for stabilizing a plant. In [45] a method to find a lower and upper bound for this minimum is given. The conditions in (4.1) and (4.2) are convex for the case of full order controller, i.e. where $n = k$. In this case the rank constraints in (4.2) are trivially satisfied and the problem is reduced to an LMI convex feasibility problem. However, the low order controller case $k < n$ is no longer convex since the rank conditions are not convex in the parameter space (R, S) (see [33] for more details).

Remark 4.3: In [46] it is shown that there exist an open set in the set of n_{th} order SISO systems such that every system in this set is not stabilizable by a controller of order smaller than $n - 1$!

Remark 4.4: Suppose $G(s) = \frac{a(s)}{b(s)}$ be a transfer function (SISO) of order n and relative degree r and $a(s)$, $b(s)$ are coprime. The maximum required order for stabilization of

G is $n-1$ [57]. It is shown in [56] that the upper bounds on the number of right half plane poles and zeros and an upper bound on the relative degree do not allow these results to be improved (counter examples!). But if $a(s)$ is Hurwitz the maximum required order reduces to $r-1$. In [56] it is shown that the maximal controller order $r-1$ may be necessary even when an upper bound on the number of right half plane poles is known.

Another way to cope with minimal order stabilizer problem is directly analyzing the characteristic polynomial of the closed loop system. The following two terms are more common for this approaches: "output feedback stabilization" (normally for memory less feedback) and "fixed order feedback stabilization" (for higher orders). These methods are based on Hurwitz, Hermit Biehler theorem [31], Taraski theorem [4], Seidenberg theorem [4] and decision algebra[15]. Although these methods have been well developed by different scientist from control and mathematic society, their applicability because of high complexity is still questionable!

Strong stability and realizability

Definition 4.2: Consider the feedback system shown in Figure 4.3, if this system is stabilized by a stable controller K, then K is said to be a strongly stabilizable controller.

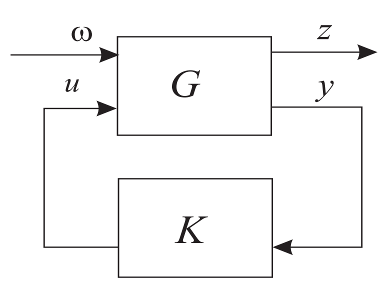

Figure 4.3: General feedback control system

Definition 4.3: Let the distinct real zeros of $G(s)$ be denoted by $z_1, z_2, \cdots, z_l$ (infinity included) and let the total number of the real poles of $G(s)$ to the right of z_i, each counted according to its McMillan degree, be denoted by a_i $(i = 1, 2, \cdots, l)$. Then the plant $G(s)$ is said to have the parity interlacing property if the integers $a_1, a_2, \cdots, a_l$ are either all even or all odd.

A SISO plant is said to satisfy the parity interlacing property if the number of unstable real poles between any two unstable real zeros is even; $+\infty$ counts as an unstable zero if the plant is strictly proper. The following theorem shows the necessary and sufficient condition for existing a strongly stabilizing controller.

Theorem 4.3: A plant is strongly stabilizable (can be stabilized by a stable controller) iff the plant satisfies the parity interlacing property [69].

There are several procedures for constructing strongly stabilizable controllers, see e.g. [61, 69] and [19] (on p. 70). Most of these methods involve constructing a stable transfer matrix, satisfying certain interpolation conditions, that usually results in large order controllers. It has bee shown that the order of strongly stabilizing controllers may have to be very large depending on the pole zero pattern of the plant [57]. Obviously, if a plant is strongly stabilizable the choice of $K(s)$ is not unique and in general, each possible closed loop configuration defines a different overall transfer function $T(s)$.

Definition 4.4: The transfer matrix $T(s)$ is called strongly realizable for $G(s)$ if the plant is strongly stabilizable with a closed loop configuration possessing the transfer function $T(s)$.

The following theorem shows the necessary and sufficient condition for strong realizability of a pair $(G(s), T(s))$.

Theorem 4.4: The transfer function $T(s) \in \mathcal{RH}_\infty$ is strongly realizable for $G(s)$ iff

- Every unstable zero of G (including infinity) is a zero of T of at least the same multiplicity.

- No unstable pole of G is a zero of T.
- The unstable real zeros of T (including infinity) and the unstable real poles of G possess the parity interlacing property.

4.4 Controller Reduction

It is clear that when we are dealing with low order controllers, there are fewer things to go wrong in the hardware and software; they are easier to understand and the computational requirements are less. Therefore simple controllers are normally preferred over complex ones. In order to achieve these advantages, there are lots of efforts to invent new methods which yield low order controllers in comparison to the typical ones. These methods are divided into two classes: direct and indirect. In direct methods the parameters defining the low order controllers are evaluated by some predefined procedure, while in indirect methods at first a high order controller is found and then a procedure (like weighted approximation) is used to simplify it. It should be noted that reducing the designed controller for a high order plant is more effective than designing a controller for the reduced plant, because:

- Error propagation might occur when the reduction is done before controller design.
- the plant approximation needs knowledge of the controller (which has not yet been designed) and therefore one is impaled on the horns of a dilemma! (comprehensive discussion in [50] on p. 95)

In practice exploiting the same methods as in the model reduction are not appropriate and it might cause instability or aggravate the performance drastically. Therefore the absolute error is not an effective indicator to judge the quality of controller reduction, but some other indicators and approaches should be presented. In this section well-known methods of controller reduction are concisely reviewed.

Extended Balanced Truncation

In [73] a controller reduction algorithm based on balancing and truncations is proposed which diagonalizes the (extended) closed loop ($T(s)$) Gramians without mixing the plant and controller states. Based on this approach the reduced order controller will be obtained by balancing and truncation.

Stochastic Balanced Truncation (Multiplicative and Relative Error)

In controller reduction, instead of a reduced order model with small absolute error sometimes it is more straight forward to look for one with small multiplicative (or relative) error. Assume $G_r(s)$ is a reduced form of $G(s)$, then the multiplicative and relative error are defined as follows:

$$\Delta_{mul} = \quad G_r^{-1}(G_r - G), \quad \Delta_{rel} = G^{-1}(G - G_r)$$

and they satisfy the following relationships:

$$\begin{aligned} G(s) = &\quad G_r(s)(I - \Delta_{mul}), \quad G_r(s) = G(s)(I - \Delta_{rel}) \\ \Delta_{mul} = &\quad (I - \Delta_{rel})^{-1}\Delta_{rel} \end{aligned}$$

Motivation: The reduced order controller is accurate enough (bandwidth ω_{BT}) provided that

$$\overline{\sigma}(\Delta_{mul}(j\omega)) < 1, \quad \omega < \omega_{BT}$$

A known method to find a reduced order model with small multiplicative (relative) error is stochastic balanced truncation.

For overview of this reduction approach see [50] (p. 63-80). The a priori error bound is given as follows:

$$\sigma_{k+1} \leq \|G_r^{-1}(G_r - G)\|_\infty \leq \prod_{r+1}^{n} \frac{1+\sigma_i}{1-\sigma_i} - 1$$

$$\sigma_{k+1} \leq \|G^{-1}(G - G_r)\|_\infty \leq \prod_{r+1}^{n} \frac{1+\sigma_i}{1-\sigma_i} - 1$$

Remark 4.5: Stochastic balanced truncation can be assumed as special case of weighted balancing represented in the next section. If G^{-1} is stable and proper the same results can be achieved by frequency weighted balancing and therefore the same error bound is valid there!

In general, assuming original controller to be stable, the stability of reduced controller is guaranteed but the stability of closed loop system is open to doubt. Applicability of this approach is restricted to stable square systems with no zeros on the extended imaginary axis but there are some extensions of this algorithm to non-square transfer function matrices.

Frequency Weighted Absolute Error

The controller reduction problem can also be regarded as minimizing the $\mathcal{H}_\infty$ norm of the absolute error (difference between the original controller and the reduced one) with a frequency dependent weight. The weight serves to say that it is more important to approximate the controller well at certain frequencies (where the weight is high) than at others. In order to elaborate this fact, we can focus on the scalar case, where we know that the closed loop transfer function has the greatest gain near the unity gain crossover frequency. Evidently, it is more important to approximate the controller accurately near the crossover frequency, an idea which is familiar from classical control reduction methods. Some other motivations for frequency weighted approximation are:

- *Maintaining stability in the closed loop system*: As depicted in Figure 4.4, using small gain theorem, results in one sided frequency weighted inequality as follows:

$$\|(K - K_r)\underbrace{G(I + KG)^{-1}}_{V}\|_\infty < 1$$

  and to achieve higher robustness, it converts to a minimization problem.

- *Similar closed loop behavior*: Closed loop approximation results in two sided

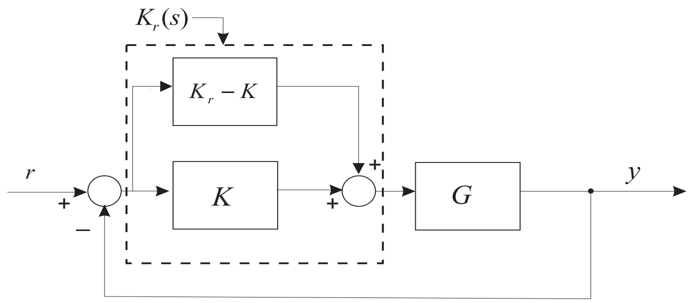

Figure 4.4: Rearrangement of feedback system with reduced order controller

frequency weighted norm as follows:

$$\|T(s) - T_r(s)\|_\infty \approx \| \underbrace{(I + GK)^{-1}G}_{W}(K - K_r) \underbrace{(I + GK)^{-1}}_{V} \|_\infty$$

and minimizing the above norm, guarantees a similar closed loop behavior.

- *Power spectrum*: The behavior of the controller is more important in the frequency that its input has the most energy. Thus, by considering $\Phi(j\omega) = VV^H$ where $\Phi(j\omega)$ denotes the power spectrum of the controller's input, reduction is defined as minimizing the following expression.

$$\|(K - K_r)V(s)\|_\infty$$

- *Suboptimal H_∞ controller*: Consider a class of (reduced-order) controllers, which can be represented in the following form

$$K_r = K + V^{-1}\Delta W^{-1}$$

where K may be interpreted as a nominal higher order $\mathcal{H}_\infty$ suboptimal controller, and Δ is a stable perturbation with stable minimum phase and invertible weighting functions V^{-1} and W^{-1}. It can be shown that a reduced order controller K_r in this class will preserve the stability and $\mathcal{H}_\infty$ performance bound, if

$$\|\Delta\|_\infty = \|W(K_r - K)V\|_\infty < 1$$

Thus controller reduction using suboptimal $\mathcal{H}_\infty$ design approach with objective consideration (stabilizing and satisfaction of gain constrains) boils down to a weighted frequency reduction (for more details see [74] (p. 309)).

- *Model reduction for control purpose*: If we reduce the plant before controller design, this controller (K) also stabilizes the original plant (G), if

$$\|\Delta G_r K(I + G_r K)^{-1}\|_\infty < 1$$

where $\Delta = (G - G_r)G_r^{-1}$. This can be derived easily from small gain theorem. This gives us some clues for multiplicative error reduction of a plant with rough information from the possible controller (cut-off frequency, slope and etc.).

$$\|\Delta G_r K(I + G_r K)^{-1}\|_\infty \approx \|(G - G_r)\underbrace{K(I + GK)^{-1}}_{V}\|_\infty < 1$$

It could be concluded that the plant reduction can be considered as a relative error problem together with a possible frequency weighting involving a 20 dB per decade roll off. For more details see [50] (p. 101-102).

Frequency Weighted Balancing

One way to solve $\min_{K_r} \|W(K_r - K)V\|_\infty$ is the idea of frequency weighted balanced truncation. In this approach we balance and then truncate those parts of the system (its state vector) which are clearly associated with K rather than weights (W, V). For instance in the following one sided case

$$KV = \left[\begin{array}{cc|c} A_k & B_k C_v & B_k D_v \\ 0 & A_v & B_v \\ \hline Cv & 0 & 0 \end{array}\right]$$

it is not hard to see that the top left block of the observability Gramian is equal to the observability Gramian of K, but the controllability Gramian is affected by the input weight. Roughly speaking, the view of state variables of K has not changed from the output, what has changed is the effect of input on these states. For balancing we now suppose a coordinate change (that only affects the states of K) such that the top left block of the (overall) controllability Gramian becomes diagonal and equal to the observability Gramian of K. Similar approach can be used for VK. In this case the stability is guaranteed[4] but no a priori error bound is known.

The two sided case is also similar and it balances the top left block of the observability and controllability Gramians by similarity transformation that only affects the states of K.

$$WKV = \left[\begin{array}{ccc|c} A_k & 0 & B_kC_v & B_kD_v \\ B_wC_k & A_w & 0 & 0 \\ 0 & 0 & A_v & B_v \\ \hline D_wC_k & C_w & 0 & 0 \end{array}\right]$$

In this case there is no guarantee for the stability. The alternative methods in [58] propose minor adjustments which guarantee the stability of reduced controller but not the closed loop system.

Remark 4.6: For frequency weighted reduction there is no neat a priori error bound and is only applicable to stable controllers.

Remark 4.7: Frequency weighted balancing without weight! In some cases the input and output weighting are not given, but the frequency range of interest is known. This problem can be attacked directly, without constructing input and output weights. This is achieved by using the frequency domain representation of Gramians as follows.

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

Thus if we are interested in the frequency interval $[0, \omega]$, we simultaneously diagonalize the two Gramians $P(\omega)$ and $Q(\omega)$ defined above and proceed as before. An error bound and more details can be found in [29].

[4] Similar to stability proof for balance and truncation, the Lyapunov equation is the key to stability.

Reduction of Coprime Factors

As it is mentioned before, one way to handle instability in reduction is to reduce the stable part and add the unstable part in the end. But copying the unstable part directly into the reduced stable part is not reliable (or optimal). Coprime factorization cope with this problem in model reduction, just by converting an unstable system to two stable transfer matrices. This idea can be expanded for controller reduction. Suppose the following coprime factorization of $K(s)$:

$$K(s) = X_k(s)Y_k^{-1}(s) \quad (M_kX_k + N_kY_k = I)$$

we find the reduced controller

$$K_r(s) = X_{kr}(s)Y_{kr}^{-1}(s) \quad (M_{kr}X_{kr} + N_{kr}Y_{kr} = I)$$

by minimizing Δx and Δy defined below

$$\Delta x = X_{kr} - X_k, \quad \Delta y = Y_{kr} - Y_k$$

One of the possible interpretation of this approach is for LQG controller reduction where X_k and Y_k are two meaningful (and distinguishable) transfer matrices in the closed loop as shown in Figure 4.5 (see [50] p. 138).

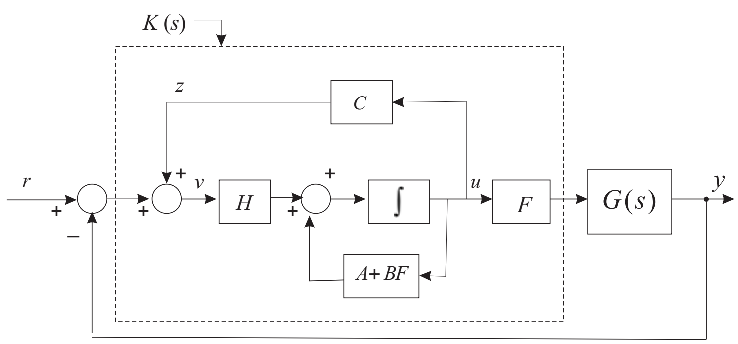

Figure 4.5: Controller-plant loop resulting from LQG design, coprime factors $X_k(s)$ and $Y_k(s)$ are the transfer matrices form ν to u and ν to z respectively, assuming the feedback z is disconnected.

The following theorem shows the effect of reduction based on coprime factors on the closed loop behavior.

Theorem 4.5: If $\| \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \|_\infty \leq \epsilon < 1$, then $K(s)$ stabilizes the plant and it implies the following bound on the closed loop error:

$$\frac{\|T(G, K_r) - T(G, K)\|_\infty}{\|T(G, K)\|_\infty} \leq \frac{\epsilon}{1 - \epsilon}$$

The idea of frequency weighting can be applied here as well, which implies that the reduced controller can be derived by minimizing the following norm [39]

$$\min_{X_{kr}, Y_{kr}} \|[\Delta y \ \Delta x] \begin{bmatrix} V \\ W \end{bmatrix} \|_\infty$$

where V and W are weighting matrices. The frequency weighted coprime factorization reduction is also an effective approach to preserve $\mathcal{H}_\infty$ performance by reduction in suboptimal case. Similarly the relative error approximation for controller reduction can be formulated with comprime factors as follows:

$$\begin{bmatrix} Y_{kr} & -M_{kr} \\ X_{kr} & N_{kr} \end{bmatrix} = (I - \Delta_{rel}) \begin{bmatrix} Y_k & -M_k \\ X_k & N_k \end{bmatrix}$$

$\mathcal{H}_\infty$ Controller Reduction

In $\mathcal{H}_\infty$ performance-preserving controller reduction, one way is additive reductions ($K_r = K + V^{-1}\Delta W^{-1}$) that was briefly discussed in motivation items of frequency weighted reduction. The procedure below, can be used to generate a required reduced-order controller that will preserve the closed loop $\mathcal{H}_\infty$ performance bound (γ).

1. Let K be a full order controller that fulfills the $\mathcal{H}_\infty$ performance bound (γ).

2. Compute weighting functions V^{-1} and W^{-1} using the algorithm in [26].

3. Use a weighted reduction method to find K_r so that

$$\|\Delta\|_\infty = \|W(K_r - K)V\|_\infty < 1$$

Another approach to preserve $\mathcal{H}_\infty$ performance by reduction in suboptimal case ($\gamma > \gamma_{min}$) is coprime factorization approach. This approach ends in a frequency weighted coprime factorization reduction for more details see [74] (p. 312).

Remark 4.8: The inequalities that the reduced controller are found in here are just the sufficient conditions to preserve stability and $\mathcal{H}_\infty$ performance. It means that there might be some other reduced controllers with the same properties that don't satisfy the norm inequalities of this approach.

In calculating the optimal $\mathcal{H}_\infty$ controller two Riccati equations have to be solved. It's easy to see that by balancing and truncating these equation, two reduced Riccati equation can be achieved that we can interpret them as the reduced plant with its optimal $\mathcal{H}_\infty$ controller. The key question is whether this reduced controller will function satisfactorily with the original plant, which is not guaranteed by the design procedure. A sufficient condition for closed loop stability is given in [49].

Part II

New Results

Chapter 5

PRESERVING STABILITY IN MODEL REDUCTION

5.1 Introduction

This chapter considers the problem of instability in similarity transformation based model reduction methods. The main results chiefly contribute to linear time-invariant (LTI) stable models, but using some known techniques the results are extended to LTI unstable cases.

Stability is an essential property that has to be preserved in the process of model reduction. Some approaches that focus on preserving special characteristics of the original transfer matrix might easily lose the stability of the reduced system. For instance in Krylov subspace methods for model reduction, the focus is on matching the coefficients of the Taylor (or McLaurin) series of the transfer matrix, but in general, there is no guarantee for the stability of the reduced model [5]. Some modification approaches are proposed in [34] and [27] to rectify this problem. Another example is balancing and truncation with two sided weighting (Enns' method [21]) that focus on approximating the original model in a specific range of frequency. Unfortunately it also suffers the lack of stability.

Our treatment of stability here follows that of [29, 60], where stability is guaranteed by generalized Gramians, which are solutions of two Lyapunov equations. This idea is generalized in section 5.2 by introducing the modified Lyapunov Criterion that later is used as the base of our stability arguments. The approach here, like in [37], utilizes the theoretic framework developed in [10] by exploiting strict Lyapunov inequalities.

These mathematical preliminaries serve our main result which parameterizes a set of reduced models that preserve the stability of the original stable model. This result is summarized in Proposition 5.1 of section 5.3. An interesting application of this parameterization is an algorithm to preserve stability in currently available reduction approaches that suffer from the lack of stability, by slightly modifying the corresponding similarity transformations that presented in section 5.6. This is demonstrated in two examples of a Butterworth filter and a four disk system in Chapter 8 to compare the results with those in [50] and [74]. The modification procedure presented here is applicable to any reduction method based on similarity transformation. It is shown that the efforts by [38, 60, 65] and [29] for two sided frequency weighting reduction method can be formulated as a special case of our approach.

5.2 *Modified Lyapunov Criterion*

From definition 2.2 a system is stable if all signals, containing all measurable and immeasurable signals in all components including even hidden modes, remain bounded provided that the all injected signals (at any possible location) and initial conditions are bounded. Considering a state space representation as follows:

$$G(s) : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] = C(sI - A)^{-1}B + D \tag{5.1}$$

using theorem 2.3, analyzing stability in this domain can be done by analyzing the eigenvalues of matrix A in representation (5.1). Equivalently from Lyapunov criterion (Corollary 2.1), matrix A is a stability matrix ($Re(\lambda_i(A)) < 0$) iff for any positive definite (symmetric) matrix Q, exists a positive definite (symmetric) matrix P such that $AP + PA^T = -Q$.

The guaranteed stability in the balancing approaches for order reduction (for an overview see [29]) is deduced from *Lyapunov Criterion* where $Q = BB^T$ (and/or $Q = C^TC$) as shown in the following Corollaries:

Corollary 5.1: If $AP + PA^T + BB^T = 0$ where P is a positive definite (symmetric) matrix and the pair (A, B) controllable, then A is a stability matrix[52].

If the pair (A, B) is not controllable, it follows that the eigenvalues of A lie in the closed left half plane and each imaginary eigenvalue corresponds to a non-controllable mode.

Corollary 5.2: If $A^T P + PA + C^T C = 0$, where P is a positive definite (symmetric) matrix, and the pair (A, C) observable, then A is a stability matrix.

If the pair (A, C) is not observable, A has some pure imaginary eigenvalues, which their associated eigenvectors lie in the null-space of C, i.e., they correspond to some non-observable modes.

In this section we propose a modified version of Lyapunov criterion in the following theorem, which represents the stability of a system in form of a linear matrix inequality (LMI). This theorem will be used in the proceeding lemmas to state the stability of reduced models derived by truncation or singular perturbation presented in section 3.2.

Theorem 5.1: **Modified Lyapunov Criterion**: A is a stability matrix iff there exists a positive definite (symmetric) matrix P, such that $AP + PA^T < 0$.

Proof: The necessary condition is direct result of Corollary 2.1, by choosing a $Q > 0$. For the sufficient condition, let v_i be the left eigenvector of A and $\lambda_i(A)$ the corresponding eigenvalue ($v_i^* A = \lambda_i v_i^*$). Then we multiply the given matrix inequality from left with v_i^* and from right with v_i, consequently:

$$\begin{aligned}
AP + PA^T < 0 \quad &\Rightarrow \quad v_i^* APv_i + v_i^* PA^T v_i < 0 \\
&\Rightarrow \quad \lambda_i(A)v_i^* Pv_i + v_i^* P\lambda_i^*(A)v_i < 0 \\
&\Rightarrow \quad (\lambda_i(A) + \lambda_i^*(A))v_i^* Pv_i < 0 \\
(v_i^* Pv_i > 0) \quad &\Rightarrow \quad (\lambda_i(A) + \lambda_i^*(A)) < 0 \\
&\Rightarrow \quad Re(\lambda_i(A)) < 0
\end{aligned}$$

Thus the real part of every eigenvalue of A is in the left half plane, therefore A is a stability matrix. ■

In model reduction methods, which are based on the truncation of a special representation of a model, the stability of the reduced system depends on a sub-matrix of A. Without loss of generality we can assume the upper left block of A (A_{11}) as the reduced system's main matrix. The following Lemma, based on *Modified Lyapunov Criterion*, presents a sufficient condition for A_{11} to be a stability matrix.

Lemma 5.1: Let $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$ be a stability matrix. A_{11} is also a stability matrix, if there exists a symmetric matrix $P = diag(P_1, P_2) > 0$ (size(A_{11})=size(P_1)), which satisfies $AP + PA^T < 0$.

Proof:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix}+\begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix}\begin{bmatrix} A_{11}^T & A_{21}^T \\ A_{12}^T & A_{22}^T \end{bmatrix}< 0$$

$$\Rightarrow \begin{bmatrix} A_{11}P_1 + P_1A_{11}^T & * \\ * & * \end{bmatrix} < 0$$

$$\Rightarrow A_{11}P_1 + P_1A_{11}^T < 0$$

■

As it was mentioned, an application of Lemma 5.1 is the stability of reduced model. We formulate this result in the following Lemma:

Lemma 5.2: Let the stable transfer matrix $G(s)$ with a state space realization as follows:

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} A_{11} & A_{12} & B_1 \\ A_{21} & A_{22} & B_2 \\ \hline C_1 & C_2 & D \end{array}\right] \tag{5.2}$$

be such that there exists a block diagonal matrix

$$P = diag(P_1, P_2) > 0, \quad size(A_{11}) = size(P_1)$$

satisifying

$$AP + PA^T < 0$$

then the reduced system derived by truncating $G(s)$,

$$G_r(s) \quad = \quad \left[\begin{array}{c|c} A_{11} & B_1 \\ \hline C_1 & D \end{array}\right], \tag{5.3}$$

is stable.

Corollary 5.3: If $G(s)$ satisfies the conditions in Lemma 5.2, then the reduced system derived by singular perturbation,

$$G_{re}(s) \quad = \quad \left[\begin{array}{c|c} A_{11} - A_{12}A_{22}^{-1}A_{21} & B_1 - A_{12}A_{22}^{-1}B_2 \\ \hline C_1 - C_2A_{22}^{-1}A_{21} & D - C_2A_{22}^{-1}B_2 \end{array}\right],$$

is also stable.

Proof: From Lemma 5.1 we assume

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix} + \begin{bmatrix} P_1 & 0 \\ 0 & P_2 \end{bmatrix}\begin{bmatrix} A_{11}^T & A_{21}^T \\ A_{12}^T & A_{22}^T \end{bmatrix} < 0$$

After premultiplying and postmultiplying this Lyapunov inequality by T and T^T, where

$$T = [I, \ -A_{12}A_{22}^{-1}]$$

we obtain

$$(A_{11} - A_{12}A_{22}^{-1})P_1 + P_1(A_{11} - A_{12}A_{22}^{-1})^T < 0$$

Thus, based on Theorem 5.1 $A_{11} - A_{12}A_{22}^{-1}$ is a stability matrix and consequently G_{re} is stable. ■

In the next step, we are looking for a set of similarity transformations that all reduced models of order k, derived by truncating the transformed models, preserve stability. Therefore, it is necessary to contemplate the effect of similarity transformation on the modified Lyapunov criterion. The following Lemma shows how a similarity transformation matrix reshapes the modified Lyapunov criterion.

Lemma 5.3: If $AP + PA^T < 0$ $(P > 0)$, and we apply the similarity transformation T to A, then $\hat{A}$ $(= TAT^{-1})$ satisfies $\hat{A}\hat{P} + \hat{P}\hat{A}^T < 0$, where $\hat{P} = TPT^T$.

Lemma 5.3: *Proof:*

$$\begin{aligned} \hat{A}\hat{P} + \hat{P}\hat{A}^T &= TAT^{-1}(TPT^T) + (TPT^T)T^{-T}AT^T \\ &= TAPT^T + TPAT^T \\ &= T(AP + PA)T^T < 0 \end{aligned}$$

■

A reasonable question rises from Lemma 5.2 is how a state space representation which satisfies the conditions mentioned there can be found. Proposition 5.1 answers this question by presenting a set of similarity transformations (T) that shapes the matrix P as it is required in Lemma 5.2. It also parameterizes all stable reduced systems (subject to a specific P) of order k in the state space, derived by truncating the transformed system.

5.3 Parameterizing Stable Reduced Models

Consider the system given by (5.1), where the system matrices are decomposed as in (5.2). Using these partitioned matrices, a reduced order model can be obtained as in (5.3). The order of this model is k and the quality of approximation depends on the state coordinate basis of the original system and its stability is verified by the eigenvalues of A_{11}. In order to improve the quality of approximation and/or the stability of reduced model, other realization of $G(s)$ can be used. In other words stability and quality of reduction based on truncation is realization dependent. To derive different realizations of $G(s)$ similarity transformations are powerful means. After applying a similarity transformation T to a state space realization of $G(s)$, yielding

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{c|c} TAT^{-1} & TB \\ \hline CT^{-1} & D \end{array}\right] = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array}\right] = \left[\begin{array}{cc|c} \hat{A}_{11} & \hat{A}_{12} & \hat{B}_1 \\ \hat{A}_{21} & \hat{A}_{22} & \hat{B}_2 \\ \hline \hat{C}_1 & \hat{C}_2 & \hat{D} \end{array}\right]$$

another reduced model of order k can be derived as follows:

$$G_r(s) = \left[\begin{array}{c|c} \hat{A}_{11} & \hat{B}_1 \\ \hline \hat{C}_1 & \hat{D} \end{array}\right]. \tag{5.4}$$

Therefore different similarity transformations yield unlike reduced models with distinct properties. In this part we present a large set of similarity transformations and accordingly a set of coordinate basis to secure the stability of the reduced model. The choice of similarity transformation is then recast as a choice of matrices L and R such that

$$\hat{A}_{11} = LAR, \quad \hat{B}_1 = LB, \quad \hat{C}_1 = CR, \quad LR = I.$$

In this way we parameterize a large set of stable reduced models of order k. In fact L and R can be identified by the first k rows of T (similarity transformation) and by the first k columns of its inverse (T^{-1}) respectively. The following proposition summarizes our result by parameterizing all block diagonalizing similarity transformations and all the corresponding set of reduced (stable) systems.

Proposition 5.1: Let the transfer matrix $G(s)$ in (5.1) be such that there exists a positive definite symmetric matrix P that satisfies

$$AP + PA^T < 0 \tag{5.5}$$

and $U = chol(P)$, then the set of all block diagonalizing similarity transformations can be described as

$$T = \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix} U^{-T} \tag{5.6}$$

for any *arbitrary* pair of full row-rank matrices $X \in \mathbb{R}^{k\times n}$ and $Y \in \mathbb{R}^{(n-k)\times(n-k)}$ $(1 \leq k \leq n)$, and the corresponding set of reduced (stable) systems of order k can be parameterized as

$$G_r(s) = \left[\begin{array}{c|c} LAR & LB \\ \hline CR & D \end{array}\right], \tag{5.7}$$

where

$$L = XU^{-T}, \quad R = U^T \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix}^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix}. \tag{5.8}$$

Proof: The proof of this proposition relies heavily on the preceding lemmas. To start, we show that the matrix T in (5.6) is a block diagonalizing similarity transformation. Since T is a similarity transformation we have:

$$\begin{aligned} G(s) &= \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{c|c} TAT^{-1} & TB \\ \hline CT^{-1} & D \end{array}\right] \\ &= \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array}\right] = \left[\begin{array}{cc|c} \hat{A}_{11} & \hat{A}_{12} & \hat{B}_1 \\ \hat{A}_{21} & \hat{A}_{22} & \hat{B}_2 \\ \hline \hat{C}_1 & \hat{C}_2 & \hat{D} \end{array}\right] \end{aligned} \tag{5.9}$$

From (5.5) and (5.6), Lemma 5.3 shows that $\hat{A}$ satisfies

$$\hat{A}\hat{P} + \hat{P}\hat{A}^T < 0 \tag{5.10}$$

where

$$\begin{aligned} \hat{P} = TPT^T &= \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix} U^{-T} P \left(\begin{bmatrix} X \\ YX^{\perp} \end{bmatrix} U^{-T} \right)^T \\ &= \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix} U^{-T} U^T U U^{-1} \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix}^T \\ &= \begin{bmatrix} XX^T & XX^{\perp T} Y^T \\ YX^{\perp} X^T & YX^{\perp} X^{\perp T} Y^T \end{bmatrix} = \begin{bmatrix} XX^T & 0 \\ 0 & YX^{\perp} X^{\perp T} Y^T \end{bmatrix} \\ &= diag(\hat{P}_1, \hat{P}_2) > 0, \quad size(\hat{A}_{11}) = size(\hat{P}_1) \end{aligned} \tag{5.11}$$

Thus, T is a block diagonalizing similarity transformation. From (5.9), (5.10) and (5.11), Lemma 5.2 shows

$$G_r(s) = \left[\begin{array}{c|c} \hat{A}_{11} & \hat{B}_1 \\ \hline \hat{C}_1 & \hat{D} \end{array}\right] = \left[\begin{array}{c|c} LAR & LB \\ \hline CR & D \end{array}\right],$$

where L and R are given in (5.8), is stable.
To complete the proof, it suffices to show that if $\hat{P} = diag(\hat{P}_1, \hat{P}_2) > 0$, there exists a similarity transformation with the structure shown in (5.6), which satisfies $\hat{P} = TPT^T$. Choosing

$$X = [chol(\hat{P}_1) \quad 0_{n-k\times k}],\ X^{\perp} = [0_{k\times n-k} \quad I_k],\ Y = chol(\hat{P}_2)$$

and substituting in (5.6), gives the desired result. ■

It should be noted that if A is stable, on one hand there are infinity number of P matrices that satisfy the LMI in (5.5) and on the other hand by varying the arbitrary matrices X and Y in (5.6), a *large set* of stable reduced models can be achieved.

Remark 5.1: (*singular perturbation*) By exploiting Corollary 5.3 The results in Proposition 5.1 can be extended to the reduced systems derived by singular perturbation approximation. It means that if we apply the similarity transformation T in (5.6) to the original model, independent of using truncation or singular perturbation the reduced model of order k is stable. Therefore in addition to (5.7) and (5.8), another class of reduced models of order k with guaranteed stability and DC gain can be parameterized. But due to the complexity of their structure, the parameterization of singular perturbed reduced models have not been presented.

Remark 5.2: (*Unstable models*) Since our approach demands the stability of the original model, one might ask what happens if the model is originally unstable. One well-known idea is to compute an additive stable-unstable spectral decomposition of G as $G = G_s + G_u$, where G_s of order n_s contains the stable poles of G and G_u of order $n - n_s$ contains the unstable poles of G as depicted in Fig. 5.1. Then reduce the stable part of the model and copy the unstable part into the reduced model, which means $G_r = G_{sr} + G_u$ as shown in Fig. 5.2. Consequently G_r in Fig. 5.2 is the reduced model of G. Another method is to represent the model transfer matrix as ND^{-1} where N and D are both stable rational transfer matrices. Then we can approximate both the numerator N and the denominator D and construct the reduced model $G_r = N_r D_r^{-1}$. But there is an infinite number of fractions so the question arises which fraction should be chosen. In [39, 44] a normalized representation so-called coprime factorization has been used, but there is no proof on the superiority of this selection.

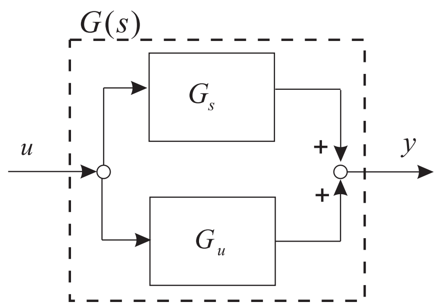

Figure 5.1: Stable-Unstable decomposition

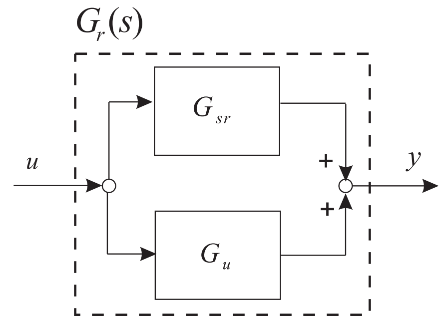

Figure 5.2: Reduced unstable model

5.4 Numerical issues

Regarding Theorem 5.1, existing a positive definite matrix P, which satisfies (5.5) in Proposition 5.1 only requires the stability of G and does not add any other constraints on it. For a stable system the inequality in (5.5) is a standard LMI, which is always solvable, but the solution is not unique. $P = \int_0^\infty e^{A\tau} Q e^{A^T\tau} d\tau$ is an explicit solution to (5.5), where Q is an arbitrary positive definite matrix. This inequality can also be solved using some well-known convex programming methods, see e.g. [14] and references therein. Another point, which is worth to be noted here, is the convexity of the set of solutions to (5.5). Assume that P_1 and P_2 are solutions to (5.5), which means

$$AP_1 + P_1A^T < 0, \quad AP_2 + P_2A^T < 0.$$

Then $c_1P_1 + c_2P_2$, for any positive real scalars c_1 and c_2 satisfies

$$A(c_1P_1 + c_2P_2) + (c_1P_1 + c_2P_2)A^T = c_1 \underbrace{(AP_1 + P_1A^T)}_{<0} + c_2 \underbrace{(AP_2 + P_2A^T)}_{<0} < 0$$

and therefore it is a solution to (5.5) as well and implies that the set of solutions to (5.5) is convex.

Obviously, the LMI in (5.5) imposes weaker restriction on P in comparison to the Lyapunov equation ($AP+PA^T = -Q$) where matrix $Q > 0$ is given, but a comparison study of the numerical effort is still required.

5.5 Dual Results

An interesting feature of the arguments in the previous sections is the validity of their dual forms. In this section the dual forms are presented concisely.

Theorem 5.2: (Dual of Lyapunov Criterion) A is a stability matrix iff there exists a positive definite symmetric matrix Q, such that $A^TQ + QA < 0$.

Lemma 5.4: (Dual of Lemma 5.2) Let the stable transfer matrix $G(s)$ have a state space realization as in (5.2). $G_r(s)$ in (5.3) is stable if there exists a block diagonal matrix $Q = diag(Q_1, Q_2) > 0$ satisfying $A^TQ + QA < 0$.

Lemma 5.5: (Dual of Lemma 5.3) If $A^TQ + QA < 0$ ($Q > 0$), and we apply the similarity transformation T to A, then $\hat{A}$ is equal to TAT^{-1} and satisfies $\hat{A}^T\hat{Q}+\hat{Q}\hat{A} < 0$, where $\hat{Q} = T^{-T}QT^{-1}$.

Proposition 5.2: (Dual of Proposition 5.1) If there exists a positive definite symmetric matrix Q which satisfies $A^TQ + QA < 0$ then for any *arbitrary* full row-rank matrices $X \in \mathbb{R}^{k\times n}$ and $Y \in \mathbb{R}^{k\times k}$, the reduced system $G_r(s)$ with realization (L_qAR_q, L_qB, CR_q, D) of order k is stable, where

$$U_q = chol(Q), \quad L_q = XU_q, \quad R_q = U_q^{-1} \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix}^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix}.$$

5.6 Application Outlook

In this section two algorithms illustrate how we can benefit from the parameterization of stable reduced models presented in (5.7). In the first algorithm we show how one of the famous model reduction methods, known as Krylov subspace methods, can be modified in order to guarantee the stability of reduced model. In the second one, we procure a solution to the problem of stabilizing the model resulted from a double-sided frequency weighting reduction method. The effectiveness of the proposed techniques are demonstrated by way of two numerical examples in Chapter 8. It should be noted that the application range of the proposed theorems is not confined to these two algorithms.

5.6.1 Preserving stability in Krylov subspace based model reduction methods

The output Krylov subspace of order k corresponding to the state space realization in (5.1) is defined as follows:

$$\mathcal{K}_k(A^{-T}, A^{-T}C) \triangleq colspan[A^{-T}C, \cdots, (A^{-T})^{k-1}A^{-T}C]$$

The output Krylov subspace can be used to find a projection, resulting in a reduced model that approximates the original transfer matrix in terms of matching the first k coefficients of its Taylor series (so-called moments). These applications of Krylov subspaces, which were first noted in [64], have been well developed and numerically enhanced during the last decade, for a comprehensive overview see [28] and [5]. It is proved that in (5.7) if the columns of L^T span $\mathcal{K}_k(A^{-T}, A^{-T}C)$, then the first k moments of the original and the reduced model match. In general, there is no guarantee for the stability of the reduced model, but if L^T is orthonormalized, which is always the case in the Arnoldi algorithm [5], and R is chosen equal to L^T then the reduced model preserves passivity. When L is given, the following Corollary shows how we can construct R based on Proposition 5.1 to preserve stability, while still the moments are matched. This Corollary also generalizes the result presented in [63].

Please note that the result is not restricted to passive models and only the stability of the original model is required.

Corollary 5.4: Given a stable system as in (5.1), for any $P > 0$ that satisfies $AP + PA^T < 0$ and any arbitrary full row rank matrix $\hat{L} \in \mathbb{R}^{k\times n}$, the reduced model $G_r(s) = (\hat{L}A\hat{R}, \hat{L}B, C\hat{R}, D)$ is stable if $\hat{R} = P\hat{L}^T(\hat{L}P\hat{L}^T)^{-1}$.

Proof: We show that $\hat{L}$ and $\hat{R}$ satisfy

$$\hat{L} = XU^{-T}, \quad \hat{R} = U^T \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix}^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix}$$

for special matrices X and Y. Assuming $U = chol(P)$ and substituting $\hat{L}U^T$ for X and I_{n-k} for Y in (5.8), we have

$$\hat{L}U^TU^{-T} = \hat{L}, \quad U^T \begin{bmatrix} \hat{L}U^T \\ I_{n-k}(\hat{L}U^T)^{\perp} \end{bmatrix}^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix} = PL^T(LPL^T)^{-1} = \hat{R}$$

Since $\hat{L}$ and $\hat{R}$ have the same structure as in (5.7), Proposition 5.1 shows that $G_r(s) = (\hat{L}A\hat{R}, \hat{L}B, C\hat{R}, D)$ is stable. ■

In other words if the columns of $\hat{L}^T$ are chosen form $\mathcal{K}_k(A^{-T}, A^{-T}C)$ and $\hat{R} = P\hat{L}^T(\hat{L}P\hat{L}^T)^{-1}$, the reduced model will be stable and k moments are matched. This application can be extended to input Krylov subspaces using the following Corollary.

Corollary 5.5: Given a stable system as in (5.1), for any $Q > 0$ that satisfies $A^TQ + QA < 0$ and any arbitrary full column rank matrix $\hat{R} \in \mathbb{R}^{n\times k}$, the reduced model $G_r(s) = (\hat{L}A\hat{R}, \hat{L}B, C\hat{R}, D)$ is stable if $\hat{L} = (\hat{R}^TQ\hat{R})^{-1}\hat{R}^TQ$.

Proof: Similar to Corollary 5.4. ■

Therefore if the columns of $\hat{R}$ are chosen form the input Krylov subspace and $\hat{L} = (\hat{R}^TQ\hat{R})^{-1}\hat{R}^TQ$, then the reduced model is stable and k moments are matched.

It is worth to be noted that, besides stability consideration, by adding more constraints on the inequality in (5.5) the properties of reduced model can be manipulated. The following corollary shows for instance how we can modify the conditions in Corollary 5.4 to control the distance of the poles of the reduced model from the imaginary axis. To be more precise, using the following corollary we can keep all the eigenvalues of the reduced model in the left-hand side of the eigenvalues of the original model[1].

Corollary 5.6: Given a system as in (5.1) which A satisfies $Re\{\lambda_i(A)\} < -\sigma \leq 0$. Then for any arbitrary full row rank matrix $\hat{L} \in \mathbb{R}^{k\times n}$ and $\hat{R} = P\hat{L}^T(\hat{L}P\hat{L}^T)^{-1}$, where $P > 0$ and satisfies $AP + PA^T + 2\sigma P < 0$, the reduced model $G_r(s) = (\hat{L}A\hat{R}, \hat{L}B, C\hat{R}, D)$ is stable and $A_r = \hat{L}A\hat{R}$ satisfies $Re\{\lambda_i(A_r)\} < -\sigma$.

Proof: Since $Re\{\lambda_i(A)\} < -\sigma \leq 0$, $(A + \sigma I)$ is a stability matrix therefore

$$(A + \sigma I)P + P(A + \sigma I)^T < 0$$

that is equivalent to

$$AP + PA^T + 2\sigma P < 0$$

Pre- and post-multiplying this inequality with $\hat{L}$ and $\hat{L}^T$ respectively, we have

$$\hat{L}AP\hat{L}^T + \hat{L}PA^T\hat{L}^T + 2\sigma\hat{L}P\hat{L}^T < 0$$

Noting that $(\hat{L}P\hat{L}^T)^{-1}(\hat{L}P\hat{L}^T)$ and $(\hat{L}P\hat{L}^T)^T(\hat{L}P\hat{L}^T)^{-T}$ are identity terms, this inequality can be modified as follows

$$\hat{L}AP\hat{L}^T \underbrace{(\hat{L}P\hat{L}^T)^{-1}(\hat{L}P\hat{L}^T)}_{I} + \underbrace{(\hat{L}P\hat{L}^T)^T(\hat{L}P\hat{L}^T)^{-T}}_{I} \hat{L}PA^T\hat{L}^T + 2\sigma \underbrace{\hat{L}P\hat{L}^T}_{P_r} < 0$$

By substituting A_r for $\hat{L}A\hat{R} = \hat{L}AP\hat{L}^T(\hat{L}P\hat{L}^T)^{-1}$ and P_r for $\hat{L}P\hat{L}^T$, we obtain

$$A_rP_r + P_rA_r^T + 2\sigma P_r < 0$$

or equivalently

$$(A_r + \sigma I)P_r + P_r(A_r + \sigma I)^T < 0$$

[1]The author would like to acknowledge the advice and criticism of Prof. P.C. Müller and Prof. A.C. Antoulas at GMA workshop (Modeling, identification and simulation in automatic control) at Bostalsee (Germany) that led to this result.

Therefore $A_r + \sigma I$ is a stability matrix that means $Re\{\lambda_i(A_r)\} < -\sigma \leq 0$ which is the desired result. ■

Remark 5.3: The result here is not restricted to stabilizing model reduction by input and output Krylov subspace method, but it can be used to associate with any model reduction methods that only specify L (or R). The result can be generalized to the case, where the whole transformation matrix is given. Consider $\breve{T}$ is a transformation matrix from a (truncation based) reduction method that does not guarantee the stability of the reduced model. Then the reduced model can be stabilized by projecting $\breve{T}$ onto the space of stabilizing transformations introduced in Proposition 5.1. This is equivalent to finding the matrices X and Y by solving the following optimization problem,

$$\min_{X,Y} \left\| \breve{T} - \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix} U^{-T} \right\|,$$

where $U = chol(P)$. This approach is further illustrated in the next subsection.

5.6.2 Preserving stability in frequency weighted reduction methods

The Enns' two sided frequency weighted reduction was briefly introduced in section 4.4. The goal of this approach is minimizing the index in (5.12), where $G(s)$ is the original model with the realization shown in (5.1), $G_r(s)$ is the reduced model and $W_1(s)$ and $W_2(s)$ are the frequency weights.

$$\min_{G_r(s)} \|W_1(s)(G(s) - G_r(s))W_2(s)\|_\infty \tag{5.12}$$

In some applications we look for a reduced model that approximates its original high order model in a specific range of frequency. In this case by choosing proper weighting matrices, Enns' method can be exploited. Enns proposed to calculate the transformation matrix $\breve{T} \in \mathbb{R}^{n\times n}$ that identifies the new coordinate basis by balancing the upper left $n \times n$ block of the controllability and observability Gramians of $G(s)W_2(s)$ and $W_1(s)G(s)$ respectively. Let $W_1(s) = (A_1, B_1, C_1, D_1)$ and $W_2(s) = (A_2, B_2, C_2, D_2)$

be minimal state space realizations of the weighting matrices. Consider the following realizations of $G(s)W_2(s)$ and $W_1(s)G(s)$

$$G(s)W_2(s) = \left[\begin{array}{cc|c} A & BC_2 & BD_2 \\ 0 & A_2 & B_2 \\ \hline C & DC_2 & DD_2 \end{array}\right], W_1(s)G(s) = \left[\begin{array}{cc|c} A_1 & B_1C & B_1D \\ 0 & A & B \\ \hline C_1 & D_1C & D_1D \end{array}\right]$$

where

$$P_{GW_2} = \begin{bmatrix} P_{11} & P_{12} \\ P_{12}^T & P_{22} \end{bmatrix}, Q_{W_1G} = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix}, \tag{5.13}$$

are the controllability Gramian of $G(s)W_2(s)$ and the observability Gramian of $W_1(s)G(s)$ respectively. Then the reduced model is obtained by truncating the new realization (after performing $\breve{T}$ on $G(s)$) as in (5.4). Although this method has been successfully employed in many applications, its main weakness is the lack of guaranteed stability of the reduced model. In this part we briefly present two previous works on Enns' approach in order to guarantee stability, then we suggest a third alternative algorithm for preserving stability by slight modification of Enns' proposed coordinate transformation.

- *modification by Lin and Chiu:* A modification of Enns' approach has been proposed by Lin and Chiu in [38], which under certain assumptions, guarantees the stability of the reduced model. Provided P_{22} and Q_{11} in (5.13) are nonsingular (condition ensured if $W_1(s)$ and $W_2(s)$ are minimal realizations), Lin and Chiu proposed to choose the frequency-weighted Gramians as

  $$P_L = P_{11} - P_{12}P_{22}^{-1}P_{12}^T, \quad Q_L = Q_{22} - Q_{12}^T Q_{11}^{-1} Q_{12}.$$

  It is shown that P_L and Q_L are the Gramians of $\hat{G}(s) = (A, \hat{B}, \hat{C}, D)$, where

  $$\hat{B} = BD_2 - P_{12}P_{22}^{-1}B_2, \quad \hat{C} = D_1C - C_1Q_{11}^{-1}Q_{12}.$$

  By balancing $\hat{G}(s)$ we obtain a coordinate transformation ($\hat{T}$) that makes the Gramians equal and diagonal. The balancing and truncation approach implies that the upper $k \times k$ block of $\hat{T}A\hat{T}^{-1}$, denoted by $\hat{A}_{11}$, is a stability matrix. Therefore if we apply $\hat{T}$ to $G(s)$ and noting that matrix A remains unchanged, the stability of the reduced model is automatically guaranteed.

- *modification by Wang et al.:* Another modification of Enns' method has been proposed by Wang et al. in [65]. In this method it is proposed to choose the frequency-weighted Gramians as P_W and Q_W, where they are the Gramians of $\tilde{G}(s) = (A, \tilde{B}, \tilde{C}, D)$, where $\tilde{B}$ and $\tilde{C}$ are computed from the orthogonal eigendecompositions of the symmetric matrices $X = -(AP_{11} + P_{11}A^T)$ and $Y = -(A^T Q_{22} + Q_{22}A)$ as follows:

$$\tilde{B} = U\sqrt{|\Sigma_1|},\ \tilde{C} = \sqrt{|\Sigma_2|}V^T$$

  where $X = U\Sigma_1 U^T$, $Y = V\Sigma_2 V^T$ and Σ_1 and Σ_2 are diagonal matrices. Then the transformation matrix $\tilde{T}$ is calculated by balancing $\tilde{G}(s)$. The balancing and truncation approach implies that the upper $k \times k$ block of $\tilde{T}A\tilde{T}^{-1}$, denoted by $\hat{A}_{11}$, is a stability matrix. Similar to the argument in Lin and Chiu approach, if we apply $\tilde{T}$ to $G(s)$ and noting that the main matrix of $G(s)$ and $\tilde{G}(s)$ are equal (A), the stability of the reduced model is automatically guaranteed.

- *An alternative based on Propositon 5.1:* Using similar notation as in section 5.3, a reduced model can be stabilized by the modification discussed in Remark 5.3. The construction of the projected transformation can be summarized as follows:

  1. find a $P > 0$, such that it satisfies $AP + PA^T < 0$
  2. calculate $U = chol(P)$
  3. calculate the modified transformation by projecting $\breve{T}$ onto the space of matrices defined in (5.6). A simple suboptimal solution is $X = \breve{T}_u U^T$, $Y = \breve{T}_l U^T X^{\perp\dagger}$, where $\breve{T}_u$ denotes the upper rows of $\breve{T}$ compatible to the size of X and $\breve{T}_l$ denotes the lower rows of $\breve{T}$ compatible to the size of $X^{\perp}$. This solution can be achieved by solving $\min_X \|\breve{T}_u - XU^{-T}\|$ and $\min_Y \|\breve{T}_l - YX^{\perp}U^{-T}\|$ sequentially that gives

$$T = \begin{bmatrix} \breve{T}_u U^T \\ \breve{T}_l U^T (\breve{T}_u U^T)^{\perp\dagger} (\breve{T}_u U^T)^{\perp} \end{bmatrix} U^{-T}. \tag{5.14}$$

In this part we elaborate the relationship between different alternatives described in this section.

- In the approach of Lin and Chiu $\hat{T}$ guarantees stability by diagonalizing a specific $P_L > 0$ that satisfies $AP_L + P_L A^T = -\hat{B}\hat{B}^T \leq 0$.

- In Wang's approach $\tilde{T}$ guarantees stability by diagonalizing a specific $P_W > 0$ that satisfies $AP_W + P_W A^T = -\tilde{B}\tilde{B}^T \leq 0$.

- In our approach T guarantees stability by block diagonalizing an arbitrary $P > 0$ that satisfies $AP + PA^T < 0$.

Based on the following lemma it can be seen that all of the mentioned approaches can be formulated as a special marginal case of our algorithm.

Lemma 5.6: If $AP_1 + P_1 A^T \leq 0$, for all $\epsilon > 0$ there exists a $P_2 > 0$ such that $AP_2 + P_2 A^T < 0$ and $\|P_1 - P_2\| < \epsilon$.

Proof: Assuming $AP_1 + P_1 A^T = -Q_1$, where Q_1 is a symmetric positive semi-definite matrix and using the idea in [51], this equation can be written as

$$(A \otimes I_n + I_n \otimes A^T)vec(P_1) = -vec(Q_1), \tag{5.15}$$

Without loss of generality we assume the following eigenvlaue decomposition of Q_1:

$$Q_1 = V diag(\lambda_1, \cdots, \lambda_s, 0, \cdots, 0)V^T \geq 0$$

where s is the number of nonzero eigenvalues of Q_1. If we define Q_2 for positive real numbers $\sigma_1, \cdots, \sigma_{n-s+1}$ as

$$Q_2 = V diag(\lambda_1, \cdots, \lambda_s, \sigma_1, \cdots, \sigma_{n-s+1})V^T > 0$$

then Q_2 is a symmetric positive definite matrix and therefore

$$Q_1 - Q_2 = \sum_{i=1}^{n-s+1} \sigma_i V_i V_i^T \tag{5.16}$$

where V_i is the eigenvector corresponding to σ_i. Considering P_2 satisfies $AP_2 + P_2 A^T = -Q_2 < 0$, we have

$$(A \otimes I_n + I_n \otimes A^T)vec(P_2) = -vec(Q_2), \tag{5.17}$$

Subtracting (5.17) from (5.15), we obtain

$$(A \otimes I_n + I_n \otimes A^T)(vec(P_1) - vec(P_2)) = -(vec(Q_1) - vec(Q_2)).$$

Noting that $vec(P_1) - vec(P_2) = vec(P_1 - P_2)$, we have

$$vec(P_1 - P_2) = -(A \otimes I_n + I_n \otimes A^T)^{-1} vec(Q_1 - Q_2).$$

For any matrix norm we know that $\|Av\| \leq \|A\| \, \|v\|$, thus

$$\|vec(P_1 - P_2)\| \leq \| - (A \otimes I_n + I_n \otimes A^T)^{-1}\| \, \|vec(Q_1 - Q_2)\|. \tag{5.18}$$

By using (5.16) and triangle inequality we have

$$\|vec(Q_1 - Q_2)\| = \|vec(\sum_{i=1}^{n-s+1} \sigma_i V_i V_i^T)\| \leq \sum_{i=1}^{n-s+1} \sigma_i \|vec(V_i V_i^T)\|$$

which means the upper bound of $\|vec(P_1 - P_2)\|$ in (5.18) and consequently $\|P_1 - P_2\|$ can be assigned to any $\epsilon > 0$ by proper choices of $\sigma_1, \cdots, \sigma_{n-s+1}$. ■

Chapter 6

PRESERVING STABILITY IN CONTROLLER REDUCTION

6.1 Introduction

As we discussed in chapter 4, one of the shortcomings of modern controller design packages is offering low order controllers without impacting the closed-loop stability and performance. We reviewed different methods which are developed for reducing high order controllers in section 4.4. All of these methods must consider maintaining the performance for the reduced controller and this focus sometimes violates the stability requirement of the closed loop system. For instance among the existing methods two sided frequency weighting controller reduction known as Enns' method [21] has been very successful in application. Unfortunately this method neither guarantees the stability of the reduced controller nor the closed loop system. In [60] some efforts to modify this method to obtain a stable low order controller are reviewed but in all approaches mentioned in [60] the stability of the reduced closed loop system is still open to doubt.

This chapter provides a solution to this problem in general by presenting a general framework to maintain the stability of the closed loop system. The main result chiefly contributes to LTI stable controllers, but some known tricks extend the result to special unstable cases. Our treatment of stability follows that of section 5.2, where stability is analyzed by utilizing the theoretic framework developed in Theorem 5.1 and exploiting strict Lyapunov inequalities. Section 6.4 expands the results by presenting the dual form of proved statements in section 6.2. As an application of our results in section 6.2, we modify the Enn's approach to guarantee the stability of the reduced controller. But the major improvement we are seeking is preserving the internal stability of the closed loop system that makes our modification superior to the existing ones as in [38, 65] and [60].

6.2 Parameterizing Stabilizing Reduced Controllers

To best present our result in controller reduction, we consider the closed loop system shown in Fig. 6.1 and we make the assumption that

$$\text{Plant} \qquad G(s) = \left[\begin{array}{c|cc} A_g & B_{g1} & B_{g2} \\ \hline C_{g1} & D_{11} & D_{12} \\ C_{g2} & D_{21} & D_{22} \end{array}\right] \qquad m_{th} \text{ order}$$

$$\text{Controller} \qquad K(s) = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right] \qquad n_{th} \text{ order}$$

Thus, the transfer matrix from ω to z is given by

$$T_{z\omega}(s) = \left[\begin{array}{c|c} A_t & B_t \\ \hline C_t & D_t \end{array}\right], \tag{6.1}$$

where

$$\begin{aligned} A_t &= \begin{bmatrix} A_g + B_{g2}MD_kC_{g2} & B_{g2}MC_k \\ B_kNC_{g2} & A_k + B_kND_{22}C_k \end{bmatrix} \\ B_t &= \begin{bmatrix} B_{g1} + B_{g2}MD_kD_{21} \\ B_kND_{21} \end{bmatrix} \\ C_t &= \begin{bmatrix} C_{g1} + D_{12}D_kNC_{g2} & D_{12}MC_k \end{bmatrix} \\ D_t &= D_{11} + D_{12}D_kND_{21} \\ M &= (I - D_kD_{22})^{-1}, \ N = (I - D_{22}D_k)^{-1}. \end{aligned}$$

At this point, we define x_g and x_k as the set of plant's and controller's states respectively. Note that we use the index g for the plant and k for the controller. Regarding the fact that the state vector in the closed loop system represented by (6.1) is equal to $\begin{bmatrix} x_g \\ x_k \end{bmatrix}$, in controller reduction we consider only similarity transformations with a block diagonal structure, e.g. $\begin{bmatrix} T_1 & 0 \\ 0 & T \end{bmatrix}$. In this way we prevent constructing a combination of x_g and x_k as a new state, therefore the truncated new realization results directly in a reduced controller. The following proposition parameterizes all stabilizing reduced order controllers derived this way.

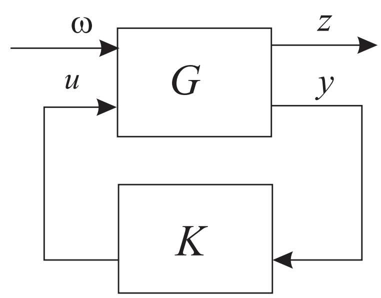

Figure 6.1: General feedback system

Proposition 6.1: For the closed loop system in (6.1), assume there exists a symmetric positive definite matrix $P = diag(P_g \in \mathbb{R}^{m\times m}, P_k \in \mathbb{R}^{n\times n})$ such that

$$A_t P + P A_t^T < 0, \tag{6.2}$$

and let $U = chol(P_k)$, then

$$K_r(s) = \left[\begin{array}{c|c} LA_kR & LB_k \\ \hline C_kR & D_k \end{array}\right], \tag{6.3}$$

where

$$L = XU^{-T}, \; R = U^T \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix}^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix} \tag{6.4}$$

for any *arbitrary* full row rank matrices $X \in \mathbb{R}^{k\times n}$ and $Y \in \mathbb{R}^{(n-k)\times(n-k)}$ $(1 \le k \le n)$, parameterizes a set of stabilizing k_{th} order reduced order controllers. Furthermore, assuming $D_{22} = 0$, $K_r(s)$ is stable.

Proof: In the following, we first show that a if we apply the following similarity transformation to (6.1)

$$\begin{bmatrix} T_1 & 0 \\ 0 & \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix} U^{-T} \end{bmatrix} \tag{6.5}$$

where $T_1 \in \mathbb{R}^{m\times m}$, $X \in \mathbb{R}^{k\times n}$ and $Y \in \mathbb{R}^{(n-k)\times(n-k)}$ are all arbitrary full row rank matrices and $U = chol(P_k)$, then $\hat{A}_t$ in the new realization $(\hat{A}_t, \hat{B}_t, \hat{C}_t, \hat{D}_t)$ satisfies

$$\hat{A}_t\hat{P} + \hat{P}\hat{A}_t^T < 0, \tag{6.6}$$

where $\hat{P}$ has a special block diagonal structure as follows:

$$\hat{P}{=}diag(\hat{P}_g \in \mathbb{R}^{m\times m}, P_{k1} \in \mathbb{R}^{k\times k}, P_{k2}). \tag{6.7}$$

To simplify the proof, we assume without any loss of generality that $T_1 = I_m$. We denote the lower block of (6.5) by T as shown below:

$$T = \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix} U^{-T} \tag{6.8}$$

By applying the similarity transformation in (6.5) ($[\begin{smallmatrix} I_m & 0 \\ 0 & T \end{smallmatrix}]$) to $T_{z\omega}$, we obtain a new realization $(\hat{A}_t, \hat{B}_t, \hat{C}_t, \hat{D}_t)$, where

$$\hat{A}_t = \begin{bmatrix} A_g + B_{g2}\hat{M}\hat{D}C_{g2} & B_{g2}\hat{M}\hat{C}_1 & B_{g2}\hat{M}\hat{C}_2 \\ \hat{B}_1\hat{N}C_{g2} & \hat{A}_{11} + \hat{B}_1\hat{N}D_{22}\hat{C}_1 & \hat{A}_{12} + \hat{B}_1\hat{N}D_{22}\hat{C}_2 \\ \hat{B}_2\hat{N}C_{g2} & \hat{A}_{21} + \hat{B}_2\hat{N}D_{22}\hat{C}_1 & \hat{A}_{22} + \hat{B}_2\hat{N}D_{22}\hat{C}_2 \end{bmatrix}$$

$$\hat{B}_t = \begin{bmatrix} B_{g1} + B_{g2}\hat{M}\hat{D}D_{21} \\ \hat{B}_1\hat{N}D_{21} \\ \hat{B}_2\hat{N}D_{21} \end{bmatrix}$$

$$\hat{C}_t = \begin{bmatrix} C_{g1} + D_{12}\hat{D}\hat{N}C_{g2} & D_{12}\hat{M}\hat{C}_1 & D_{12}\hat{M}\hat{C}_2 \end{bmatrix}$$

$$\hat{D}_t = D_{11} + D_{12}\hat{D}\hat{N}D_{21}$$

$$\hat{M} = (I - \hat{D}D_{22})^{-1}, \ \hat{N} = (I - D_{22}\hat{D})^{-1}, \hat{A}_{11} \in \mathbb{R}^{k\times k}$$

$$K(s) = \left[\begin{array}{cc|c} \hat{A}_{11} & \hat{A}_{12} & \hat{B}_1 \\ \hat{A}_{21} & \hat{A}_{22} & \hat{B}_2 \\ \hline \hat{C}_1 & \hat{C}_2 & \hat{D} \end{array}\right] = \left[\begin{array}{c|c} TA_kT^{-1} & TB_k \\ \hline C_kT^{-1} & D_k \end{array}\right]. \tag{6.9}$$

Inequality (6.2), together with application of Lemma 5.3, implies that $\hat{A}_t$ satisfies $\hat{A}_t\hat{P} + \hat{P}\hat{A}_t^T < 0$, where

$$\hat{P} = \begin{bmatrix} I_m & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} P_g & 0 \\ 0 & P_k \end{bmatrix} \begin{bmatrix} I_m & 0 \\ 0 & T \end{bmatrix}^T \tag{6.10}$$

Here we must show that $\hat{P}$ has a block diagonal structure as in (6.7). Substituting U^TU for P_k and T from (6.8) in (6.10) shows that

$$\hat{P} = \begin{bmatrix} P_g & 0 & 0 \\ 0 & XX^T & 0 \\ 0 & 0 & YY^T \end{bmatrix} > 0, \tag{6.11}$$

which completes the first part of the proof.

Based on this result, (6.2) implies that there exists a realization of $T(s) = (\hat{A}_t, \hat{B}_t, \hat{C}_t, \hat{D}_t)$ that $\hat{A}_t$ satisfies (6.6) and (6.7). Therefore applying Lemma 5.2 shows that the system derived by truncating $(\hat{A}_t, \hat{B}_t, \hat{C}_t, \hat{D}_t)$ to order $m + k$, denoted by $T_r(s)$ is stable.

$$T_r(s) = \left[\begin{array}{cc|c} A_g + B_{g2}\hat{M}\hat{D}C_{g2} & B_{g2}\hat{M}\hat{C}_1 & B_{g1} + B_{g2}\hat{M}\hat{D}D_{21} \\ \hat{B}_1\hat{N}C_{g2} & \hat{A}_{11} + \hat{B}_1\hat{N}D_{22}\hat{C}_1 & \hat{B}_1\hat{N}D_{21} \\ \hline C_{g1} + D_{12}\hat{D}\hat{N}C_{g2} & D_{12}\hat{M}\hat{C}_1 & D_{11} + D_{12}\hat{D}\hat{N}D_{21} \end{array}\right] \tag{6.12}$$

Note that the structure in (6.5) does not blend the plant's and controller's states, hence $T_r(s)$ can be presented as a general feedback system of $G(s)$ and a controller defined as follows

$$\hat{K}_r(s) = \left[\begin{array}{c|c} \hat{A}_{11} & \hat{B}_1 \\ \hline \hat{C}_1 & \hat{D} \end{array}\right].$$

Using (6.9) and (6.8) shows that $\hat{K}_r(s)$ is equal to $K_r(s)$ defined in (6.3). Hence $K_r(s)$ is a reduced order controller of order k that the general feedback system of $G(s)$ and $K_r(s)$, presented in (6.12), is stable, which is the desired result.

In the following we show if $D_{22} = 0$, then $K_r(s)$ is stable. Considering (6.6) and (6.11), it can be easily concluded that

$$(\hat{A}_{11} + \hat{B}_1\hat{N}D_{22}\hat{C}_1)(XX^T) + (XX^T)(\hat{A}_{11} + \hat{B}_1\hat{N}D_{22}\hat{C}_1)^T < 0$$

Assuming $D_{22} = 0$ we have

$$\hat{A}_{11}(XX^T) + (XX^T)\hat{A}_{11}^T < 0$$

Hence, by Theorem 5.1 $\hat{A}_{11}$ is a stability matrix and $K_r(s)$ is stable. ■

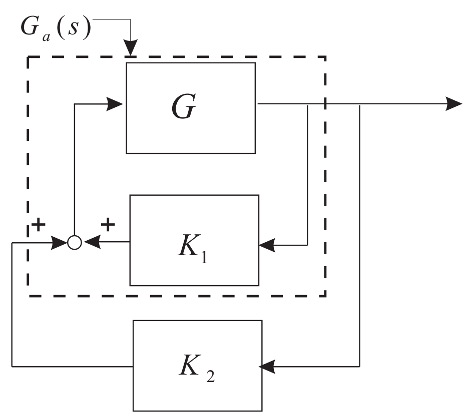

Figure 6.2: Stable-stabilizer decomposition of controller

Remark 6.1: In particular, if $D_{22} = 0,\ D_k = 0$, from inequality (6.2), we have

$$A_gP_g + P_gA_g^T < 0,\ A_kP_k + P_kA_k^T < 0$$

These inequalities imply that both A_g and A_k are stability matrices, therefore both the plant and the controller have to be stable. Hence the parameterization in Proposition 6.1 is not applicable to a system with either unstable plant or unstable controller. As a remedy, a simple idea proposed by [73] can be applied here as well. The idea is to compute an additive stable-stabilizer spectral decomposition of $K(s)$ as $K(s) = K_1(s) + K_2(s)$, where $K_1(s)$ stabilizes the plant and $K_2(s)$ is stable. If such a decomposition exists, denote the subsystem with the plant $G(s)$ and the controller $K_1(s)$ by $G_a(s)$ as shown in a dashed box in Fig. 6.2. Then Proposition 6.1 can be applied to $G_a(s)$ and $K_2(s)$. Assume that $K_{2r}(s)$ is a reduced order controller of $K_2(s)$, then $K_r(s) = K_1(s) + K_{2r}(s)$ is a reduced controller of $K(s)$. In some cases, other ideas like bilinear transformation and linear fraction transformation can also help.

Special Case

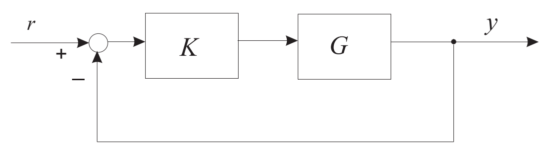

Figure 6.3: Negative feedback system

The results in this section can be easily applied to one degree of freedom control configuration. Considering the closed loop system shown in Fig. 6.3 and

$$\begin{array}{lll} \text{Plant} & G = \left[\begin{array}{c|c} A_g & B_g \\ \hline C_g & D_g \end{array}\right] & m_{th} \text{ order} \\ \text{Controller} & K = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right] & n_{th} \text{ order} \end{array}.$$

For simplicity we assume $D_g = 0$, $D_k = 0$. Thus, the transfer matrix from ω to z is given by

$$T_{z\omega} = \left[\begin{array}{cc|c} A_g & B_g C_k & 0 \\ -B_k C_g & A_k & B_k \\ \hline C_g & 0 & 0 \end{array}\right] = \left[\begin{array}{c|c} A_t & B_t \\ \hline C_t & D_t \end{array}\right]$$

Proposition 6.2: (special case of Proposition 6.1) Assume there exists $P = diag(P_g, P_k) > 0$, where $P_g \in \mathbb{R}^{m\times m}$ and $P_k \in \mathbb{R}^{n\times n}$, such that

$$A_t P + P A_t^T < 0$$

and let

$$K_r = \left[\begin{array}{c|c} L A_k R & L B_k \\ \hline C_k R & D_k \end{array}\right]$$

where

$$X \in \mathbb{R}^{k\times n} \text{ arbitrary}, \qquad U = chol(P_k),$$
$$L = X U^{-T}, \quad R = U^T \begin{bmatrix} X \\ X^{\perp} \end{bmatrix}^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix},$$

then the closed loop system with the reduced order controller K_r is stable. Furthermore K_r is also stable.

6.3 Numerical Issues and Solvability Condition

Numerical issues: For a stable closed loop system the inequality in (6.2) with the given constraint on the structure of P is a standard LMI problem. The development of computational methods for finding feasible solutions to LMIs, such as the Lyapunov inequalities considered here, has recently been a rather popular research area in the control community. Many efficient convex optimization algorithms have already been developed in particular by the authors of [14]. The following proposition gives a conservative sufficient condition for the feasibility of this problem.

Proposition 6.3: Assuming $D_{22}=0$, the LMI in (6.2) is solvable if

$$Z^T(A_g + B_{g2}D_kC_{g2} + (A_g + B_{g2}D_kC_{g2})^T)Z$$

is a stability matrix, where Z is an orthonormal matrix whose columns span $null(B_kC_{g2})$.

Proof: To prove Proposition 6.3, we require a preliminary lemma.

Lemma 6.1: If $Z^T(A_g+B_{g2}D_kC_{g2} + (A_g+B_{g2}D_kC_{g2})^T)Z$ is a stability matrix, then for any positive definite matrix P_k with appropriate dimensions, the pair of $(\tilde{A},\tilde{B})$ defined as

$$\tilde{A} = \underbrace{A_g+B_{g2}D_kC_{g2}}_{\hat{A}} - B_{g2}C_kP_k(A_kP_k+P_kA_k^T)^{-1}B_kC_{g2}, \qquad \tilde{B} = C_{g2}^TB_k^T \tag{6.13}$$

is stabilizable.

Proof: Assume that $Z^T(\hat{A}+\hat{A}^T)Z$ is a stability matrix, therefore we have

$$Re\{eig(\frac{1}{2}(Z^T(\hat{A}+\hat{A}^T)Z))\} = Re\{eig(\frac{1}{2}(Z^T\hat{A}Z+(Z^T\hat{A}Z)^T))\} < 0 \tag{6.14}$$

A non-symmetric matrix $(Z^T\hat{A}Z)$ is negative definite iff its symmetric part $(\frac{1}{2}(Z^T\hat{A}Z+(Z^T\hat{A}Z)^T))$ is a stability matrix, consequently, (6.14) implies $Z^T\hat{A}Z<0$. Note that $Z \subseteq null(\tilde{B})$, therefore

$$\begin{aligned}\hat{A}Z &= \hat{A}Z - B_{g2}C_kP_k(A_kP_k+P_kA_k^T)^{-1}\underbrace{B_kC_{g2}Z}_{\tilde{B}^TZ=0}\\ &= (\hat{A}-B_{g2}C_kP_k(A_kP_k+P_kA_k^T)^{-1}B_kC_{g2})Z = \tilde{A}Z.\end{aligned}$$

where P_k is a solution to $A_kP_k + P_kA_k^T < 0$ and consequently

$$Z^T\hat{A}Z = Z^T\tilde{A}Z < 0.$$

Thus, for any vector x (with an appropriate size) we have

$$\forall\lambda\in\mathbb{C}^+:\ x^T(Z^T\tilde{A}Z - \lambda Z^TZ)x < 0 \Rightarrow \underbrace{(Zx)}_{\in null(\tilde{B})}^T(\tilde{A}-\lambda I) \neq 0, \tag{6.15}$$

Noting that Z is an orthonormal matrix whose columns span $null(\tilde{B})$ and x is arbitrary, Zx spans $null(\tilde{B})$. Thus, (6.15) implies that $(\tilde{A}, \tilde{B})$ is stabilizable.

We now prove Proposition 6.3. The LMI in (6.2) is equal to

$$\begin{bmatrix} (A_g+B_{g2}D_kC_{g2})P_g+P_g(A_g+B_{g2}D_kC_{g2})^T & B_{g2}C_kP_k+P_gC_{g2}^TB_k^T \\ B_kC_{g2}P_g+P_kC_k^TB_{g2}^T & A_kP_k+P_kA_k^T \end{bmatrix} < 0 \tag{6.16}$$

Using Schur complements lemma, (6.16) is equivalent to

$$\underbrace{A_kP_k + P_kA_k^T}_{R} < 0, \tag{6.17}$$

$$(A_g + B_{g2}D_kC_{g2})P_g + P_g(A_g + B_{g2}D_kC_{g2})^T - \underbrace{(B_{g2}C_kP_k + P_gC_{g2}^TB_k^T)}_{S} R^{-1}S^T < 0, \tag{6.18}$$

$$S(I - RR^{-1}) = 0. \tag{6.19}$$

In other words the LMI in (6.16) is solvable iff the inequalities in (6.17), (6.18) and (6.19), with respect to variables P_k and P_g, are solvable. Since A_k is assumed to be a stability matrix, the LMI in (6.17) is solvable and has infinite solutions. Consequently R is a square negative definite matrix, therefore condition (6.19) is also fulfilled. Thus, (6.16) has a solution iff (6.18) is solvable. By rearranging (6.18) and using (6.13) we obtain

$$\tilde{A}P_g + P_g\tilde{A}^T - P_g\tilde{B}R^{-1}\tilde{B}^TP_g - B_{g2}C_kP_kR^{-1}P_kC_k^TB_{g2}^T < 0 \tag{6.20}$$

The inequality in (6.20) is a Riccati inequality and it has a solution if $(\tilde{A}, \tilde{B})$ is stabilizable. Applying Lemma 6.1 shows that (6.20) and consequently (6.2) is solvable if $Z^T(\hat{A} + \hat{A}^T)Z$ is a stability matrix, which is the desired result. ■

6.4 Dual Results

An interesting feature of the results in the previous sections is the validity of their dual forms. In this section the dual forms are presented concisely.

Proposition 6.4: (Dual of Proposition 6.1) Assume there exists a symmetric positive definite matrix $Q = diag(Q_g, Q_k)$ such that

$$A_t^T Q + QA_t < 0 \tag{6.21}$$

and let $U_q = chol(Q)$, then

$$K_r(s) = \left[\begin{array}{c|c} L_q A_k R_q & L_q B_k \\ \hline C_k R_q & D_k \end{array}\right] \tag{6.22}$$

where

$$L_q = XU_q, \quad R = U_q^{-1} \begin{bmatrix} X \\ YX^{\perp} \end{bmatrix}^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix}$$

for any *arbitrary* full row rank matrices $X \in \mathbb{R}^{k \times n}$ and $Y \in \mathbb{R}^{(n-k)\times(n-k)}$ $(1 \leq k \leq n)$, parameterizes a set of stabilizing k_{th} order reduced order controllers. Furthermore, assuming $D_{22} = 0$, $K_r(s)$ is stable.

Proposition 6.5: (Dual of Proposition 6.3) Assuming $D_{22} = 0$, the inequality in (6.21) is solvable if

$$Z_d^T((A_g + B_{g2}D_kC_{g2})^T + A_g + B_{g2}D_kC_{g2})Z_d$$

is a stability matrix, where Z_d is an orthonormal matrix, which its columns span $null(C_k^T B_{g2}^T)$.

6.5 Application Outlook

In this section an algorithm illustrates how we can benefit from the parameterization of stabilizing reduced controllers presented in (6.3). The idea is that we can exploit the result in Proposition 6.1 to modify other controller reduction methods based on similarity transformation in order to guarantee the closed loop stability. Assume $\tilde{T} \in \mathbb{R}^{n\times n}$ is a transformation matrix from a controller reduction approach, which does not guarantee the stability of closed loop system, like balancing, frequency weighted balancing and so forth. Note that $\tilde{T}$ acts on the full order controller. Using similar notation as in section 6.2, the closed loop system can be stabilized by projecting $\tilde{T}$ onto the space of stabilizing transformations introduced in Proposition 6.1. The construction of the projected transformation (T) is summarized as follows:

1. Find a $P = diag(P_g, P_k) > 0$, such that it satisfies $A_t P + P A_t^T < 0$. This is an LMI problem with a given structure for matrix P, which can be formulated easily by available software tools such as MATLAB LMI toolbox. The computation effort of solving this LMI is $\mathcal{O}((m+n)^6)$.

2. Calculate $U = chol(P_k)$. For calculating Cholesky factorization of a positive definite matrix, there is stable optimized code available in LAPACK software [3] which has the computation effort of order $\mathcal{O}(\frac{n^3}{3} + 3n^2)$.

3. Calculate the modified transformation by projecting $\tilde{T}$ onto the space of matrices defined in (6.8). This can be carried out in different ways, a simple suboptimal solution is given by

$$T = \begin{bmatrix} \tilde{T}_u U^T \\ \tilde{T}_l U^T (\tilde{T}_u U^T)^{\perp\dagger} (\tilde{T}_u U^T)^{\perp} \end{bmatrix} U^{-T},$$

   where $\tilde{T}_u$ denotes the upper k rows of $\tilde{T}$, $\tilde{T}_l$ denotes the rest lower rows of $\tilde{T}$ and the reduced controller is obtained as in (6.3). Since T has the same structure as in (6.8) using Proposition 6.1 the stability of the closed loop system is guaranteed. In this part an orthonormal basis for the null space of $\tilde{T}_u U^T$ can be obtained from the singular value decomposition algorithm, as implemented in [3], which has a computation effort of order $\mathcal{O}(12k^3)$.

Using the above algorithm, we can easily procure a solution to the problem of stabilizing the closed loop system resulted from the Enns' method (frequency weighted controller reduction). The effectiveness of the proposed technique is demonstrated by way of a numerical example in chapter 8.

Chapter 7

IMPACT OF CONTROLLER REDUCTION ON EMBEDDED CONTROLLERS

7.1 Introduction

A control system is an implemented strategy used to cause a physical system, or plant, to behave in a desired manner. As the control strategy increases in complexity, it becomes more difficult to apply analog components for its implementation. On one hand precise implementation of desired dynamic requires high accuracy stable components that work in different environmental conditions like ambient temperature. On the other hand the dynamics in an analog control loop always interact with each other that makes it more difficult to match desired controller characteristics. To alleviate these problems, in modern control systems the control strategy is typically implemented in software on microprocessors. The advantages of using digital control include improved measurement sensitivity, the use of digitally coded signals, digital sensors and transducers; reduced sensitivity to signal noise and the capability to easily reconfigure the control algorithm in software.

Perhaps more than any other factor, the development of microprocessors has been responsible for the explosive growth of embedded systems. Early microprocessors required many additional components and peripherals in order to perform any useful work. For example the first microprocessor was the Intel 4004, which found its way into calculators and other small systems, but required external memory and support chips. By the mid-80s, most of the previously external system components had been integrated into the same chip as the processor. This progress were due to increasing use of very large scale integration (VLSI) semiconductor fabrication techniques. A further extension of the integration is the single chip microcontroller, which adds analog and binary I/O (inputs/outputs), timers, and counters such that it is capable of carrying out real-time control functions with almost no additional hardware. The basic characteristics of a microcontroller are as follows:

- A built-in flash memory within the chip to store the control program.
- A built-in RAM for temporary data storage.
- A CPU that uses single bit instructions so that the limited program memory is effectively used.
- Built-in timers/counters which can be set by users.
- Built-in I/O ports (Digital and Analog) for easy interaction with external devices.

Some microcontrollers have additional features like communication and external interrupt modules [16].

After enormous use of microcontrollers in different applications, the term "embedded systems" was coined. An embedded system is a special-purpose system in which the computer is completely encapsulated by the device it controls. Unlike a general-purpose computer, such as a personal computer, an embedded system performs predefined tasks, usually with very specific requirements. Since the system is dedicated to a specific task, design engineers can optimize it, reducing the size and cost of the product. Embedded systems are often mass-produced, so the cost savings may be multiplied by millions of items[66]. The software written for embedded systems is often called firmware, and is stored in flash memory chips rather than a disk drive. This software often runs with limited hardware resources: small or no keyboard, screen, and little RAM memory. Embedded systems reside in machines that are expected to run continuously for years without errors, and in some cases recover by themselves if an error occurs. Therefore the software is usually developed and tested more carefully than that for personal computers, and unreliable mechanical moving parts such as disk drives, switches or buttons are avoided. Recovery from errors may be achieved with techniques such as a watchdog timer that resets the computer unless the software periodically notifies the watchdog.

As the cost of a microcontroller fell below one dollar, it became feasible to replace expensive analog components such as potentiometers and variable capacitors with digital electronics controlled by a small microcontroller. By the end of the 80s, embedded systems were the norm rather than the exception for almost all electronic devices, a trend which has continued since. Since there has been an enormous rise in processing power and functionality of microcontrollers, it might seem that control engineers are now able to implement even very complex controllers without taking care of hardware limitations. But it should be noted that for high volume productions minimizing cost is usually the primary design consideration. Engineers should typically select hardware that is just good enough to implement the necessary functions. The choice of hardware for embedded controllers is mostly related to the required flash memory, RAM and processing speed for the given task. It is shown through this chapter how the mentioned parameters are connected to the complexity (order) of the controller and how controller reduction can help us to make our product cost-effective. Section 7.2 describes a path from control loops to real-time programs by introducing a common structure used for implementing real-time controllers. Some important restrictions on embedded controllers like the limited processing speed and memory are discussed in section 7.3 and it is shown how controller reduction can help to deal with these constraints.

7.2 From Control Loops to Real-Time Programs

An analog control system uses analog devices to realize the control task, in contrast, a microcontroller uses digital electronics hardware as the heart of the controller. Like analog controllers, microcontrollers normally have analog elements at their periphery to interface with the real (analog) plant; thus, it is the internal structure of the controller that distinguishes digital controllers from analog ones. As a result of using digital processors, the signals in the microcontroller must be in form of digital signals, and the control system itself usually is treated mathematically as a discrete-time system. A block diagram of a digital control system is shown in Fig. 7.1. The (digital) microcontroller in this system receives the measurement signals in digital form and performs calculations in order to provide an output in digital form. Thus, an embedded control system uses digital signals and a (digital) microcontroller to control a process. The measurement data are converted from analog to digital form by means of the AD converter (analog-to-digital converter) shown in Fig. 7.1, which is normally provided inside the microcontroller chip. After processing the inputs, the digital computer provides an output in digital form. This output is then converted to analog form by the DA converter (digital-to-analog converter) shown in Fig. 7.1.

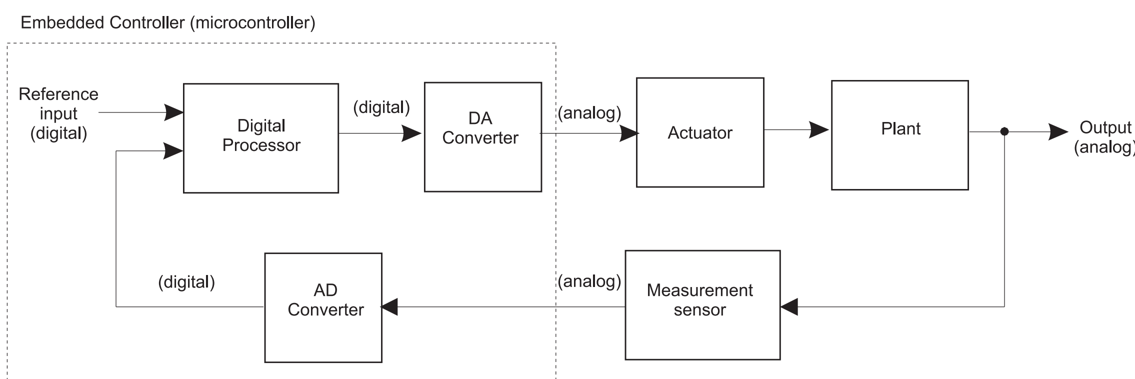

Figure 7.1: A block diagram of an embedded control system

Sample-rate selection, finding a discrete-time equivalents of continuous-time systems and quantization are three major issues in obtaining a real time program (discrete controller) from a continuous-time controller designed for the plant. In the following these issues are discussed.

Sample-rate selection

Sampling is the process of deriving a discrete-time sequence from a continuous-time function, this exactly what happens in the AD converter. Usually, but not always, the samples are evenly spaced in time. Reconstruction is the inverse of sampling; it is the formation of a continuous-time function from a sequence of samples, this exactly what happens in the DA converter. From Shanon's theorem it is known that in terms of sampling period, aliasing will not occur if $T < \frac{1}{2f_B}$, where T is the sampling time and f_B is the band-limit frequency[6]. From this theorem it is apparent that the sampling rate is a critical design parameter in the design of digital control systems. Usually, as the sampling rate is increased, the performance of a digital control system improves. However, computer costs also increase because less time is available to process the controller equations, and thus higher performance computer must be used. Additionally, for systems with AD converters, higher sample rates require faster AD conversion speed that may also increase system costs. On the other hand reducing the sample rate for the sake of reducing cost might degrade system performance or even cause instability. Aside form cost, the selection of sampling rate depends on some other factors like smoothness of the time response, effects of disturbances and sensor noise, parameter variation, and quantization. In general, the best sampling rate which can be chosen for a digital control system is the slowest rate that meets all performance requirements[6]. There are some rules of thumb that can help us to guess the proper sampling time. Usually the first figure of merit that the designer selects is the closed loop bandwidth, ω_{BT}, of the feedback system. Because ω_{BT} is related to the speed at which the feedback system can track the command input. As general rule, the sampling period should be chosen in the range

$$\frac{2\pi}{30} < T\omega_{BT} < \frac{2\pi}{5}$$

where ω_{BT} is in radians per second. Finding the proper sampling time in this range needs trial and error by simulation and of course experience could play a significant role.

Finding a discrete-time equivalents of continuous-time systems

There are different methods to convert a continuous-time controller designed for the plant to a discrete-time controller. These methods approximate the integrations of the continuous-time controller with discrete-time operations or try to match the step (or other) response samples to samples of the analog controller's step (or other) response. Discrete-time equivalents of continuous-time systems are discussed in great detail in [54]. Some of the well-known approaches are

- Zero-order hold (or n_{th}-order hold)[23]: Each incoming impulse to the zero-order hold produces a rectangular pulse of duration T (sampling period). In this method it is assumed that the control inputs are piecewise constant over the sampling period.

- Bilinear transformation (Tustin): This method is based on bilinear (Tustin) approximation to the derivative

$$
\begin{aligned}
z &= e^{sT} \approx \frac{1 + sT/2}{1 - sT/2} \\
s &\approx \frac{2}{T}\,\frac{z-1}{z+1}
\end{aligned}
$$

  A variation of this method is Tustin approximation with frequency pre-warping. This variation ensures the matching of the continuous- and discrete-time frequency responses at any deliberate frequency ω.

- Matched pole-zero: The matched pole-zero method applies only to SISO systems [36]. The continuous and discretized systems have matching DC gains and their poles and zeros correspond in the transformation $z = e^{sT}$.

Besides continuous- to discrete-time transformation there are some other points that must be considered. Fig. 7.2 shows a set of possible hardware and software components in form of block digram for water level control in a coupled tank system. In this figure it is assumed that the level of the water varies between 0 to 20 centimeters and

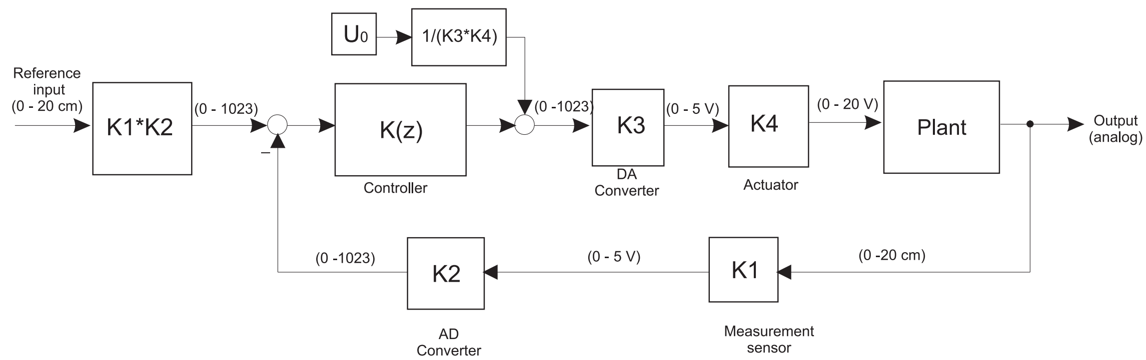

Figure 7.2: An example of possible additional gains in an embedded control system

a voltage between 0 to 20 Volts can be applied to the plant. Usually there is a sensor and an electronic circuit that convert the actual value of the output (in this case water level) to an electrical signal (in this case a voltage proportional to the level). Assuming that this block has a very fast dynamic that can be ignored, it is equivalent to a gain block presented by $K1(=\frac{1}{4})$. Then the output signal can be used as an input for the AD converter. For instance in this case, if we assume an AD converter with 10 bits resolution, the values between 0 to 5 Volts are linearly mapped to values between 0 to 1023 $(= 2^{10} - 1)$. Again ignoring the dynamic (or delay) in AD converter, it is equivalent to a gain block presented by $K2(\approx 205)$. Another point that should be noted is calculating the error signal by proper scaling of the reference input (block $K1 * K2$). It should be noted that the reference signal in Fig. 7.2 contains the nominal value of the output and is not only the deviation reference. Similarly, the gain blocks $K3(\approx \frac{1}{205})$ and $K4(= 4)$ represent the behavior of DA converter and the actuator respectively. Often in the design process only the dynamic of the plant is considered, but for implementation and finding $K(z)$, the additional blocks relating to interfacing components must be taken into account. For instance for the example in Fig. 7.2 the following relation has to be satisfied.

$$K4 * K3 * K(z) * K2 * K1 \approx K(s) \Rightarrow K(z) \approx K3^{-1} * K4^{-1} * K(s) * K1^{-1} * K2^{-1}$$

In additon Fig. 7.2 shows how and where the nominal value of the input must be added to the manipulating signal generated by the controller (U_0).

Quantization

Usually, continuous- to discrete-time conversion results in a controller whose coefficients have infinite precision. However embedded controllers are implemented with finite word length registers and finite precision arithmetic; therefore their signal and coefficients can attain only discrete values. For example consider a DC motor as the plant described by the following transfer function:

$$G(s) = \frac{1.5}{s^2 + 4s + 40.02}$$

The details of this model can be found in [42]. A PID controller with an approximate derivative term performs an acceptable closed loop behavior.

$$K(s) = 315.125 + \frac{148.245}{s} + \frac{2.135s}{100s + 1}$$

By selecting $T = 0.001$ as the sampling rate, an equivalent controller in discrete-time domain (using zero-order hold method) is obtained as follows:

$$K(z) = \frac{315.1463z^2 - 630.1413z + 314.9950}{z^2 - 2.0000z + 1.0000}$$

Because the controller is implemented with finite word length registers, each of its coefficients must be quantized. For instance if the binary forms of the coefficients of this controller are limited to 9 bits (including sign bit) for the left of the binary point and three bits to the right of the binary point, then the quantized controller transfer function becomes

$$K_q(z) = \frac{255.125z^2 - 255.125z + 255.875}{z^2 - 2.000z + 1.000}$$

It can be shown that the closed loop system with $K_q(z)$ is unstable! A simple solution to this problem is using the equivalent canonical state space representation of $K(z)$ as follows:

$$x(k+1) = \begin{bmatrix} 2.000 & -1.000 \\ 1.000 & 0.000 \end{bmatrix} x(k) + \begin{bmatrix} 0.500 \\ 0.000 \end{bmatrix} u(k)$$
$$y(k) = \begin{bmatrix} 0.255 & -0.255 \end{bmatrix} + 255.875u(k)$$

There are different methods for coefficient sensitivity analysis, for further details see [53]. This simple example showed that further analysis is needed to determine if the performance of a resulting digital controller in the presence of signal and coefficient quantization is acceptable.

Structure of an embedded software

After assigning the sampling rate and obtaining an equivalent discrete-time controller the embedded software can be programmed. The main program should contain functions to perform initialization, facilitate data access, and complete tasks before program termination. A sample initializing subroutine for ATMEL's ATmegaX microcontroller is as follows[24]:

```
void
ioinit (void)
{
/* Prescaler set to clk/8 in Timer 0 */
TCCR0 = (1<<CS01);
/* Enable timer 0 overflow interrupt. */
TIMSK = (1<<TOIE0);
sei ();
/* Timer 1 is 10-bit PWM, phase correct*/
//clear OCR1A on compare match
TCCR1A = (1<<COM1B1) | (1<<COM1A1) | (1<<WGM10) | (1<<WGM11);
/* Prescaler set to clk/8 in Timer 1 */
TCCR1B = (1<<CS10);//| (1<<WGM13);
OCR1A = 0x0200;
/* Enable PB1, PB2 as output. */
DDRB = (1<<DDB1)|(1<<DDB2);
/* Test Conversion */
/* ADC enable + Set prescaler to 8 */
```

```
ADMUX |= (1<<REFS1) | (1<<REFS0);
ADCSRA = (1<<ADSC) | (1<<ADEN) | (1<<ADPS1) | (1<<ADPS0);
// check if the conversion is completed
while ( ADCSRA & (1<<ADSC) );
}
```

In Fig. 7.3 an embedded software architectures for a simple control loop is presented. In this design, the software simply has a loop which calls subroutines to manage a part of the hardware or software[17, 18]. The subroutines then write their calculated data to the real-time model. This control loop routine reads inputs from the external hardware, calculates the model outputs, writes outputs to external hardware, with the sequence shown in Fig. 7.3.

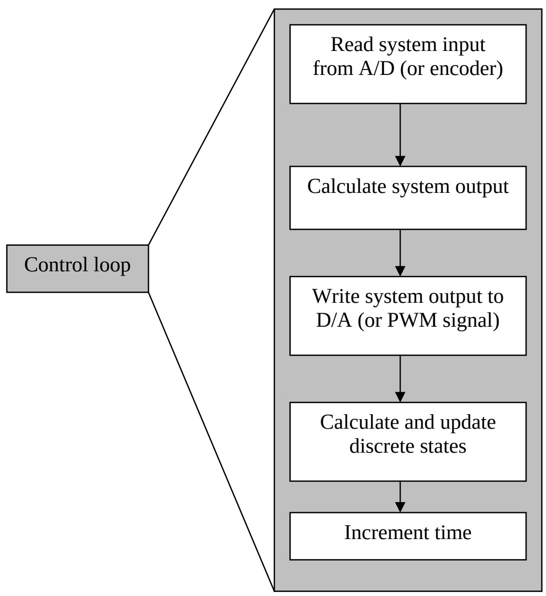

Figure 7.3: An embedded software architectures for a simple control loop

In order to decrease the computation costs, it is more appropriate to implement a controller with its state space representation in canonical form. A sample control subroutine for ATMEL's ATmegaX microcontroller is shown below.

```
#include "fixedpt.h" // fixed-point math support
/* Initial conditions*/
extern s32 x[n]={10.25,15.21};
/* Controller parameters */
#define n 2
#define U0 0x0F // operating point
extern s32 A[n]={-1.8,-0.15};
extern s32 B   =1;
extern s32 C[n]={1.24,2.41};
extern int32_t Reference_input   =8;
/* Timer0 overfolw interrupt (control loop)*/
ISR (TIMER0_OVF_vect)
{
/* Initializing the variables*/
uint8_t i=0; // counter
s32 x1temp =0; // buffer
s32 output =0;
s32 er_signal =0;
/* Calculate system output*/
for(i=0;i<n;++i)
output += C[i]*x[i];
/* Write output (PWM)*/
OCR1A = U0 + output;     /* Compare register */
/* Read system input */
input = ReadChannel(0);
er_signal=Reference_input-input;
/*Calculate and update states*/
for (i=0;i<n;++i)
x1temp +=A[i]*x[i];
for (i=1;i<n;++i)
```

```
x[i]=x[i-1];
x[0]=x1temp+B*er_signal;
}
```

Reading the measured variable is done be activating the AD converter. The following subroutine shows how this task can be carried out.

```
uint16_t ReadChannel(uint8_t chan)
{
uint16_t result = 0;
ADMUX  |= (1<<chan);    // Choose the input channel
/* AD Converter */
ADCSRA |= (1<<ADSC);    // Start Conversion
/* check if the conversion is completed */
while ( ADCSRA & (1<<ADSC) );
result=ADC;
return result;
}
```

Note that this scheme writes the system outputs to the hardware before the states are updated. Separating the state update from the output calculation minimizes the time between the input and output operations. If these computations require the plant output of the previous sample time, they must be performed in one sample interval. This implies synchronization between the execution of the logical program by the controller and the dynamic behavior of the plant in real time.

7.3 Restrictions on Embedded Systems and Controller Reduction

Flash Memory

The primary restriction on embedded systems is the size of memory allocated to the program (firmware). This memory is basically a non-volatile flash memory, which means that it does not need power to maintain the information stored in the chip and therefore is used to store the program that must be run on the microcontroller. A list of available flash memories for different ATmegaX series microcontrollers are given in Table 7.1. As it can be seen from Table 7.1, normally the other specifications of a microcontroller is proportional to the available flash memory[1].

Table 7.1: Atmel's microcontrollers: ATmegax series [1]

Microcontroller	flash (Byte)	RAM (Byte)	EEPROM (Byte)	price ($)
ATmega8	8K	512	256	2.98
ATmega16	16K	1K	512	6.05
ATmega32	32K	2K	1K	7.82
ATmega64	64K	4K	2K	11.33
ATmega128	128K	4K	4K	13.12
ATmega256	256K	8K	4K	16.24

As it was mentioned in the previous sections the main program (for control purposes) usually contains at least four major tasks as shown in Table 7.2. The number of instructions required to implement each task varies corresponding to the the application at hand. To have an idea about the range of flash memory needed for each task, some typical values are shown in Table 7.2. Assuming *float* variables for the controller parameters, the size of the corresponding memory depends linearly on the order of the controller. Our experiments on ATmegaX show the following relationship, that

[1] The prices are based on the information on the website of Arrow Electronics Inc. (Aug. 2006).

Table 7.2: ATmegax series

Tasks	flash (Byte)
Initialization	500-1500
Reading input	300-1000
controller	900-?
safety check	500-2000

is also shown graphically in Fig. 7.4.

$$\text{flash memory (KByte)} = \underbrace{672}_{\text{initialization}} + \underbrace{323}_{\text{reading input}} + \underbrace{7957}_{\text{controller init.}} + \underbrace{n}_{\text{order}} *37$$

It is worth to mention that there are different data types available with different sizes to assign to controller parameters. For instance in our case: *char* is 8 bits, *int* is 16 bits, *long* is 32 bits, *long long* is 64 bits, *float* and *double* are 32 bits. Besides these standard format, in order to compress the program size and decrease the computation effort, using fixed point integers and fixed point arithmetic toolbox is an effective approach.

Computation Time (Sampling Rate)

Another restriction on embedded systems is an upper bound on the computation time, which guarantees a given sampling rate. In other words, it is required to have plant input computed at a given input in time. As a result, the controller cannot rely on iterative computational schemes unless the upper bound of the iterations is fixed. Based on Fig. 7.3, the first part of the control loop program is reading the system input. Normally it is necessary to read the input from an AD converter. In ATmegaX microcontrollers this task takes around 13 clock cycles. After reading the input, the next step is calculating the system output and updating the output signal[2]. For a

[2]For instance if the output is a PWM signal it takes 2 clock cycles to update the corresponding counter/Timer.

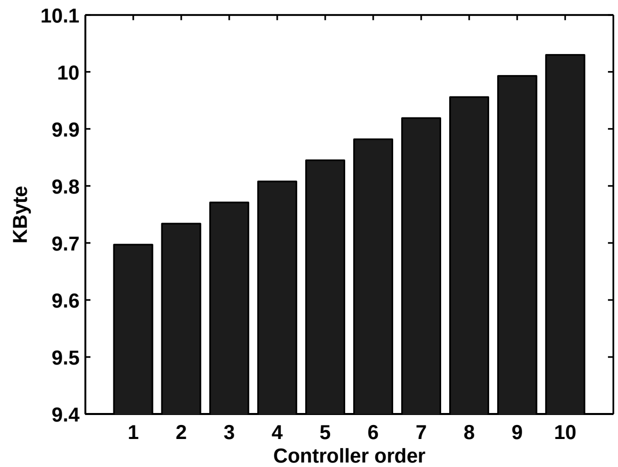

Figure 7.4: Required flash memory space vs. controller order

SISO controller based on our experiments the following relationship holds between the number of cycles to calculate the output variable and the order of the controller.

$$\text{cycles} \leq 1216 + \underbrace{n}_{\text{order}} * 520$$

After updating the controller output-variable, the next step is updating the state variables. For a SISO controller based on our experiments, an upper bound on the number of cycles to calculate the new state variables follows the following inequality

$$\text{cycles} \leq 1224 + \underbrace{n}_{\text{order}} * 568$$

where n is the order of the controller. As a summary the entire control loop calculations (Fig. 7.3), evaluated based on the number of clock cycles, nearly obey the following inequality.

$$\text{cycles} \leq \underbrace{2560}_{\text{initialization}} + \underbrace{n}_{\text{order}} * 1088$$

Given the order of controller this inequality implies a fixed upper bound for the controller response time. Fig. 7.5 graphically represents the effect of controller order

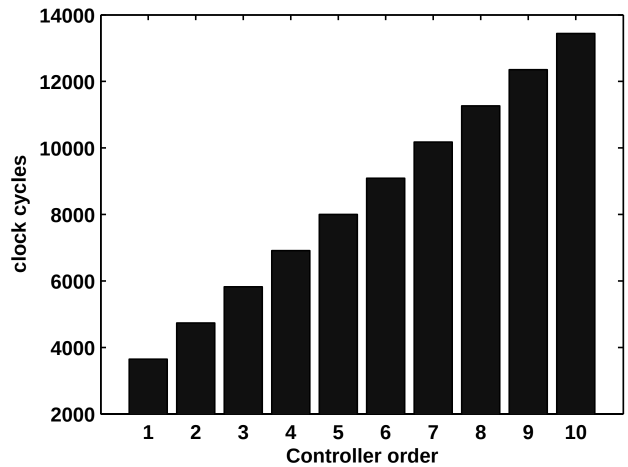

Figure 7.5: Clock cycles vs. controller order

on the clock cycles required to accomplish the whole calculations of the control loop. As it was discussed in the previous sections the sample rate is determined by control law analysis and depends on the time constants of the plant: the faster the plant, the higher the required sampling rate. To have a feeling about the effect of order on the controller response time, assume a $4MHz$ clock source, which means that every cycle takes $0.25\mu s$. Fig. 7.6 shows the computation time for different controller orders. For instance if the controller is of order 40, the microprocessor requires around $0.01s$ to calculate the entire control loop. Thus, a closed loop system with a bandwidth less than $4Hz$ can be realized, while in practice the required bandwidth usually is more than $4Hz$.

RAM Memory

Another restriction on embedded systems is the limited size of the RAM memory. It means that the number of variables and their sizes are restricted to the available RAM. A SISO controller is defined by its state space representation (canonical form) needs to save $3n + 6$ *float* number as detailed in Table 7.3. Assuming *float* variables

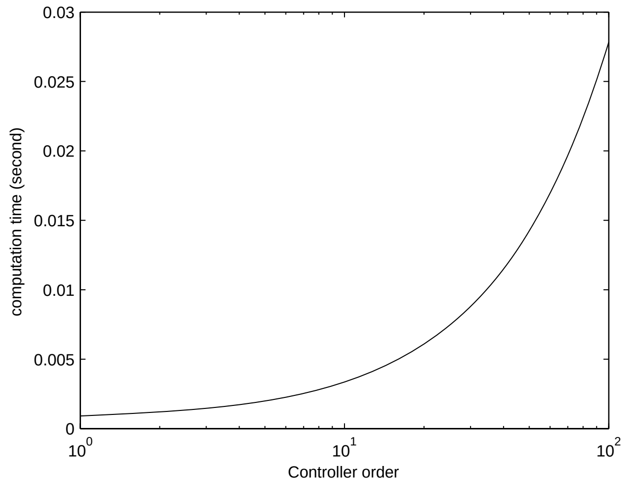

Figure 7.6: Computation time vs. controller order

for the controller parameters, it occupies $12n + 24$ Bytes of the RAM. The growth of the required RAM memory with respect to the controller order is depicted in Fig. 7.7.

Table 7.3: Required RAM memory for a controller of order n

Data	$A[n]$	B	$C[n]$	D	$x[n]$	y	$input$	$error$
size	n	1	n	1	$n+1$	1	1	1
RAM (Bit)	$n * 32$	$1 * 32$	$n * 32$	$1 * 32$	$(n+1) * 32$	32	32	32

By reviewing different constraints presented in this chapter, it seems the high computation time that implies low sample rate is the main problem in implementing high order controllers. There are two ways to overcome this problem, "*controller reduction*" and "*fixed point arithmetic*". Due to the nature of fixed point arithmetic it is often necessary to check and estimate the range of each number appears in the program and the result of each operation separately. Consequently, it is more convenient to apply fixed arithmetic to low order controllers rather than high orders. Therefore it is common to use the both approach to find an optimize code that accomplish its task in a given period of time and uses the minimum number of resources.

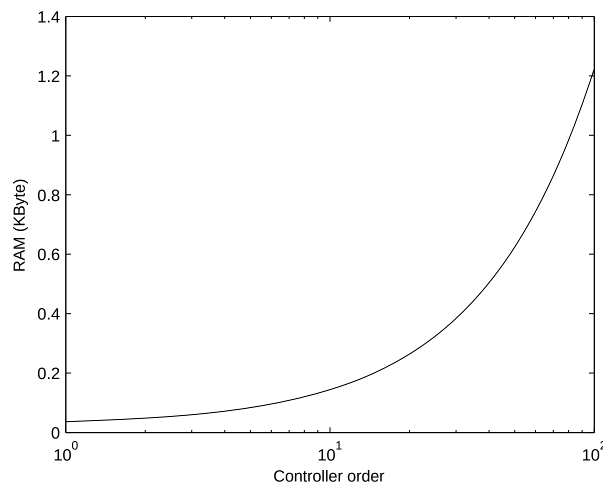

Figure 7.7: Required RAM memory vs. controller order

Part III

Applications

Chapter 8

ILLUSTRATIVE EXAMPLES

8.1 Example 1

Butterworth Filter (Output Krylov)

To illustrate the efficiency of the stabilizing method presented in details in chapter 5, a low-pass Butterworth filter of order 100 with cut-off frequency of 1 rad/sec and gain 1 in the passband is chosen. Model reduction of this plant by means of Krylov subspaces has been also studied in [5]. We approximate the system with models of order 30 to 39 using output Krylov subspace and we choose $R = L^T$. It should be noticed that the filter is not passive, therefore choosing $R = L^T$ does not guarantee the stability of the reduced models. The numerical results show that all reduced models are unstable, but by exploiting the modification algorithm in subsection 5.6.1 all reduced models are stabilized. Table 8.1 lists the reduction errors in terms of H_2 norm. The following abbreviations are used.

k	The order of the reduced model
U	The reduced model is unstable
KS_o	Output Krylov subspace method with $R = L^T$
KS_o^+	Modified Krylov subspace method as in Corollary 5.4

Fig. 8.1 shows the bode plot of the original and the reduced systems. Normally the quality of approximation reduces by decreasing the order of the reduced model. In this case our analysis shows that for orders between 19 and 39 the reduced models obtained

by output Krylov subspaces are all unstable. But with the stabilizing algorithm in Corollary 5.4 we can modify the reduction procedure to obtain stable reduced models. In general, the assessment of error introduced by the stabilization algorithm

Table 8.1: The stability and the H_2 norm of error

k	39	38	37	36	35	34	33	32	31	30
KS_o	U	U	U	U	U	U	U	U	U	U
KS_o^+	1.65	1.69	1.73	1.76	1.86	1.93	2.07	2.22	2.53	2.95

in Corollary 5.4 is not straight forward. Because, first there are some free parameters in the algorithm that has not been specified. Second distinct reduction methods serve different objectives that the error must be estimated according to that objectives. For instance in Krylov model reduction method the goal is matching k number of moments and it can be shown that the algorithm in subsection 5.6.1 maintains this property completely by leaving L unchanged. Therefore assigning proper values to the free parameters in the proposed algorithms and assessing their impact on the quality of reduction are matters that must be examined specifically for distinguish cases.

8.2 Example 2

Four Disk System (Enns' Method)

In order to show the efficiency of the algorithm in subsection 5.6.2, Enns' method was applied to the example studied by [21, 75] and [50]. The basic plant is a four disk system, which is embedded in a generalized plant. The physical plant comprises four spinning disks, connected by a flexible rod, with torque applied to the third disk. Angular displacement of the first disk is of interest. The plant is modeled by an eighth order system which has a double integrator and three vibration modes. A minimal

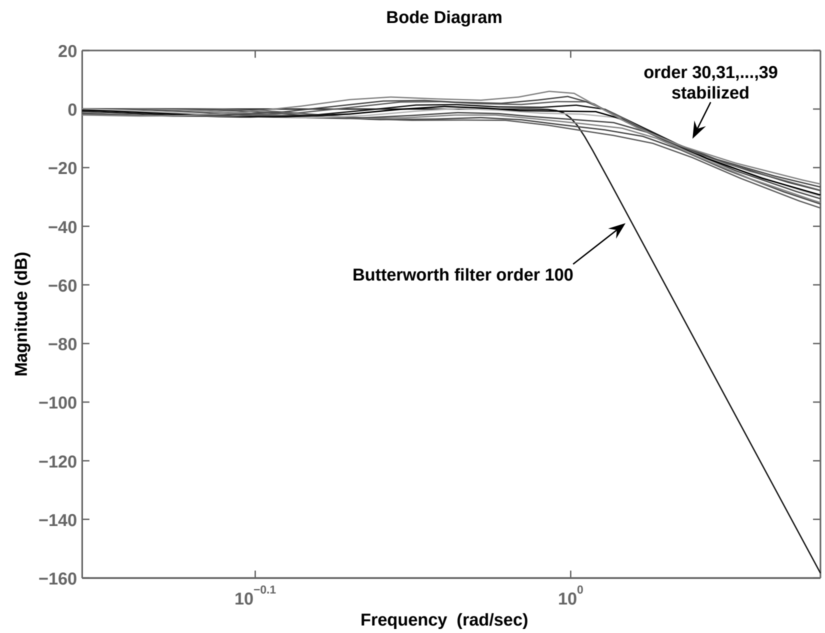

Figure 8.1: Bode plot of original and reduced systems for Butterworth filter

realization of the plant in modal coordinates is given by

$$A_g = diag\left(\begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}, \begin{bmatrix}-0.015 & 0.765\\ -0.765 & -0.015\end{bmatrix}, \begin{bmatrix}-0.028 & 1.410\\ -1.410 & -0.028\end{bmatrix}, \begin{bmatrix}-0.04 & 1.85\\ -1.85 & -0.04\end{bmatrix}\right),$$

$$B_g = \begin{bmatrix}0.026 & -0.251 & 0.033 & -0.886 & -4.017\times 2 & 0.145\times 2 & 3.604 & 0.280\end{bmatrix},$$

$$C_g = \begin{bmatrix}-0.996\times 3 & -0.105\times 3 & 0.261 & 0.009 & -0.001 & -0.043 & 0.002 & -0.026\end{bmatrix},$$

$$D_g = 0.$$

The basic plant is the four disk system, but this must be embedded in a generalized plant for controller design. The setup for the generalized plant and the system matrices are given in [75] and are as follows:

$$
\begin{aligned}
\dot{x} &= \begin{bmatrix} A_1 \\ \begin{bmatrix} I_7 & 0_{7\times 1} \end{bmatrix} \end{bmatrix} x + B_1 w + B_2 u, \\
z &= \begin{bmatrix} \sqrt{q_1} H \\ 0 \end{bmatrix} x + \begin{bmatrix} 0 & 1 \end{bmatrix}^T u, \\
y &= C_2 x + \begin{bmatrix} 0 & 1 \end{bmatrix} \omega, \\
A_1 &= \begin{bmatrix} -0.161 & -6.004 & -0.582 & -9.983 & -0.407 & -3.982 & 0 & 0 \end{bmatrix}, \\
B_2^T &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \\
B_1 &= \begin{bmatrix} \sqrt{q_2} B_2 & 0 \end{bmatrix}, \\
H &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0.55 & 11 & 1.32 & 18 \end{bmatrix}, \\
C_2 &= \begin{bmatrix} 0 & 0 & 6.4432 \times 10^{-3} & 2.31 \times 10^{-3} & 7.12 \times 10^{-2} & 1 & 0.10 & 55 & 0.99 \end{bmatrix}.
\end{aligned}
$$

where $q1 = 1 \times 10^{-6}$, $q2 = 1$, and ω is the disturbance, z is the controlled variable. The controller $K(s)$ feeds back y to u. The transfer function from ω to z is denoted by $T_{z\omega}(s)$. The optimal H_∞ norm for the closed loop system is $\gamma_{opt} = 1.12$. We choose $\gamma = 1.14$ to compute an 8_{th} order suboptimal controller K. The weighting transfer matrices are chosen as follows:

$$
W_1(s) = (I + G(s)K(s))^{-1}G(s), \quad W_2(s) = (I + G(s)K(s))^{-1} \tag{8.1}
$$

where $G(s)$ is the four disk plant and $K(s)$ is the suboptimal 8_{th} order controller. Fig. 8.2 shows the Bode diagram of $W_1(s)$ and $W_2(s)$ for the given plant and the controller. These choices of weighting matrices are very common in controller reduction, for more details see [50]. The controller is reduced using Enns' method and its three modifications presented in subsection 5.6.2. Table 8.2 shows the stability and the quality of reduction in the sense of H_∞ norm of weighted error ($\|W_1(s)(K(s)-K_r(s))W_2(s)\|_\infty$) and the following abbreviations are used:

k	The order of the reduced controller $K_r(s)$
U	The reduced controller is unstable
Enns	Enns' method, two sided frequency weighting [21]
LinChiu	Modified Enns by the algorithm presented in [38]
Wang	Modified Enns by the algorithm presented in [65]
Enns+	Modified Enns' method by the transformation presented in (5.14)

As it is shown in Table 8.2, Enns' method fails to maintain the stability of the reduced controller for all orders less than 6. In Lin and Chiu modification it is required that no pole-zero cancellations occur when forming $G(s)W_2(s)$ or $W_1(s)G(s)$. This restriction prevents the applicability of this method to solve controller reduction problems involving weights as in (8.1), where pole-zero cancellation takes always place. In order to apply this method $W_1(s)$ and $W_2(s)$ are approximated with two weights of order 8. The results show that the quality of reduction is not acceptable although the stability

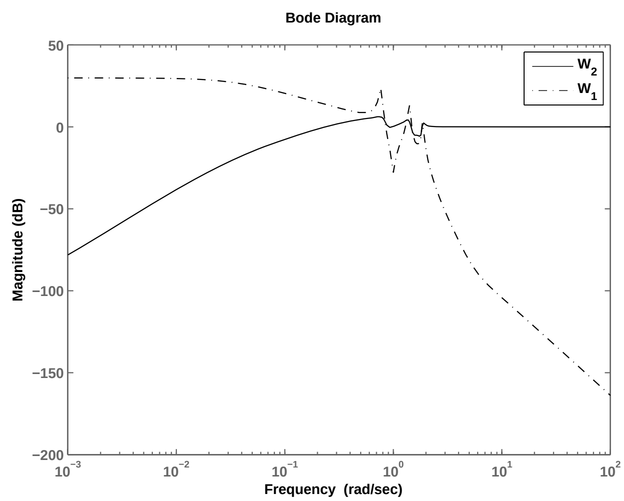

Figure 8.2: Bode diagram of input and output weights

of the reduced controller is guaranteed. Wang's method results in an high weighted error in comparison to Enns' approach. Using the results of Proposition 5.1 and the algorithm in subsection 5.6.2 the Enns' transformation matrices are replaced with the one in (5.14). For calculating the modified transformation matrix (5.14), a solution of $AP + PA^T < 0$ must be evaluated. In this example we used Matlab LMI toolbox [43] and the first feasible solution that only required 9 iterations was taken into account. Table 8.2 shows that the quality of reduced controllers by our new modification are in acceptable range and all reduced controllers are stable. Based on the error norms better results in comparison to other existing modification approaches are obtained. As it is mentioned by specified choices of the free parameters of the third alternative, it converges to Wang et al. approach or similarly to Lin and Chiu's approach. Therefore it seems possible to get even better results by finding an automatic procedure to assign proper values to the free parameters in our new method.

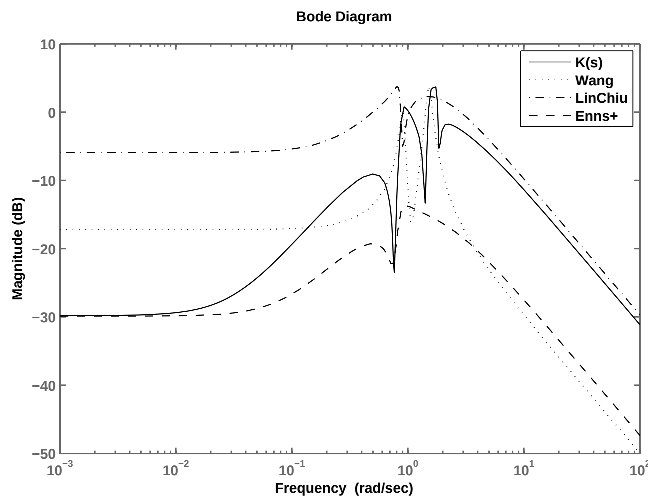

Figure 8.3: Bode diagram of reduced controllers of order 4

Fig. 8.3 compares the frequency response of the reduced 4_{th} order controllers derived by different approaches. The bode magnitude plots imply that each of the modification approaches has its own advantages. In this example Lin and Chiu's approach gives a better approximation for high frequencies, Wang et al.'s approach is more accurate in middle range frequencies and our new modification is more precise in low frequency range. Therefore it can be concluded that the free parameters in our approach can change the frequency range of interest in the reduction procedure.

Table 8.2: The stability and H_∞ norm of weighted error ($W_1(K-K_r)W_2$)

k	7	6	5	4	3	2
Enns	2.3471	2.3089	U	U	U	U
LinChiu	12.3092	6.1499	152.1967	43.2345	189.6318	65.1409
Wang	10.0073	7.4632	9.9924	12.9023	9.8409	12.1775
Enns+	2.3471	2.3089	2.5717	2.4843	2.5526	2.2459

8.3 *Example 3*

Closed Loop Stability

In order to show the efficiency of the algorithm in section 5.6, consider the closed loop system with the block diagram depicted in Fig. 8.4, where

$$G(s) = 0.014\frac{(s+14.82)(s+70.36)(s+105.4)(s+119.6)}{(s+120.2)(s+116.8)(s+74.68)(s+21.6)(s+1.178)}$$

is the plant and

$$K(s) = 0.505\frac{(s+8.56)(s+70.02)(s^2+235.6s+1.39e^4)}{(s+75.09)(s+21.8)(s+1.23)(s^2+226.6s+1.29e^4)}$$

is a 5_{th} order controller. Since our discussion is based on the general configuration depicted in Fig. 6.1, we converted the classic negative feedback in Fig. 8.4 to the general feedback structure depicted in Fig. 8.5. Starting from the general closed loop system in Fig. 8.5, the poles of the closed loop transfer function ($T_{z\omega}(s)$) are placed as follows:

$$p_1 = -120.16,\ p_2 = -116.78,\ p_{3,4} = -113.29 \pm 9.19i,\ p_5 = -75.09,$$
$$p_6 = -74.68,\ p_7 = -21.79,\ p_8 = -21.61,\ p_{9,10} = -1.20 \pm 0.02i$$

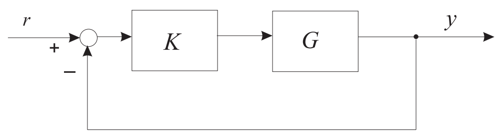

Figure 8.4: Negative feedback system

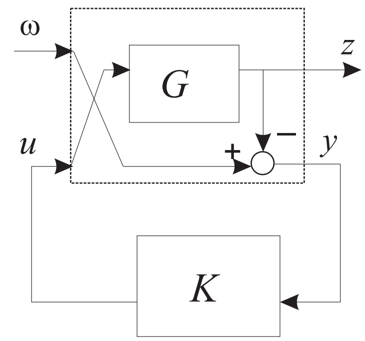

Figure 8.5: General feedback configuration

The closed loop performance and robustness against uncertainty is given by the constraints on the loop gain. To approximate $K(s)$ around $10rad/s$, two first order weighting functions $W_1(s)$ and $W_2(s)$ were selected as shown in Fig. 8.6. To find a reduced controller Enns' method was applied. Table 8.3 shows the poles and consequently the stability of the reduced closed loop systems. Note that the first row in Table 8.3 (k) indicates the order of reduced controller $K_r(s)$. In this example Enns's approach failed to maintain the stability of the closed loop system for order 8 and resulted in the following third order controller

$$K_r(s) = 0.513\frac{(s+8.59)(s-0.0061)}{(s+22.05)(s+1.23)(s-0.0052)}$$

that has an unstable pole in the right half plane. In order to guarantee the stability of the closed loop system, the algorithm in section 5.6 was used. The block diagonal matrix P required in step one was calculated by finding a solution to $A_tP+PA_t^T<0$,

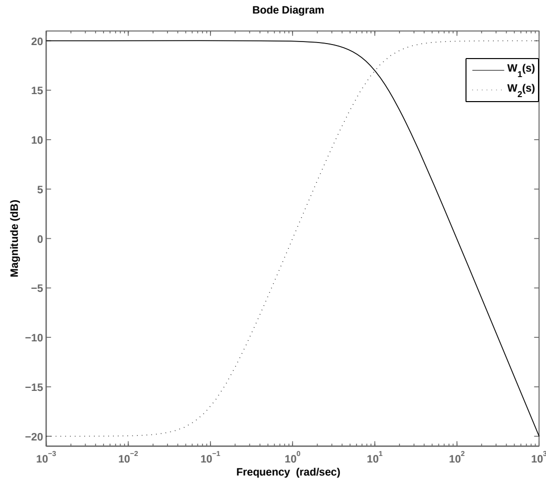

Figure 8.6: Bode plot of frequency weighting functions

which resulted in

$$\begin{bmatrix} 0.95 & 1.12 & 1.02 & 0.93 & 0.98 & 0 & 0 & 0 & 0 & 0 \\ 1.10 & 1.96 & 2.98 & 4.02 & 4.95 & 0 & 0 & 0 & 0 & 0 \\ 0.98 & 3.09 & 5.96 & 9.89 & 14.89 & 0 & 0 & 0 & 0 & 0 \\ 1.12 & 4.01 & 10.12 & 19.89 & 35.2 & 0 & 0 & 0 & 0 & 0 \\ 0.99 & 5.14 & 14.86 & 35.01 & 69.14 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.82 & 0.98 & 1.21 & 0.84 & 0.89 \\ 0 & 0 & 0 & 0 & 0 & 0.97 & 2.01 & 2.83 & 3.94 & 5.08 \\ 0 & 0 & 0 & 0 & 0 & 1.16 & 3.05 & 5.92 & 10.15 & 14.83 \\ 0 & 0 & 0 & 0 & 0 & 0.89 & 4.18 & 10.21 & 19.54 & 34.35 \\ 0 & 0 & 0 & 0 & 0 & 1.23 & 4.85 & 14.85 & 35.12 & 69.86 \end{bmatrix}.$$

By modifying the transformation matrix the following third order controller was obtained.

$$K_r^+(s) = 0.516\frac{(s+8.55)(s+76.53)}{(s+1.23)(s+21.8)(s+77.45)}$$

Table 8.3: The poles of $T_r(s)$

k	4	3	2	1
p_1	-120.16	-120.16	-120.16	-120.16
p_2	-116.78	-116.78	-116.78	-116.78
p_3	-109.12	-74.68	-74.68	-74.68
p_4	-76.48	-22.05	-22.05	-21.60
p_5	-74.68	-21.60	-21.60	-2.45
p_6	-21.80	-1.25	-1.25	-1.17
p_7	-21.59	-1.15	-1.15	-
p_8	-1.25	**+0.007**	-	-
p_9	-1.15	-	-	-

The relative error of the modified controller is

$$\frac{\|K_r(s) - K_r^+(s)\|_\infty}{\|K_r(s)\|_\infty} = 1.63e^{-4}$$

which indicates that $K_r^+(s)$ is a good approximation of $K_r(s)$ in the space of stable third order LTI controllers. The closed loop system as expected is stable and its poles are located as follows:

$$p_1 = -120.16,\ p_2 = -116.78,\ p_3 = -74.68,\ p_4 = -77.45,$$
$$p_5 = -21.79,\ p_6 = -21.61,\ p_{7,8} = -0.12 \pm 0.0002i$$

In general, the assessment of error introduced by the stabilization algorithm in section 6.5 is not straight forward. Because, first there are some free parameters in the algorithm that has not been specified. Therefore assigning proper values to the free parameters in the proposed algorithms and assessing their impact on the quality of reduction are matters that must be examined specifically for distinguish cases.

Chapter 9

CONCLUSIONS AND DISCUSSIONS

In this dissertation a general framework was proposed which guarantees the stability of reduced models in case of model reduction and the stability of closed loop system in case of controller reduction, which substantially enlarges the applicability of similarity transformation based approaches. The first contribution which concerns stability in model reduction methods based on similarity transformations was summarized in Proposition 5.1. This proposition parameterizes a large set of reduced models that preserve the stability of the original model. As an application of this result, it was shown how different model reduction methods can be modified, if they fail to maintain stability. These results provide also a complementary result to those of [25, 38, 60, 65] and [73]. The second contribution which isolates the problem of guaranteeing the internal stability of the reduced closed loop system from the problem of transfer matrix matching was summarized in Proposition 6.1. This proposition parameterizes a set of reduced controllers that preserve the stability of the closed loop system. The condition for existence of a solution and related numerical algorithm have been developed and studied. As an application of this result an algorithm was presented that helps to preserve stability in currently available reduction approaches that suffer from the lack of stability, by slightly modifying the corresponding similarity transformations. Using the Corollary 5.3 all results can be extended to reduced systems derived by singular perturbation approximation (instead of truncation) as well.

The framework presented in this dissertation only serves the *stability* in reduction methods. But there are some free parameters in this framework that were chosen arbitrary. To satisfy other objectives of reduction, these free parameters could be assigned with more caution. An approach to assign these free parameters could be an interesting topic for future works. If such an approach is developed, it will be a rival to the existing reduction methods. These free parameter even can be used for the assessment of error introduced by the stabilization algorithms in section 5.6 and section 6.5. By assigning proper values to these free parameters in the proposed algorithms, their impact on the quality of reduction can be assessed more precisely.

Based on these studies, some questions have been arisen that can be the topic of future works in the field of model and controller reduction. Among different approaches in this field the reduction methods based on the balancing and truncation method (B&T) are really interesting. But there are still some issues and ideas related to B&T which has not been fully investigated. The following shows some of these issues:

- When B&T fails?
  By approximating a matrix with a lower rank one, there is a direction that we lose accuracy! Because B&T based on a similar approach (svd based), there must be a set of inputs, which the reduced order system reacts completely different from the original one, although the omitted Hankel singular values are small. "How this set can be found?"
  application: In some applications it is better "to tune" the model reduction tools to certain classes of inputs. Therefore, it might be possible to tune B&T to a specified class of inputs!

- Structured balancing!
  The transformation for balancing ($x_b = Tx$)[1] can be found by :

  $$\min_T trace[TPT^T + T^{-T}QT^{-1}]$$

  and the minimum is equal to $2\sum_i \sigma_i$.
  idea: A structured T that minimizes the above cost, could be useful for controller reduction and structured balancing. As an example for a third order system, suppose we want the third state in the process of balancing (finding

[1] x_b is the new (balanced) state vector and x is the original state vector.

the (non)dominant states) be independent of x_1. This requirement impose the following structure on T:

$$x_b = Tx, \quad T = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ 0 & T_{32} & T_{33} \end{bmatrix}$$

Thus, instead of minimizing the above cost we can minimize the interaction of x_3 with the other states, input and output considering the given structure on T. In fact we are only looking for T_{32} and T_{33} and (assuming T to be full rank) the other elements are free! This idea can be exploited for iterative balancing and controller reduction. The question should be answered is: "How to represent controllability, observability, interaction and cost function such that the resulted optimization problem be tractable?"

- Iterative Balancing!
  Roughly speaking, the idea behind B&T is constructing a new state (nondominant state) which is a linear combination of the system's states and has low observability, controllability and consequently low interaction with other states. If we represent these specification properly, it might be possible to construct the most nondominant state which is boycotted by other states. In this way we're not supposed to apply a similarity transformation to the whole system and B&T can be carried out iteratively.

Low order controller design should find the minimum order controller that guarantees stability and performance. As discussed in section 4.3, the order of the minimal stabilizer can be searched using the following matrix inequalities:
If $k \geq k_{min}$, there are symmetric matrices R, S and a scalar $\gamma > 0$ such that

$$\begin{aligned} &1) \quad AR + RA^T < BB^T, \\ &2) \quad A^T S + SA < C^T C, \\ &3) \quad \begin{bmatrix} \gamma R & I \\ I & \gamma S \end{bmatrix} \geq 0 \\ &4) \quad Rank \begin{bmatrix} \gamma R & I \\ I & \gamma S \end{bmatrix} \leq n + k \end{aligned}$$

Finding k_{min} is a nonconvex optimization problem and solving the above LMIs with rank constraint is not straight. Even if we suppose that the minimal order is found, an algorithm to construct the minimal controller is still missed. Therefore the following question has not been completely answered "How to design a minimal order stabilizer?".

In the same field there are several methods of constructing strongly stabilizable controllers. Most of these methods involve constructing a stable transfer matrix, satisfying certain interpolation conditions, that usually results in large order controllers. Thus, an algorithm to construct a minimal stabilizable controller sounds challenging. The question should be answered is "How to design a minimal stabilizer for a plant, which satisfies the parity interlacing property?".

GLOSSARY

Abbrevations

AD converter: analog to digital converter

B&T: balance and truncation

CPU: central processing unit

DA converter: digital to analog converter

I/O: input/output

ISE: integral square error

LTI: linear time invariant

LMI: linear matrix inequality

lcf: left coprime factorization

PID: proportional-integral-derivative controller

PWM: pulse width modulation

MIMO: multiple input multiple output system

RAM: random access memory

rcf: right comprime factorization

SISO: single input single output system

VLSI: very large scale integration

vs.: versus

General symbols

$\mathbb{R}$: set of real numbers.

$\mathbb{C}$: set of complex numbers.

$tr(A)$: trace A.

$null(A)$: null space of A.

$\lambda_i(A)$: i_{th} eigenvalue of A.

$\overline{\sigma}(A)$: the largest singular value of A.

$\underline{\sigma}(A)$: the smallest singular value of A.

$\sigma_i(A)$: i_{th} singular value of A.

$\rho(A) = \max_i |\lambda_i(A)|$: spectral radius of A.

$\gamma(G) = \overline{\sigma}(G)/\underline{\sigma}(G)$: condition number.

$Mp\omega$: The maximum peak of $T(s)$ in $\frac{rad}{sec}$

M_{pt}: Overshoot of step response

ω_B: Bandwidth in terms of $S(s)$ in $\frac{rad}{sec}$

ω_{BT}: Bandwidth in terms of $T(s)$ in $\frac{rad}{sec}$

ω_c: Bandwidth in terms of $L(s)$ in $\frac{rad}{sec}$

$\mu(M)$: structured singular value:

$$\mu(M)^{-1} \triangleq \min_{\Delta}\{\overline{\sigma}(\Delta)|det(I - M\Delta) = 0 \text{ for structured } \Delta\}$$

which means "Find the smallest structured Δ (measured in terms of $\overline{\sigma}(\Delta)$) which makes $det(I - M\Delta) = 0$, then $\mu(M) = \frac{1}{\overline{\sigma}(\Delta)}$". Clearly, $\mu(M)$ depends not only on M but also on the allowed structure for Δ. This is sometimes shown explicitly by using the notation $\mu_\Delta(M)$.

$\mathcal{RH}_\infty$: The set of all real rational (prefix $\mathcal{R}$) functions bounded on $Re(s) = 0$ (including at ∞) and analytic in $Re(s) > 0$.

Some definitions

The whole closed loop, continuous system:

$$\mathcal{S} : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases} = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = C(sI - A)^{-1}B + D$$

where, u_i could be any possible input of the system like reference signal r, input disturbance d_i, output disturbance d_o or measurement noise n.

$$\text{Plant} \quad G = \left[\begin{array}{c|c} A_g & B_g \\ \hline C_g & D_g \end{array}\right]$$

$$\text{Controller} \quad K = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right]$$

$$\text{Measurement} \quad F = \left[\begin{array}{c|c} A_f & B_f \\ \hline C_f & D_f \end{array}\right]$$

The loop transfer function $L(s) = F(s)G(s)K(s)$

$\mathcal{F}_l(G, K)$: lower linear fractional transformation (more details in A.7 [55])

Vector (Holder) norms:

$$\begin{aligned}
\|v\|_p &= (\sum_i |v_i|^p)^{1/p},\ p \geq 1 \\
\|v\|_2 &= \sqrt{\sum_i |v_i|^2} \\
\|v\|_\infty &= \max_i |a_i|
\end{aligned}$$

Matrix (induced) norms:

$$\begin{aligned}
\|A\|_p &= \max_{v \neq 0} \frac{\|Av\|_p}{\|v\|_p} \\
\|A\|_1 &= \max_j (\sum_i |a_{ij}|) \quad \text{“maximum column sum”} \\
\|A\|_2 &= \overline{\sigma}(A) = \sqrt{\rho(A^H A)} \quad \text{“singular value or spectral norm”} \\
\|A\|_\infty &= \max_i (\sum_j |a_{ij}|) \quad \text{“maximum row sum”}
\end{aligned}$$

Matrix (Schatten) norms:

$$\|A\|_p = \begin{cases} (\sum_{i=1}^m \sigma_i^p(A))^{\frac{1}{p}}, & 1 \leq p < \infty \\ \overline{\sigma}(A), & p = \infty \end{cases}$$

Note that the Schatten infinity norm is equal to induced 2-norm. The Schatten 2-norm is also called Frobenius norm and can be evaluated simpler as follows:

$$\|A\|_F = \sqrt{\sum_{ij} |a_{ij}|^2} = \sqrt{tr(A^H A)} \quad \text{“Frobenius or Euclidean norm”}$$

In addition to the four norm properties, matrix norm also satisfies multiplicative property ($\|AB\| \leq \|A\|\|B\|$).

Signal Norms:

$$\begin{aligned}
\|u\|_1 &= \int_{-\infty}^{\infty} \sum_i |u_i(t)| dt \\
\|u\|_2 &= \sqrt{\int_{-\infty}^{\infty} \sum_i |u_i(t)|^2 dt} \\
\|u\|_\infty &= \sup_t (\max_i |u_i(t)|) \\
\|u\|_{pow} &= \sqrt{\lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{T} \sum_i |u_i(t)|^2 dt}
\end{aligned}$$

$\| \quad \|_{pow}$ is just a semi-norm because it fulfills all properties of a norm except that some nonzero signals have zero average power.

System Norms:

$$\begin{aligned}
\|G(s)\|_1 &= \int_{-\infty}^{\infty} \overline{\sigma}(g(t))dt \\
\|G(s)\|_2 &= \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} tr(G(j\omega)^H G(j\omega))d\omega} = \sqrt{\int_{-\infty}^{\infty} \underbrace{tr(g(\tau)^T g(\tau))}_{\|g(\tau)\|_F^2 = \sum_{ij}|g_{ij}(\tau)|^2} d\tau} \\
&= \sqrt{tr(CPC^T)} = \sqrt{tr(B^T QB)} \\
\|G(s)\|_\infty &= \max_{\omega} \overline{\sigma}(G(jw)) \\
\|G(s)\|_H &= \sqrt{\rho(PQ)}
\end{aligned}$$

where $P = \int_0^\infty e^{A\tau} BB^T e^{A^T_\tau} d\tau$ and $Q = \int_0^\infty e^{A^T_\tau} C^T C e^{A\tau} d\tau$ are the controllability and observability Gramians Respectively.

BIBLIOGRAPHY

[1] http://www.atmel.com/products/avr/.

[2] Anderson, B. D. O. A note on the Youla-Bongiorno-Lu condition. *Automatica*, 12:387–388, 1976.

[3] Anderson, E., Z. Bai,C. Bischof, S. Blackford, J. Demmel,J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney and D. Sorensen. *LAPACK User's Guide*. SIAM, Philadelphia, 1999.

[4] Anderson,B. D. O., Bose, N. K. and Jury, E. I. Output feedback stabilization and related problems: solution via decision methods. *IEEE Trans. Automat. Contr.*, 20:53–66, 1975.

[5] Antoulas, A. C. *Approximation of Large-Scale Dynamical Systems*. SIAM, 2005.

[6] Astrom, K. J. and Wittenmark, B. *Computer Controlled Systems-Theory and Design*. Prentice-Hall, 1996.

[7] Athans, M. The role and use of the stochastic linear-quadratic-gaussian problem in control system design. *IEEE Trans. on Automatic Control*, 16:529–552, 1971.

[8] Barrat, C. and Boyd, S. Example of exact trade-offs in linear controller design. *IEEE Control System Magazine*, pages 46–52, January 1989.

[9] Beard, R. W. Linear operator equations with applications in control and signal processing. *IEEE Control Systems Magazine*, pages 69–79, Apr. 2002.

[10] Beck, C.L., Doyle, J. and Glover, K. Model reduction of multidimensional and uncertain systems. *IEEE Trans. on Automatic Control*, 41:382–387, 1996.

[11] Bellman, R.E. *Stability Theory of Differential Equations*. Dover, 1953.

[12] Bennett, S. A brief history of automatic control. *Control Systems Magazine, IEEE*, 16(3):17–25, June 1996.

[13] Bernstein, D.S. Feedback control: an invisible thread in the history of technology. *Control Systems Magazine, IEEE*, 22(2):53–68, April 2002.

[14] Boyd, S., El Ghaoui, L., Feron,E. and Balakrishnan, V. *Linear Matrix Inequalities in System and Control Theory.* SIAM, 1994.

[15] Byrnes, C. I. and Anderson,B. D. O. Output feedback and generic stabilizability. *SIAM J. Control and Optimization*, 22:362–380, 1984.

[16] Cates, R. Processor architecture considerations for embedded controller applications. *Micro, IEEE*, 8:28–38, 1988.

[17] Cervin, A. and Eker, J. Feedback scheduling of control tasks. In *Proc. 39th IEEE Conf. Decision and Control*, pages 4871–4876, 2000.

[18] Correa, C.R and Awad, S.S. Embedded controller software and algorithm development tool. *Instrumentation and Measurement, IEEE Transactions on*, 52:885–890, 2003.

[19] Doyle, J., Francis, B and Tannenbaum, A. *Feedback Control Theory.* 1990.

[20] El Ghaoui, L. and Gahinet, P. Rank minimization under lmi constraints: A framework for output feedback problems. In *Proc. 32nd CDC*, 1993.

[21] Enns, D. F. Model reduction with balanced realizations: An error bound and a frequency weighted generalization. pages 127–132. Proceedings of 23rd IEEE Conference on Decision and Control, 1984.

[22] Ackermann, J. et al. *Robust Control, The parameter space approach.* Springer, 2002.

[23] Franklin, G.F., Powell, J.D. and Workman, M.L. *Digital Control of Dynamic Systems.* Addison-Wesley, 1990.

[24] Gadre, D. *Programming and Customizing the AVR Microcontroller*. McGraw-Hill, 2000.

[25] Gawronski, W. and Juang, J.-N. Model reduction in limited time and frequency intervals. *International Journal of Systems Science*, 21:349376, 1990.

[26] Goddard, P. J. and Glover, K. Controller reduction: weights for stability and performance preservation. In *Proc. 32nd CDC*, pages 2903–2908, 1993.

[27] Grimme, E. , Sorensen, D. and Van Dooren, P. Model reduction of state space systems via an implicitly restarted lanczos method. *Numerical Algorithms*, 128:1–31, 1996.

[28] Grimme, E. J. *Krylov Projection Methods for Model Reduction*. PhD thesis, Dep. of Electrical Engineering, University of Illinois at Urbana Champaign, 1997.

[29] Gugercin, S. and Antoulas, A. C. A survey of model reduction by balanced truncation and some new results. *International Journal of Control*, 77(8):748–766, 2004.

[30] Halevi, Y. Can any reduced order model be obtained via projection? In *Proc. ACC 2004*, pages 113–118, 2004.

[31] Datta, A. Ho, M.-T. and Bhattacharayya, S. P. A new approach to feedback stabilization. In *Proc. 35th CDC*, 1996.

[32] Hyland, D. C. and Bernstein, D. S. The optimal projection equations for model reduction and relationships among the methods of Wilson, Skelton, and Moore. *IEEE Trans. Automat. Contr.*, 29:1034–1037, 1984.

[33] Iwasaki, T. and Skelton, R. E. All controllers for the general h_∞ control design problem: Lmi existence conditions and state space formulas. *Automatica*, 30:1307–1317, 1994.

[34] Jaimoukha, I. M. and Kasenally, E. M. . Implicitly restarted krylov subspace methods for stable partial realizations. *SIAM J. Matrix Anal. Appl.*, 18:633–652, 1997.

[35] Khalil, H.K. *Nonlinear Systems*. Prentice Hall, third edition, 2002.

[36] Franklin, G.F. Kollr, I. and Pintelon, R. On the equivalence of z-domain and s-domain models in system identification. In *Proceedings of the IEEE Instrumentation and Measurement Technology Conference*, pages 14–19, 1996.

[37] Lall, S. and Beck, C. Error-bounds for balanced model-reduction of linear time-varying systems. *IEEE Trans. on Automatic Control*, 48:946–956, 2003.

[38] Lin, C. A. and Chiu, T. Y. Model reduction via frequency weighted balanced realization. *Control Theory and Advanced Technology*, 8:341–351, 1992.

[39] Liu, Y. , Anderson, B. D. O. and Ly, U. L. Coprime factorization controller reduction with bezout identity induced frequency weighting. *Automatica*, 26:233–249, 1990.

[40] Lu, W. M., Zhou, K. and Doyle, J. Stabilization of uncertain linear systems: an LFT approach. *IEEE Trans. on Automatic Control*, 41:50–65, 1996.

[41] Luenberger, David G. *Introduction to Dynamic Systems Theory, Models & Applications*. John Wiley & Sons, 1979.

[42] The MathWorks, Natick, MA. *Control Toolbox User's Guide, version 6*, 2006.

[43] The MathWorks, Natick, MA. *LMI Toolbox User's Guide*, 2006.

[44] McFarlane, D.,Glover, K. and Vidyasagar, M. Reduced-order controller design using coprime factor model reduction. *IEEE Trans. Automat. Contr.*, 35:369–373, 1990.

[45] Mesbahi, M. On the rank minimization problem and its control applications. *System & Control Letters*, 33:31–36, 1998.

[46] Mevenkamp, M. and Linnemann, A. Generic stabilizability of single input single output plants. *System & Control Letters*, 10:93–94, 1988.

[47] Moore, B.C. Principal component analysis in linear systems: controllability, observability and model reduction. *IEEE Trans. on Automatic Control*, AC-26:17–31, 1981.

[48] Morari, M. and Zafiriou, E. *Robust Process Control*. Prentice-Hall, 1989.

[49] Mustafa, D. and Glover, K. Controller reduction by $\mathcal{H}_\infty$ balanced truncation. *IEEE Trans. Automat. Contr.*, 36:668–682, 1991.

[50] Obinata, G. and Anderson, B. D. O. *Model Reduction for Control System Design*. Springer, 2001.

[51] Pace, I. S. and Barnett, S. Comparison of numerical methods for solving lyapunov matrix equations. *International Journal of Control*, 15:907–915, 1972.

[52] Pernebo, L. and Silverman, L. Model reduction via balanced state space representations. *IEEE Trans. on Automatic Control*, 27:1466–1477, 1996.

[53] Sanita, M.S. and Stubberud, A.R. *Quantization Effects*. CRC Press, Boca Raton, FL, 1996.

[54] Sanita, M.S. and Stubberud, A.R. *Discrete-Time Equivalents to Continuous-Time Systems*. Eolss Publishers Co. Ltd., Oxford, 2004.

[55] Skogestad, S. and Postlethwaite, I. *Multivariable Feedback Control, Analysis and Design*. John Wiley, 2001.

[56] Smith, M. C. On minimal order stabilization of single loop plants. *System & Control Letters*, 7:39–40, 1986.

[57] Smith, M. C. and Sondergeld, K. P. On the order of stable compensators. *Automatica*, 22:127–129, 1986.

[58] Sreeram, V. and Anderson, B. D. O. Frequency weighted balanced reduction technique: a generalization and an error bound. In *Proc. 34th CDC*, pages 3576–3581, 1995.

[59] Varga, A. and Anderson, B. D. O. Accuracy-enhancing methods for balancing-related frequency-weighted model and controller reduction. *Automatica*, 39:919–927, 2003.

[60] Varga, A. and Anderson, B. D. O. Accuracy-enhancing methods for balancing-related frequency-weighted model and controller reduction. *Automatica*, 39:919–927, 2003.

[61] Vidyasagar, M. *Control System Synthesis: A Factorization Approach*. MIT Press, 1985.

[62] Vieira, A. and Kailath, T. On another approach to the schur-cohn criterion. *IEEE Transaction on Circuits and Systems*, pages 218–220, April 1977.

[63] Villemagne, C. D. and Skelton, R. E. Model reduction using a projection formulation. *Int. J. Control*, 46:2141–2169, 1987.

[64] Villemagne, C.D. and Skelton, R.E. Model reduction using a projection formulation. *International Journal of Control*, 46:21412169, 1987.

[65] Wang, G., Liu, W. Q. and Sreeram, V. A new frequency weighted balanced truncation method and an error bound. *IEEE Trans. on Automatic Control*, 44:1734–1737, 1999.

[66] Wilhelm, R. . *Handbook on Embedded Systems*. CRC Press, 2005.

[67] Willems, J. C. Dissipative dynamical systems. i. general theory. *Arch. Rational Mech. Anal.*, 45:321–351, 1972.

[68] Willems, J. C. Dissipative dynamical systems. ii. linear systems with quadratic supply rates. *Arch. Rational Mech. Anal.*, 45:352–393, 1972.

[69] Youla, D. C. , Bongiorno, J. J. and Lu, C. N. Single-loop feedback stabilization of linear multivariable plants. *Automatica*, 10:159–173, 1974.

[70] Youla, D. C. and Bongiorno, J. J. A feedback theory of two-degree-of-freedom optimal wiener-hopf design. *IEEE Trans. on Automatic Control*, 30:652–665, 1985.

[71] Youla, D. C., Jabr, H. A. and Bongiorno, J. J. Modern wiener-hopf design of optimal controllers: part I. *IEEE Trans. on Automatic Control*, 21:3–13, 1976.

[72] Youla, D. C., Jabr, H. A. and Bongiorno, J. J. Modern wiener-hopf design of optimal controllers: part II. *IEEE Trans. on Automatic Control*, 21:319–338, 1976.

[73] Zhou, K. , D'Souza, C. and Cloutier , J. R. Structurally balanced controller order reduction with guaranteed closed loop performance. *System & Control Letters*, 24:235–242, 1995.

[74] Zhou, K. and Doyle, J. *Essentials of Robust Control*. 1998.

[75] Zhou, K., Doyle, J. and Glover K. *Robust and optimal control*. Prentice Hall, 1995.